Optimization, 1997, Vol.41, pp. 257-278.

NEWTON-TYPE METHODS WITH GENERALIZED DISTANCES FOR CONSTRAINED OPTIMIZATION*

Alfredo N. Iusem† and Michael V. Solodov†

ABSTRACT

We consider a class of interior point algorithms for minimizing a twice continuously differentiable function over a closed convex set with nonempty interior. On one hand, our algorithms can be viewed as an approximate version of the generalized proximal point methods and, on the other hand, as an extension of unconstrained Newton-type methods to the constrained case. Each step consists of solving a strongly convex unconstrained program followed by a one-dimensional search along either a line or a curve segment in the interior of the feasible set. The information about the feasible set is contained in the generalized distance function whose gradient diverges on the boundary of this set. When the feasible set is the whole space, the standard regularized Newton method is a particular case in our framework. We show, under standard assumptions, that every accumulation point of the sequence of iterates satisfies a first order necessary optimality condition for the problem and solves the problem if the objective function is convex. Some computational results are also reported.

Key words. proximal point methods, Newton method, generalized distances, constrained optimization
AMS subject classifications. 90C25, 90C30

---

* Research of the first author is partially supported by CNPq grant 301280/86; the second author is supported in part by CNPq grant 300734/95-6.
† Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, RJ, CEP 22460-320, Brazil. Email : *iusp@impa.br* and *solodov@impa.br*

1 Introduction

We consider the nonlinear programming problem

$$\min_{x \in C} f(x), \tag{1.1}$$

where C is a nonempty closed convex set in $\Re^n$ with nonempty interior C^0 and $f : \Re^n \to \Re$ is a twice continuously differentiable function.

The classical proximal point method generates a sequence of iterates according to the formula

$$x^{i+1} \in \arg\min_{x \in C} f(x) + \lambda_i \|x - x^i\|^2, \tag{1.2}$$

where $\lambda_i > 0$ is a regularization parameter. It was first shown in [18] that in the case when (1.1) is solvable, $f(\cdot)$ is convex and $C = \Re^n$, the sequence $\{x^i\}$ converges to a solution of the problem. Proximal point algorithms have been studied very extensively. We refer the readers to [15] for a survey.

In the general case when C is not the whole space and $f(\cdot)$ is nonconvex, the proximal algorithm is impractical because the subproblems (1.2) are as hard to solve as the original problem itself. Recently, the generalized proximal methods were proposed where the 2-norm regularization term is replaced by a Bregman distance which also has a penalization affect. This generalized method has the form

$$x^{i+1} \in \arg\min_{x \in \Re^n} f(x) + \lambda_i D_g(x, x^i), \tag{1.3}$$

where $D_g(\cdot, \cdot)$ is a Bregman distance induced by a strictly convex function $g(\cdot)$ defined on C (see Section 2 for a formal definition and further discussion). The gradient of D_g diverges on the boundary of C which ensures that the sequence of iterates generated by (1.3) is contained in C^0, the interior of the feasible set. It is clear that (1.3) is more practical than (1.2) in the sense that a sequence of constrained subproblems is replaced by a sequence of unconstrained ones (the latter, presumably, being easier to solve). We refer the readers to [11, 5, 6, 10, 14] for further details on the generalized proximal methods.

It should be noted that when $f(\cdot)$ is nonconvex (1.3) can still be difficult to solve. Thus in [13] the following approximate version of the generalized proximal algorithm was proposed:

$$x^{i+1} = \arg\min_{x \in \Re^n} \langle \nabla f(x^i), x \rangle + \lambda_i D_g(x, x^i), \tag{1.4}$$

where $\langle \cdot, \cdot \rangle$ stands for the usual Euclidean inner product. This algorithm can also be viewed as a steepest descent method with respect to the distance induced by $D_g(\cdot, \cdot)$. Indeed, (1.4) is equivalent to

$$\nabla f(x^i) + \lambda_i \nabla_x D_g(x, x^i) = 0$$

which are the first order optimality conditions for the program

$$\begin{aligned} &\min_x \langle \nabla f(x^i), x \rangle \\ &\text{s.t. } D_g(x, x^i) \le \theta_i \end{aligned}$$

for some θ_i. The subproblems (1.4) are particularly easy to solve when the gradient of $g(\cdot)$ is explicitly invertible on C^0 (see [13]).

In this paper, we propose a more efficient approximate version of the generalized proximal algorithms which can be viewed as a certain extension of the unconstrained Newton-type methods to constrained problems. In particular, we consider an iterative procedure given by the following formula :

$$x^{i+1} = \arg\min_{x \in \Re^n} \langle \nabla f(x^i), x \rangle + \frac{1}{2} \langle (\nabla^2 f(x^i) + \nu_i I)(x - x^i), x - x^i \rangle + \lambda_i D_g(x, x^i), \tag{1.5}$$

where I denotes the $n \times n$ identity matrix and $\nu_i \ge 0$ is chosen so that the matrix $\nabla^2 f(x^i) + \nu_i I$ is positive definite. The connection to quasi-Newton and trust-region methods [9] becomes apparent from the following considerations. The first order optimality conditions for (1.5) are

$$\nabla f(x^i) + (\nabla^2 f(x^i) + \nu_i I)(x - x^i) + \lambda_i \nabla_x D_g(x, x^i) = 0.$$

The latter is, in turn, equivalent to the optimality conditions for the program

$$\begin{aligned} &\min_x \langle \nabla f(x^i), x \rangle + \frac{1}{2} \langle \nabla^2 f(x^i)(x - x^i), x - x^i \rangle \\ &\text{s.t. } \|x - x^i\|^2 \le \theta_i^1, \\ &\quad\;\; D_g(x, x^i) \le \theta_i^2 \end{aligned}$$

for some θ_i^1 and θ_i^2. Since the quadratic model in (1.5) is typically a better local approximation to $f(\cdot)$ than the linear model in (1.4), it is reasonable to expect (1.5) to better approximate the generalized proximal point method given by (1.3). This also becomes clear when the properties of our Algorithm 3.1 are compared with that of Algorithm A3 in [13]. We note that, with an appropriate choice of ν_i, (1.5) is a strongly convex unconstrained program which can be solved very efficiently by a variety of optimization techniques [17] (for example, the standard Newton method applied to (1.5) is globally quadratically convergent). Estimation of ν_i can also be done in an efficient way [9].

It is clear that in both (1.4) or (1.5) λ_i has to be chosen in a special way to guarantee a decrease of the objective function $f(\cdot)$. In particular, λ_i has to be taken sufficiently large so that the step $x^{i+1} - x^i$ be sufficiently small. Or, alternatively, a one-dimensional search along the generated direction can be employed to enforce the decrease of the objective function. In this paper, we consider two algorithms. In the first one, λ_i is fixed for the given iteration. Once the subproblem (1.5) is solved, an inexact Armijo-type search [1] is performed along the generated direction. In the second algorithm, λ_i is treated as a variable and a one-dimensional curvilinear search with respect to λ_i is done for the regularized objective function $f(\cdot) + \eta_i D_g(x^i, \cdot)$, where $\eta_i > 0$ is a parameter.

The paper is organized as follows. Section 2 contains some introduction material on Bregman functions and results that will be used in establishing the properties of our algorithms. In Section 3 we formally state the methods under consideration, and Section 4 contains the convergence analysis. Preliminary computational results are reported in Section 5. We give some concluding remarks is Section 6.

2 Bregman Functions

This section contains the required material for the formulation and convergence analysis of the algorithms considered in this paper.

Let C be a closed convex subset of $\Re^n$ and C^0 be its nonempty interior. Consider a convex real function $g(\cdot)$ defined on C and let $D_g : C \times C^0 \to \Re$ be defined by

$$D_g(x, y) := g(x) - g(y) - \langle \nabla g(y), x - y \rangle. \tag{2.1}$$

The function $g(\cdot)$ is said to be a *Bregman function* (and D_g the *Bregman distance* induced by g) if the following conditions hold:

B1. $g(\cdot)$ is twice continuously differentiable on C^0 and $\nabla^2 g(x)$ is positive definite for all $x \in C^0$.

B2. $g(\cdot)$ is strictly convex and continuous on C.

B3. For all $\delta \in \Re$ the partial level sets $\mathcal{L}(D_g(x,\cdot),\delta) := \{y \in C^0 : D_g(x,y) \leq \delta\}$ are bounded for all $x \in C$.

B4. If $\{y^k\} \subset C^0$ converges to y^* then $D_g(y^*, y^k)$ converges to zero.

B5. If $\{x^k\} \subset C$ and $\{y^k\} \subset C^0$ are sequences such that $\{x^k\}$ is bounded, $\lim_{k\to\infty} y^k = y^*$ and $\lim_{k\to\infty} D_g(x^k, y^k) = 0$ then $\lim_{k\to\infty} x^k = y^*$.

The set C^0 is called the *zone* of $g(\cdot)$. These definitions originate in the results of [2]. It is easy to check that $D_g(x,y) \geq 0$ for all $x \in C$, $y \in C^0$ and $D_g(x,y) = 0$ if and only if $x = y$. We remark that B4 and B5 hold automatically when x^k, y^* are in C^0, as a consequence of B1, B2 and B3, and so they need to be checked only at points in the boundary ∂C of C. It has been proved in [8] that when $C = \Re^n$, a sufficient condition for a convex and differentiable function g to be a Bregman function is $(g(x)/\|x\|) \to \infty$ as $\|x\| \to \infty$. It is easy to check that $D_g(\cdot, y)$ is strictly convex and continuous in C for all $y \in C^0$.

Before presenting examples of Bregman functions, we introduce a subclass to be used in the sequel.

A Bregman function $g(\cdot)$ is said to be *boundary coercive* whenever the following holds :

B6. If $\{x^k\}$ is a sequence contained in C^0 which converges to a point $\bar{x}$ in the boundary ∂C of C, and y is any point in C^0 then $\lim_{k\to\infty}\langle \nabla g(x^k), y - x^k\rangle = -\infty$.

The last property is the key concept in the context of proximal point methods for constrained optimization for the following reason. It is clear from B1-B5 that if $g(\cdot)$ is a Bregman function with zone C^0 and P is an open subset of C^0 then $g(\cdot)$ is also a Bregman function with zone P, which means that one cannot recover C^0 from $g(\cdot)$. Therefore in order to use $D_g(\cdot,\cdot)$ for penalization purposes, $g(\cdot)$ has to possess an additional property. In particular, $g(\cdot)$ should contain information about C. This is precisely the role of B6 because divergence of $\nabla g(\cdot)$ on ∂C makes C univocally defined by $g(\cdot)$. We refer the readers to [4, 3] for further details on boundary coercive Bregman functions.

Example 2.1. $C = \Re^n$, $g(x) = \|x\|^2$. In this case $D_g(x,y) = \|x-y\|^2$.

Example 2.2. $C = \Re^n_+$, $g(x) = \sum_{j=1}^n x_j \log x_j$, extended with continuity to $\partial\Re^n_+$ with the convention that $0 \log 0 = 0$. In this case

$$D_g(x,y) = \sum_{j=1}^n (x_j \log \frac{x_j}{y_j} + y_j - x_j).$$

Example 2.3. $C = \Re^n_+$, $g(x) = \sum_{j=1}^n (x_j^\alpha - x_j^\beta)$ with $\alpha \in (1,+\infty)$, $\beta \in (0,1)$.
For $\alpha = 2$, $\beta = 1/2$ we get

$$D_g(x,y) = \|x-y\|^2 + \sum_{j=1}^n \frac{1}{2\sqrt{y_j}}(\sqrt{x_j} - \sqrt{y_j})^2,$$

and for $\alpha = 1$, $\beta = 1/2$ we get

$$D_g(x,y) = \sum_{j=1}^n \frac{1}{2\sqrt{y_j}}(\sqrt{x_j} - \sqrt{y_j})^2.$$

The Bregman functions of Examples 2.1–2.3 are all boundary coercive.

In the subsequent analysis we will need the following properties of Bregman functions.

Lemma 2.1 *Let $g(\cdot)$ be a Bregman function with zone C^0. Then*

1. $D_g(x,y) - D_g(x,z) - D_g(z,y) = \langle \nabla g(z) - \nabla g(y), x - z \rangle$ *for all* $x \in C$, $y, z \in C^0$.
2. $\nabla_x D_g(x,y) = \nabla g(x) - \nabla g(y)$ *for all* $x, y \in C^0$.
3. $\nabla_y D_g(x,y) = \nabla^2 g(y)(x-y)$ *for all* $x, y \in C^0$.
4. $D_g(x,y) + D_g(y,x) = \langle \nabla g(x) - \nabla g(y), x - y \rangle$ *for all* $x, y \in C^0$.

Proof. All four assertions follow directly from (2.1) and B1-B5. ∎

3 Statement of Algorithms

We are now ready to formally state our algorithms. Let $g(\cdot)$, a Bregman function with zone C^0, be given. Having $x \in C^0$, $\nu \in \Re^n_+$ and $\mu \in \Re^n_+$, define

$$h(x,\nu,\mu) := \arg\min_{y\in\Re^n} \mu\langle\nabla f(x), y\rangle + \frac{\mu}{2}\langle(\nabla^2 f(x)+\nu I)(y-x), y-x\rangle + D_g(y,x), \tag{3.1}$$

where we divided the objective function of (1.5) by λ and set $\mu := 1/\lambda$ (this setting appears to be more convenient in the subsequent analysis).

Let $\psi(y)$ be the minimand in (3.1) and let $\nu \geq 0$ be large enough so that $\nabla^2 f(x) + \nu I$ is positive definite. Then $\psi(\cdot)$ is strongly convex which ensures existence and uniqueness of the minimizer. So $h(x,\nu,\mu)$ is well defined.

We claim now that $h(x,\nu,\mu) \in C^0$. Using (2.1), the definition of $D_g(\cdot,\cdot)$, we can represent $\psi(y)$ as $\psi(y) =: \varphi(y) + g(y)$ where $\varphi(y) = (1/2)\langle Ay, y\rangle + \langle b, y\rangle + c$ with $A = \mu(\nabla^2 f(x) + \nu I)$, $b = \mu\nabla f(x) - \mu Ax - \nabla g(x)$ and $c = (\mu/2)\langle Ax, x\rangle - g(x) - \langle\nabla g(x), x\rangle$. The optimality condition for (3.1) is

$$0 \in (\partial\psi)(h(x,\nu,\mu)) = \partial(\varphi+g)(h(x,\nu,\mu)).$$

It has been proved in [14, Theorem 4.1] that $\partial(\varphi+g)(y) = \emptyset$ for any $y \notin C^0$ and any convex $\varphi(\cdot)$ provided $g(\cdot)$ is boundary coercive with zone C^0. It follows that $h(x,\nu,\mu)$ must belong to C^0 and the claim is established.

Since $h(x,\nu,\mu) \in C^0$, $g(\cdot)$ is differentiable at $h(x,\nu,\mu)$ by B1, and the optimality condition $0 \in (\partial\psi)(h(x,\nu,\mu))$ becomes

$$\mu\nabla f(x) + \mu(\nabla^2 f(x) + \nu I)(h(x,\nu,\mu) - x) + \nabla g(h(x,\nu,\mu)) - \nabla g(x) = 0. \tag{3.2}$$

Finally we claim that $h(\cdot,\cdot,\cdot)$ is continuous. Let

$$\rho(y,x,\nu,\mu) := \mu\nabla f(x) + \mu(\nabla^2 f(x) + \nu I)(y-x) + \nabla g(y) - \nabla g(x).$$

The function $\rho(\cdot,\cdot,\cdot,\cdot)$ is continuous and its Jacobian with respect to y is

$$J(y,x,\nu,\mu) = \mu(\nabla^2 f(x) + \nu I) + \nabla^2 g(y).$$

Using B1, we conclude that the Jacobian is nonsingular for any (y, x, ν, μ) with $x, y \in C^0$ and ν satisfying our assumption. Now the implicit function theorem [17, p.57] ensures continuity of the solution y of the equation $\rho(y, x, \nu, \mu) = 0$ as a function of (x, ν, μ), which is precisely $h(x, \nu, \mu)$. Note that $h(x, \nu, 0) = x$.

We now state our first algorithm.

Algorithm 3.1 *Let the following constants be given :* $\mu_u \geq \mu_l > 0$, $\beta > 0$ *and* $\gamma \in (0, 1)$. *Start with any* $x^0 \in C^0$. *Having* x^i, *check for optimality. If not satisfied, choose* $\mu_i \in [\mu_l, \mu_u]$ *and* ν_i *so that* ρ_i, *the smallest eigenvalue of* $\nabla^2 f(x^i) + \nu_i I$, *is no less than* ρ, *where* $\rho > \beta$. *Compute*

$$x^{i+1} = x^i + \alpha_i (h(x^i, \nu_i, \mu_i) - x^i) \tag{3.3}$$

with α_i *determined by the Armijo-type rule :* $\alpha_i := \gamma^{k_i}$ *where* k_i *is the smallest nonnegative integer* k *such that*

$$f(x^i) - f(x^i + \gamma^k (h(x^i, \nu_i, \mu_i) - x^i)) \geq \beta \gamma^k \|h(x^i, \nu_i, \mu_i) - x^i\|^2. \tag{3.4}$$

This algorithm appears to be more practical than a similar Algorithm A3 in [13] where exact minimization along the direction generated by (1.4) is required. It is not immediately clear if exact line search can be replaced with an Armijo-type rule in [13]. As we shall see, the direction generated by (1.5) is also "better" in the sense that it provides steeper descent for the objective function.

We now state our second algorithm.

Algorithm 3.2 *Let the following constants be given :* $\mu_u > 0$, $\eta_u \geq \eta_l > 0$. *Start with any* $x^0 \in C^0$. *Having* x^i, *check for optimality. If not satisfied, choose* $\eta_i \in [\eta_l, \eta_u]$. *Compute*

$$x^{i+1} = h(x^i, \nu_i, \mu_i),$$

where

$$\mu_i := \arg \min_{\mu \in [0, \mu_u]} f(h(x^i, \nu_i, \mu)) + \eta_i D_g(x^i, h(x^i, \nu_i, \mu)). \tag{3.5}$$

Algorithm 3.2 performs a curvilinear search with the regularized objective function $f(\cdot) + \eta_i D_g(x^i, \cdot)$ and is analogous to Algorithm A1 in [13].

4 Convergence Analysis

From now on, we assume that $f(\cdot)$ is bounded below on C.

Lemma 4.1 *For any sequence $\{x^i\}$ generated by Algorithms 3.1, 3.2 the following properties hold :*

1. $\{x^i\} \subset C^0$;
2. $\{f(x^i)\}$ *is nonincreasing;*
3. $\{f(x^i)\}$ *converges;*
4. $\{x^i\} \subset \mathcal{L}(f, x^0) := \{x \in C \mid f(x) \leq f(x^0)\}$.

Proof. The first assertion follows by recurrence from convexity of C^0 and the fact that $h(x, \nu, \mu) \in C^0$ for all $x \in C^0$. From conditions (3.4) and (3.5) it follows that the sequence $\{f(x^i)\}$ is nonincreasing. This also immediately implies that the last assertion of the lemma holds. Since $\{f(x^i)\}$ is also bounded from below, it converges. ∎

From now on, we assume that the sequence $\{x^i\}$ is contained in some bounded set D (this holds, for example, if the level set $\mathcal{L}(f, x^0)$ is bounded).

Lemma 4.2 *For any sequence $\{x^i\}$ generated by Algorithm 3.1 or 3.2 the following properties hold :*

1.
$$\lim_{i \to \infty} (h(x^i, \nu_i, \mu_i) - x^i) = 0;$$

2.
$$\sum_{i=0}^{\infty} D_g(x^i, h(x^i, \nu_i, \mu_i)) < \infty.$$

Moreover, for Algorithm 3.1,

$$\sum_{i=0}^{\infty} \|h(x^i, \nu_i, \mu_i) - x^i\|^2 < \infty.$$

Proof. We first consider the case of Algorithm 3.1. Because $\{x^i\} \subset D$ and D is bounded, $\nabla f(\cdot)$ is Lipschitz continuous on D with some constant $L > 0$. Denote $d^i := h(x^i, \nu_i, \mu_i) - x^i$. For any $\theta \in [0, 1]$ we have

$$\begin{aligned} f(x^i + \theta d^i) - f(x^i) &= \int_0^\theta \langle \nabla f(x^i + td^i), d^i \rangle dt \\ &= \theta \langle \nabla f(x^i), d^i \rangle + \int_0^\theta \langle \nabla f(x^i + td^i) - \nabla f(x^i), d^i \rangle dt \\ &\leq \theta \langle \nabla f(x^i), d^i \rangle + L \int_0^\theta t \|d^i\|^2 dt \\ &= \theta \left(\langle \nabla f(x^i), d^i \rangle + \frac{L\theta}{2} \|d^i\|^2 \right), \end{aligned} \tag{4.1}$$

where the inequality follows from the Lipschitz continuity of $\nabla f(\cdot)$ and the Cauchy-Schwartz inequality. From (3.2), it follows that

$$\begin{aligned} \langle \nabla f(x^i), d^i \rangle &= -\langle (\nabla^2 f(x^i) + \nu_i I) d^i, d^i \rangle - \frac{1}{\mu_i} \langle \nabla g(h(x^i, \nu_i, \mu_i)) - \nabla g(x^i), d^i \rangle \\ &\leq -\rho_i \|d^i\|^2 - \frac{1}{\mu_i} \left(D_g(x^i, h(x^i, \nu_i, \mu_i)) + D_g(h(x^i, \nu_i, \mu_i), x^i) \right), \end{aligned} \tag{4.2}$$

where the inequality follows from the choice of ν_i (note that $\rho_i \geq \rho > \beta > 0$) and Lemma 2.1.

Combining (4.1) and (4.2) we obtain

$$f(x^i) - f(x^i + \theta d^i) \geq \theta(\rho - L\theta/2)\|d^i\|^2 + \frac{\theta}{\mu_i} \left(D_g(x^i, h(x^i, \nu_i, \mu_i)) + D_g(h(x^i, \nu_i, \mu_i), x^i) \right). \tag{4.3}$$

Hence the inequality

$$f(x^i) - f(x^i + \theta d^i) \geq \beta\theta \|d^i\|^2$$

holds whenever

$$\theta \leq \frac{2}{L}(\rho - \beta).$$

By the stepsize choice (3.4), it follows that either $\alpha_i = 1$ or else when $\theta = \gamma^{-1}\alpha_i$ the last inequality is violated, that is

$$\gamma^{-1}\alpha_i > \frac{2}{L}(\rho - \beta).$$

Hence, for all i,

$$\alpha_i \geq \min\{1; \frac{2\gamma}{L}(\rho - \beta)\} =: \alpha_l > 0 .$$

Then, by (3.3), we have

$$f(x^i) - f(x^{i+1}) \geq \beta\alpha_l \|d^i\|^2.$$

Since the sequence $\{f(x^i)\}$ converges, the left-hand-side of the latter inequality tends to zero and it follows that $\{d^i\}$ also goes to zero. This proves the first assertion.

Moreover, we have that

$$\begin{aligned} f(x^0) - f(x^i) &= \sum_{j=0}^{i-1}(f(x^j) - f(x^{j+1})) \\ &\geq \beta\alpha_l \sum_{j=0}^{i-1} \|d^j\|^2. \end{aligned}$$

Letting $i \to \infty$ in the latter inequality and taking into account that $\{f(x^i)\}$ converges, we conclude that

$$\sum_{i=0}^{\infty} \|d^i\|^2 < \infty.$$

Similarly, assuming $\alpha_i \leq 1$, we obtain from (4.3) that

$$f(x^0) - f(x^i) \geq \frac{\alpha_l}{\mu_u} \sum_{j=0}^{i-1} \Big(D_g(x^i, h(x^i, \nu_i, \mu_i)) + D_g(h(x^i, \nu_i, \mu_i), x^i) \Big) .$$

Hence

$$\sum_{i=0}^{\infty} D_g(x^i, h(x^i, \nu_i, \mu_i)) < \infty.$$

The proof for Algorithm 3.1 is complete.

Now consider Algorithm 3.2. By (3.5), and taking into account that $h(x^i, \nu_i, \mu_i) = x^{i+1}$ and $h(x^i, \nu_i, 0) = x^i$, we have

$$\begin{aligned} f(x^{i+1}) + \eta_i D_g(x^i, x^{i+1}) &= f(h(x^i, \nu_i, \mu_i)) + \eta_i D_g(x^i, h(x^i, \nu_i, \mu_i)) \\ &\leq f(h(x^i, \nu_i, 0)) + \eta_i D_g(x^i, h(x^i, \nu_i, 0)) \\ &= f(x^i). \end{aligned}$$

Hence

$$f(x^0) - f(x^i) \geq \eta_l \sum_{j=0}^{i-1} D_g(x^i, h(x^i, \nu_i, \mu_i)).$$

Letting $i \to \infty$ in the latter inequality and taking into account that $\{f(x^i)\}$ converges, it follows that

$$\sum_{i=0}^{\infty} D_g(x^i, h(x^i, \nu_i, \mu_i)) < \infty.$$

Finally, we show that $\lim_{i\to\infty}(h(x^i, \nu_i, \mu_i) - x^i) = \lim_{i\to\infty}(x^{i+1} - x^i) = 0$. Since the sequence $\{x^i\}$ is bounded, it is sufficient to show that if some subsequence $\{x^{i_j}\}$ converges to some point $\bar{x}$ then $\{x^{i_j+1}\}$ also converges to $\bar{x}$. Since $D_g(x^{i_j}, x^{i_j+1}) \to 0$, the result follows from B5. ∎

The proof for accumulation points of the sequence of iterates that belong to the interior of the feasible set is carried out differently from that for the points on the boundary. We first consider the case of accumulation points contained in C^0, the interior of the feasible set C.

Theorem 4.1 *Let $\{x^i\}$ be any sequence generated by Algorithm 3.1 or 3.2. If there exists an accumulation point $\bar{x}$ of $\{x^i\}$ that belongs to C^0 then $\bar{x}$ is a stationary point of (1.1), that is $\nabla f(\bar{x}) = 0$.*

Proof. Let $\{x^{i_j}\}$ be any subsequence that converges to $\bar{x} \in C^0$. Since $\{x^i\}$ is bounded, so is $\{\nabla^2 f(x^i)\}$. Therefore the sequence $\{\nu_i\}$ is bounded. The sequences $\{\mu_i\}$ and $\{\eta_i\}$ are also bounded by the construction of the algorithms. Passing onto a subsequence, if necessary, we can assume that $\{\nu_{i_j}\} \to \bar{\nu}$, $\{\mu_{i_j}\} \to \bar{\mu}$ and $\{\eta_{i_j}\} \to \bar{\eta}$.

We first consider Algorithm 3.1. By the continuity of $h(\cdot,\cdot,\cdot)$ it follows that

$$\{h(x^{i_j},\nu_{i_j},\mu_{i_j})\} \to h(\bar{x},\bar{\nu},\bar{\mu}).$$

By Lemma 4.2,

$$\{h(x^{i_j},\nu_{i_j},\mu_{i_j}) - x^{i_j}\} \to 0.$$

Since

$$\{x^{i_j}\} \to \bar{x},$$

it follows that

$$h(\bar{x},\bar{\nu},\bar{\mu}) = \bar{x} \in C^0.$$

By (3.2), B1 and the fact that $\bar{\mu} \geq \mu_l > 0$ we conclude that $\nabla f(\bar{x}) = 0$.

Now we turn our attention to Algorithm 3.2. Denote

$$\psi(x,\nu,\eta,\mu) := f(h(x,\nu,\mu) + \eta D_g(x, h(x,\nu,\mu)))$$

and

$$\sigma(x,\nu,\eta) := \arg \min_{\mu\in[0,\mu_u]} \psi(x,\nu,\eta,\mu). \tag{4.4}$$

Differentiating (3.2) with respect to μ we have

$$\begin{aligned} 0 &= \nabla f(x) + (\nabla^2 f(x) + \nu I)(h(x,\nu,\mu) - x) + \mu(\nabla^2 f(x) + \nu I)\frac{d}{d\mu}h(x,\nu,\mu) \\ &\quad +\nabla^2 g(h(x,\nu,\mu))\frac{d}{d\mu}h(x,\nu,\mu). \end{aligned}$$

Denote

$$M(x,\nu,\mu) := \mu(\nabla^2 f(x) + \nu I) + \nabla^2 g(h(x,\nu,\mu)).$$

Note that $M(x,\nu,\mu)$ is positive definite and hence invertible. Then we have

$$\frac{d}{d\mu}h(x,\nu,\mu) = -M(x,\nu,\mu)^{-1}\left(\nabla f(x) + (\nabla^2 f(x) + \nu I)(h(x,\nu,\mu) - x)\right).$$

And since $h(x,\nu,0) = x$ and $M(x,\nu,0) = \nabla^2 g(x)$,

$$\frac{d}{d\mu}h(x,\nu,0) = -(\nabla^2 g(x))^{-1}\nabla f(x). \tag{4.5}$$

We further obtain

$$\begin{aligned}\frac{d}{d\mu}\psi(x,\nu,\eta,\mu) &= \langle \nabla f(h(x,\nu,\mu)), \frac{d}{d\mu}h(x,\nu,\mu)\rangle \\ &\quad +\eta\langle \nabla^2 g(h(x,\nu,\mu))(x-h(x,\nu,\mu)), \frac{d}{d\mu}h(x,\nu,\mu)\rangle,\end{aligned}$$

where the last term is by Lemma 2.1. Taking into account (4.5) and the fact that $h(x,\nu,0)=x$, we have

$$\frac{d}{d\mu}\psi(x,\nu,\eta,0) = -\langle \nabla f(x), (\nabla^2 g(x))^{-1}\nabla f(x)\rangle. \tag{4.6}$$

Suppose that

$$\frac{d}{d\mu}\psi(\bar{x},\bar{\nu},\bar{\eta},0) < 0.$$

By (4.4),

$$\begin{aligned}\psi(\bar{x},\bar{\nu},\bar{\eta},\sigma(\bar{x},\bar{\nu},\bar{\eta})) &< \psi(\bar{x},\bar{\nu},\bar{\eta},0) \\ &= f(\bar{x}).\end{aligned} \tag{4.7}$$

On the other hand,

$$\begin{aligned}f(x^{i_j+1}) &= \psi(x^{i_j},\nu^{i_j},0,\sigma(x^{i_j},\nu^{i_j},\eta^{i_j})) \\ &\le \psi(x^{i_j},\nu^{i_j},\eta^{i_j},\sigma(x^{i_j},\nu^{i_j},\eta^{i_j})) \\ &\le \psi(x^{i_j},\nu^{i_j},\eta^{i_j},\sigma(\bar{x},\bar{\nu},\bar{\eta})),\end{aligned}$$

where the second inequality follows from nonnegativity of η_{i_j} and $D_g(\cdot,\cdot)$ and the last inequality follows from (4.4). Since $\psi(\cdot,\cdot,\cdot,\cdot)$ is continuous and the sequence $\{f(x^i)\}$ converges, taking the limit as $j\to\infty$ in the above relation yields

$$f(\bar{x}) \le \psi(\bar{x},\bar{\nu},\bar{\eta},\sigma(\bar{x},\bar{\nu},\bar{\eta}))$$

which contradicts (4.7). Hence

$$\frac{d}{d\mu}\psi(\bar{x},\bar{\nu},\bar{\eta},0) \ge 0$$

and (4.6) together with the fact that $(\nabla^2 g(x))^{-1}$ is positive definite (by B1) imply that $\nabla f(\bar{x}) = 0$. The proof is complete. ∎

Now we consider the case when accumulation points of the sequence of iterates belong to the boundary of the feasible set. Recall that a point $\bar{x} \in C$ satisfies the first order necessary optimality condition [16] for problem (1.1) if

$$\langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0 \ \text{ for all } x \in C.$$

If $f(\cdot)$ is convex, this condition is also sufficient for optimality. For the case of nonconvex $f(\cdot)$, our result would be that for every $x \in C$ there exists and accumulation point $\bar{x}$ of the sequence $\{x^i\}$ such that $\langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0$ (not necessarily the same $\bar{x}$ for all x). This is slightly different from the above condition. In the convex case, this result implies that every accumulation point of the sequence of iterates solves the problem.

Theorem 4.2 *Let $\{x^i\}$ be any sequence generated by Algorithm 3.1 or 3.2 and, for Algorithm 3.1, let $D_g(x, \cdot)$ be quasiconvex for all $x \in C^0$. Then for all $x \in C$ there exists $\bar{x}$, an accumulation point of $\{x^i\}$, such that $\langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0$.*

If $f(\cdot)$ is convex then every accumulation point of the sequence $\{x^i\}$ solves (1.1).

Proof. First note that if there exists an accumulation point $\bar{x} \in C^0$ then, by Theorem 4.1, $\nabla f(\bar{x}) = 0$ and the assertion is true.

Consider the remaining case when all accumulation points of the sequence $\{x^i\}$ belong to ∂C, the boundary of C. We first show that for any $x \in C^0$, $\quad D_g(x, x^i) \to \infty$. Suppose not, that is there exist $x \in C^0$ and a subsequence $\{x^{i_j}\}$ such that the sequence $\{D_g(x, x^{i_j})\}$ is bounded. Passing onto a subsequence, if necessary, let $\{x^{i_j}\} \to \hat{x} \in \partial C$. By B6,

$$\begin{aligned} -\infty &= \lim_{j \to \infty} \langle \nabla g(x^{i_j}), x - x^{i_j} \rangle \\ &= \lim_{j \to \infty} (g(x) - g(x^{i_j}) - D_g(x, x^{i_j})) \\ &= g(x) - g(\hat{x}) - \lim_{j \to \infty} D_g(x, x^{i_j}) \end{aligned}$$

which contradicts the assumption that $\{D_g(x, x^{i_j})\}$ is bounded. Hence $D_g(x, x^i) \to \infty$ as claimed.

By (3.2), for any $x \in C^0$,

$$\begin{aligned}
\mu_i\langle \nabla f(x^i), x - h(x^i,\nu_i,\mu_i)\rangle &= \mu_i\langle (\nabla^2 f(x^i) + \nu_i I)(x^i - h(x^i,\nu_i,\mu_i)), x - h(x^i,\nu_i,\mu_i)\rangle \\
&\quad + \langle \nabla g(x^i) - \nabla g(h(x^i,\nu_i,\mu_i)), x - h(x^i,\nu_i,\mu_i)\rangle \\
&= \mu_i\langle (\nabla^2 f(x^i) + \nu_i I)(x^i - h(x^i,\nu_i,\mu_i)), x - h(x^i,\nu_i,\mu_i)\rangle \\
&\quad + D_g(x, h(x^i,\nu_i,\mu_i)) - D_g(x,x^i) + D_g(h(x^i,\nu_i,\mu_i),x),
\end{aligned} \tag{4.8}$$

where the last equality follows from Lemma 2.1. Now fix $x \in C^0$. Since $D_g(x,x^i) \to \infty$, there exists a subsequence $\{x^{i_j}\}$ such that

$$D_g(x, x^{i_j+1}) > D_g(x, x^{i_j}).$$

Note that, for Algorithm 3.2, $x^{i_j+1} = h(x^{i_j},\nu_{i_j},\mu_{i_j})$. For Algorithm 3.1, by quasiconvexity of $D_g(x,\cdot)$ and the fact that x^{i_j+1} is on a line between x^{i_j} and $h(x^{i_j},\nu_{i_j},\mu_{i_j})$, it follows that (see [16])

$$D_g(x,x^{i_j}) < D_g(x,x^{i_j+1}) \le \max\{D_g(x,x^{i_j}), D_g(h(x^{i_j},\nu_{i_j},\mu_{i_j}))\}$$

implying that

$$D_g(x,x^{i_j}) < D_g(h(x^{i_j},\nu_{i_j},\mu_{i_j})).$$

Therefore, for either algorithm, dividing both sides in (4.8) by $\mu_{i_j} > 0$ gives

$$\langle \nabla f(x^{i_j}), x - h(x^{i_j},\nu_{i_j},\mu_{i_j})\rangle \ge \langle (\nabla^2 f(x^{i_j}) + \nu_{i_j} I)(x^{i_j} - h(x^{i_j},\nu_{i_j},\mu_{i_j})), x - h(x^{i_j},\nu_{i_j},\mu_{i_j})\rangle. \tag{4.9}$$

Passing onto a subsequence, if necessary, we can assume that $\{x^{i_j}\} \to \bar{x}$. Because, by Lemma 4.2, $\{x^{i_j} - h(x^{i_j},\nu_{i_j},\mu_{i_j})\} \to 0$ it follows that $\{h(x^{i_j},\nu_{i_j},\mu_{i_j})\} \to \bar{x}$. Therefore, taking the limits as $j \to \infty$, (4.9) gives

$$\langle \nabla f(\bar{x}), x - \bar{x}\rangle \ge 0.$$

To complete the proof we next extend this result to points x on the boundary. If $x \in \partial C$, take any sequence $\{y^k\} \subset C^0$ such that $\{y^k\} \to x$. By the preceding considerations there exists a sequence $\{\bar{x}^k\}$ of accumulation points of $\{x^i\}$ so that

$$\langle \nabla f(\bar{x}^k), y^k - \bar{x}^k\rangle \ge 0.$$

Since the set of accumulation points of $\{x^i\}$ is closed and bounded, there exists a subsequence of $\{\bar{x}^k\}$ converging to some point $\bar{x}$ which is also an accumulation point of $\{x^i\}$. Taking limits along this subsequence in the last inequality establishes the claim.

Now assume that $f(\cdot)$ is convex. First note that, by the convergence of $\{f(x^i)\}$, $f(\bar{x}) = \bar{f}$ for all accumulation points $\bar{x}$ of the sequence $\{x^i\}$. Take any $x \in C$. By the first part of the theorem, there exists $\bar{x}$, an accumulation point of $\{x^i\}$, such that $\langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0$. By the convexity of $f(\cdot)$, we obtain

$$\begin{aligned} 0 &\leq \langle \nabla f(\bar{x}), x - \bar{x} \rangle \\ &\leq f(x) - f(\bar{x}) \\ &= f(x) - \bar{f}. \end{aligned}$$

Hence $\bar{f} \leq f(x)$ for any $x \in C$. Since $f(\bar{x}) = \bar{f}$ for all accumulation points $\bar{x}$ of $\{x^i\}$, these points are solutions of (1.1).

The proof is complete. ∎

The requirement of quasiconvexity of $D_g(x, \cdot)$, necessary for the convergence analysis of Algorithm 3.1, imposes strong limitations on the family of Bregman functions. It can be proved that Bregman functions with zone $\Re^n$ such that $D_g(x, \cdot)$ is quasiconvex are necessarily quadratic functions. In that case, the proximal point method with Bregman functions becomes identical to the classical method, up to a linear transformation of variables. The situation improves for some special classes of convex sets. For $C = \Re^n_+$, the Bregman function of Example 2.2 has the required property (the one in Example 2.3 does not). Bregman functions based on the $x \log x$ entropy function such that $D_g(x, \cdot)$ is quasiconvex (and, in fact, convex) can be found in [4] for the cases in which C is a box, a polyhedron, or a ball.

5 Computational Results

In this section, we report some preliminary computational results. We implemented Algorithm 3.1 in Matlab to solve a minimization reformulation of the the linear complementarity problem [7], which is to find $y \in \Re^n$ such that

$$y \geq 0, \quad My + q \geq 0, \quad \langle y, My + q \rangle = 0,$$

Problem			
Name	n	**iter.**[1]	**CPU**[2]
RanLCP1	10	21	43.0
RanLCP2	100	574	274
RanLCP3	200	2760	3152
RanLCP4	10	18	27
RanLCP5	100	313	216
RanLCP6	200	408	342

Table 1: **Results for Algorithm 3.1 on LCP Reformulation.**

[1] The starting point is $(y^0, z^0) = (1, \ldots, 1)$, and the termination criterion is $\| \max\{0, -y\}, \max\{0, -My - q\}, \langle y, My + q\rangle \| < 10^{-3}$, where the maximum is taken component-wise.

[2] Time (in seconds) obtained using the intristic Matlab function etime and with codes ran on SUN SPARCstation 5; does not include time to read problem data.

where M is an $n \times n$ real matrix and $q \in \Re^n$. In particular, we use the following bound constrained minimization reformulation (see [12])

$$\min_{(y,z)\geq 0} f(y,z) := \frac{1}{2}\|My + q - z\|^2 + \frac{1}{2}(\langle y, z\rangle)^2.$$

As the Bregman function, we chose the $x \log x$ entropy function from Example 2.2. To solve subproblems (1.5) we used the unconstrained minimization solver fminu.m from the Matlab optimization toolbox.

For test problems, we generated dense monotone linear complementarity problems in a manner similar to [19]. The first three (respectively, fourth to sixth) test problems were randomly generated with

$$M = \omega E E^T + E - E^T,$$

where $\omega = 0$ (respectively, $\omega = 1$) and every entry of the $n \times n$ matrix E was uniformly generated from $[-5, 5]$, and with $q = -M\bar{y} + \bar{z}$, where each entry of $\bar{y}$ has equal probability of being 0 or being uniformly generated from $[5, 10]$ and each entry of $\bar{z}$ is 0 if the corresponding entry of $\bar{y}$ is not 0 and otherwise has equal probability of being 0 or being uniformly generated from $[5, 10]$ (so $\bar{y}$ is a solution). In our implementation of Algorithm 3.1, we chose $\mu_l = 1, \mu_u = 2, \gamma = 1/2, \beta = 1$. The test results are summarized in Table 1.

6 Concluding Remarks

We have presented a new class of interior point algorithms for solving constrained optimization problems. The new methods can be viewed as an approximate implementation of the generalized proximal point algorithms with Bregman distances, and also as an extension of unconstrained Newton methods to problems with constraints. Every iteration of the proposed algorithms consists of solving a simple strongly convex unconstrained program followed by a one-dimensional search along a line or a curve segment in the interior of the feasible set.

In conclusion, we compare the basic structure of the algorithms considered here with the methods in [13] and [14] and also with the trust-region Newton-type methods. In the case of the exact proximal point methods with Bregman functions (e.g. [14]), the subproblems consist of solving the system of equations

$$\mu_i \nabla f(x) + \nabla g(x) = \nabla g(x^i). \tag{6.1}$$

In this case no line search is necessary. On the other hand, the convexity of $f(\cdot)$ is required for the existence of solution of (6.1). Furthermore, the system of equations (6.1) can be highly nonlinear due to the presence of the gradient of $f(\cdot)$ in the left-hand-side. Thus (6.1) can be very difficult to solve.

If we approximate $f(\cdot)$ by a linear model at x^i, as is done in [13], the following system has to be solved :

$$\nabla g(x) = \nabla g(x^i) - \mu_i \nabla f(x^i), \tag{6.2}$$

which can be done explicitly when $\nabla g(\cdot)$ is invertible on C^0. The latter is the case when the feasible set has a simple structure, for instance when it is an orthant, a polyhedron, a box, or a ball (see [4]). The resulting algorithms require a line search because otherwise the objective function may not decrease, but they can also handle nonconvex problems.

The algorithms considered in this paper use a regularized quadratic approximation of $f(\cdot)$ at x^i. This class of methods can be regarded as a compromise between the two previous approaches. The subproblems are of the type

$$A_i x + \nabla g(x) = b^i \tag{6.3}$$

with a positive definite matrix A_i. These subproblems cannot be solved explicitly but they are, in general, much simpler than (6.1) because the nonlinearities due to $f(\cdot)$ have been eliminated. In the cases of nonnegativity or box constraints, $g(\cdot)$ is typically separable (see [4]). In that case, the Jacobian matrix of the left-hand-side of (6.3) is constant except in its diagonal. As in the case of (6.2), line search procedures are necessary to enforce the decrease of the objective function, and the algorithm works also for the nonconvex case.

Finally, in the case of Newton-type methods with trust regions [9], the subproblems consist of minimizing a quadratic function over a trust region which, in the constrained case, must carry the information about the feasible set. Our algorithm has unconstrained subproblems which are equivalent, in view of (6.3), to minimizing a strongly convex quadratic function plus $g(x)$. The information about the feasible set is now encapsulated in $g(\cdot)$.

References

[1] L. Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16:1–3, 1966.

[2] L.M. Bregman. The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967.

[3] R.S. Burachik and A.N. Iusem. A generalized proximal point algorithm for the variational inequality problem in a Hilbert space. Technical Report MAT-07/95, Pontificia Universidade Catolica, Rio de Janeiro, Brazil, 1995. SIAM J. on Optimization, accepted for publication.

[4] Y. Censor, A.N. Iusem, and S.A. Zenios. An interior point method with Bregman functions for the variational inequality problem with paramonotone operators. Technical

Report 94-03, Department of Public and Business Administration, University of Cyprus, Nicosia, Cyprus, 1994. Mathematical Programming, accepted for publication.

[5] Y. Censor and S. Zenios. The proximal minimization algorithm with D-functions. *Journal of Optimization Theory and Applications*, 73:451–464, 1992.

[6] G. Chen and M. Teboulle. Convergence analysis of proximal-like optimization algorithm using Bregman functions. *SIAM Journal on Optimization*, 3:538–543, 1993.

[7] R.W. Cottle, J.-S. Pang, and R.E. Stone. *The Linear Complementarity Problem.* Academic Press, New York, 1992.

[8] A.R. De Pierro and A.N. Iusem. A relaxed version of Bregman's method for convex programming. *Journal of Optimization Theory and Applications*, 51:421–440, 1986.

[9] J.E. Dennis and R.B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, N.J., 1983.

[10] J. Eckstein. Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18:202–226, 1993.

[11] P.P.B. Eggermont. Multiplicative iterative algorithms for convex programming. *Linear Algebra and its Applications*, 130:25–32, 1990.

[12] A. Friedlander, J.M. Martínez, and S.A. Santos. Solution of linear complementarity problems using minimization with simple bounds. *Journal of Global Optimization*, 6:253–267, 1995.

[13] A. N. Iusem. Steepest descent methods with generalized distances for constrained optimization, 1994. *Acta Applicandae Mathematicae*, accepted for publication.

[14] A. N. Iusem. On some properties of generalized proximal point methods for quadratic and linear programming. *Journal of Optimization Theory and Applications*, 85:596–612, 1995.

[15] B. Lemaire. The proximal algorithm. In J.P. Penot, editor, *International Series of Numerical Mathematics*, pages 73–87. Birkhauser, Basel, 1989.

[16] O.L. Mangasarian. *Nonlinear Programming.* McGraw–Hill, New York, 1969.

[17] B.T. Polyak. *Introduction to Optimization.* Optimization Software, Inc., Publications Division, New York, 1987.

[18] R.T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM Journal on Control and Optimization*, 14(5):877–898, 1976.

[19] M. V. Solodov and P. Tseng. Modified projection-type methods for monotone variational inequalities. *SIAM Journal on Control and Optimization*, 34:1814–1830, 1996.