Comparison of two sets of first-order conditions as bases of interior-point Newton methods for optimization with simple bounds*

Diane C. Jamrog† Richard A. Tapia‡ Yin Zhang§

June, 2001

*The first author was supported in part by the W. M. Keck Center for Computational Biology (NSF RTG BIR-94-13229, NLM RTG 2T15LM07093 and NLM 5 T15 LM07093), the SAS (NSF Grant HRD-9817555) and the AGEP programs. The second author was supported in part DOE/LANL Contract 03891-99-23. The third author was supported in part by DOE Grant DE-FG03-97ER25331, DOE/LANL Contract 03891-99-23 and NSF Grant DMS-9973339. The authors wish to thank Luís Vicente and Amr El-Bakry for helpful discussions on issues related to this paper, and Theresa Parks for helping improve the presentation of the paper.

†Graduate student, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, USA.

‡Professor, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, USA.

§Associate Professor, Department of Computational and Applied Mathematics, Rice University, Houston, Texas, USA.

Abstract

In this paper, we compare the behavior of two Newton interior-point methods derived from two different first-order necessary conditions for the same nonlinear optimization problem with simple bounds. One set of conditions was proposed by Coleman and Li; the other is the standard KKT set of conditions. We discuss a perturbation of the CL conditions for problems with one-sided bounds and the difficulties involved in extending this to problems with general bounds. We study the numerical behavior of the Newton method applied to the systems of equations associated with the unperturbed and perturbed necessary conditions. Preliminary numerical results for convex quadratic objective functions indicate that, for this class of problems, Newton's method based on the perturbed KKT formulation appears to be the most robust.

Key Words. Optimization with simple bounds, first-order necessary conditions, interior-point Newton methods.

1 Introduction

For inequality constrained optimization problems, a useful interior-point method can be derived by applying Newton's method to the Karush-Kuhn-Tucker (KKT) conditions. The KKT conditions are first-order necessary conditions for nonlinear programming provided that certain constraint qualification holds. In this study, we will investigate an alternate interior-point method by applying Newton's method to a different set of first-order necessary conditions, the Coleman-Li (CL) conditions. We seek to determine whether there are advantages to using one framework over the other.

Recently, many papers have focused on comparing different formulations of the Newton method. For inequality-constrained optimization, Villallobos, Tapia, and Zhang (Ref. [1, 2]) compare the Newton primal-dual interior-point method and the Newton logarithmic barrier function method and demonstrate different behavior in the regions of convergence associated with the two methods for linear (Ref. [1]) and nonlinear programs (Ref. [2]). A recent work by Vicente (Ref. [3]) analyzes two formulations of the Newton method for nonlinear optimization with nonnegativity constraints, one derived from the KKT conditions and the other from the CL conditions, and presents a bound on the difference between the two Newton steps as the sequences of iterates converge. Another work by Vicente (Ref. [4]) proves local convergence of a quasi-Newton method based on a variation of the CL conditions for nonlinear optimization with equality and nonnegativity constraints.

Newton's method based on the CL conditions with a reflective line search was first tested by Coleman and Li on quadratic functions (Ref. [5]). Nonlinear optimization of a quadratic objective function is a fundamental problem, and quadratic functions, especially convex ones, are a natural class of objective functions for a first test of new algorithms. Therefore, in this paper we will consider optimization problems of convex quadratic functions with simple bounds on variables. These problems are a subclass of so-called convex quadratic programs.

It is well-known that under standard conditions Newton's method is guaranteed to have local convergence. The precise measure of locality is problem-dependent. Very often Newton's method actually converges in a wide region,

which may be loosely characterized as semi-global convergence. Since we are interested in the overall behavior of the method, we selected starting points that are distributed throughout the feasible region. By starting Newton's method not necessarily closed to the solution, we hope to test the robustness of the local behavior of Newton's method.

The paper is organized as follows. In Section 2, we present background material. We define the optimization problem and describe two first-order necessary conditions for the problem: the KKT conditions and the CL conditions. Then we define the Newton steps associated with these two sets of conditions. In Section 3, we present the perturbed KKT (PKKT) conditions, derive a perturbation of the CL conditions for optimization problems with one-sided bounds, and explain the difficulty in devising a sensible perturbation for problems with general bounds. Section 4 presents the interior-point Newton's method algorithms used to compare the frameworks. In Section 5, we present numerical results for a two-dimensional problem and for convex optimization problems with only lower bounds as well as with more general bounds. Finally, we give conclusions in Section 6.

2 Background

2.1 Minimization Problem with Simple Bounds

We consider the following nonlinear minimization problem with simple bounds:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & l \leq x \leq u, \end{array} \tag{1}$$

where $f(x) : \Re^n \to \Re$ is twice continuously differentiable, $x, l, u \in \Re^n$, $l < u$, and the inequalities should be read in a component-wise manner. In general, the components of $-l$ and u are allowed to be positive infinity.

2.2 The KKT Conditions

The KKT conditions are a set of first-order necessary conditions, first derived by Karush (Ref. [6]) and then independently by Kuhn and Tucker (Ref. [7]). The

Lagrangian $\ell : \Re^n \times \Re^n \times \Re^n \rightarrow \Re$ for (1) and its gradient with respect to x are defined by:

$$\begin{aligned} \ell(x,y,z) &= f(x) - y^T(x-l) - z^T(u-x) \\ \nabla_x \ell(x,y,z) &= \nabla f(x) - y + z, \end{aligned}$$

where $y, z \in \Re^n$ are vectors of Lagrange multipliers. The KKT conditions for (1) are defined by:

$$\nabla f(x) - y + z = 0 \tag{2a}$$
$$Y(x-l) = 0 \tag{2b}$$
$$Z(u-x) = 0 \tag{2c}$$
$$x - l, u - x, y, z \geq 0, \tag{2d}$$

where Y and Z denote diagonal matrices with diagonals y and z, respectively, i.e., $Y = \text{Diag}(y)$ and $Z = \text{Diag}(z)$. For more background on Lagrange multiplier theory consult (Ref. [8]), for example.

The interior-point methodology consists of applying Newton's method to the square system of equalities in the first-order necessary conditions and scaling the Newton update so that the new iterate satisfies the inequality constraints. Applying the Newton method to the three equalities in the KKT conditions (2), we see that a Newton step $(\Delta x, \Delta y, \Delta z)^T \in \Re^{3n}$ for (2) satisfies:

$$\begin{bmatrix} \nabla^2 f(x) & -I & I \\ Y & X-L & 0 \\ -Z & 0 & U-X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} \nabla f(x) - y + z \\ Y(x-l) \\ Z(u-x) \end{bmatrix}. \tag{3}$$

2.3 The CL Conditions

The CL conditions have been used as a framework for a Newton interior-point method called an interior-reflective Newton method (Ref. [5, 9]). The correction to the iterate is a Newton step scaled using a line search along a piecewise linear path. In a later paper, the direction of the Newton step is modified by a trust-region approach (Ref. [10]). In both methods, the Newton step is asymptotically used near the solution, in order to obtain quadratic convergence.

The CL first-order necessary conditions are:

$$D(x)\,\nabla f(x) \;=\; 0 \tag{4a}$$

$$x - l,\; u - x \;\geq\; 0, \tag{4b}$$

where $D(x)$ is a diagonal matrix with diagonal $d(x) \in \Re^n$, defined as:

$$d(x)_i = \begin{cases} u_i - x_i & \text{if } \nabla f(x)_i < 0 \text{ and } u_i \text{ is finite} \\ x_i - l_i & \text{if } \nabla f(x)_i \geq 0 \text{ and } l_i \text{ is finite} \\ 1 & \text{if } \nabla f(x)_i < 0 \text{ and } u_i = \infty \\ & \text{or if } \nabla f(x)_i \geq 0 \text{ and } l_i = -\infty, \end{cases} \tag{5}$$

where $d(x)_i$, $i = 1, 2, \cdots, n$, is the i-th component of $d(x)$, and similarly for $\nabla f(x)_i$. The definition above is a simplification of the notation presented in (Ref. [5]. To see that the CL conditions are first-order necessary conditions, suppose that x^* is a solution to (1) and notice that

(i) if $l_i < x_i^* < u_i$, then $\nabla f(x^*)_i = 0$ and hence $(D(x^*)\nabla f(x^*))_i = 0$;

(ii) if $x_i^* = u_i$, then $\nabla f(x^*)_i \leq 0$; if $\nabla f(x^*)_i = 0$, then $(D(x^*)\nabla f(x^*))_i = 0$; if not, then $(D(x^*)\nabla f(x^*))_i = (u_i - x_i^*)\nabla f(x^*)_i = 0$ since $u_i - x_i^* = 0$; thus $(D(x^*)\nabla f(x^*))_i = 0$ in either case;

(iii) similarly, if $x_i^* = l_i$, then $\nabla f(x^*)_i \geq 0$ and consequently $(D(x^*)\nabla f(x^*))_i = (x_i^* - l_i)\nabla f(x^*)_i = 0$.

Again we apply Newton's method to the square system of equalities contained in the first-order necessary conditions (4). Let the Jacobian of $d(x)$ be denoted by $J^d(x)$. Then a Newton step Δx is defined by:

$$\left[D(x)\,\nabla^2 f(x) + \mathrm{Diag}\left(\nabla f(x)\right) J^d\,\right] \Delta x = -D(x)\,\nabla f(x).$$

The diagonal matrix $J^d(x)$ is not defined when $\nabla f(x)_i$ is equal to zero, since $d(x)_i$ has a discontinuity at such an x_i. In this case, following Coleman and Li, we define $[J^d(x)]_{ii}$ to be zero. If the bounds are finite, the Newton step given above can be simplified to:

$$\left[D(x)\,\nabla^2 f(x) + \mathrm{Diag}\left(|\nabla f(x)|\right)\right] \Delta x = -D(x)\,\nabla f(x), \tag{6}$$

where the absolute value of a vector is taken component-wise, since in this case the diagonal matrix $\mathrm{Diag}(\nabla f(x))J^d(x)$ is defined as:

$$\left[\mathrm{Diag}(\nabla f(x))J^d(x)\right]_{ii} = \begin{cases} \nabla f(x)_i \frac{d}{dx_i}(u_i - x_i) & \text{if } \nabla f(x)_i < 0 \text{ and } u_i \text{ is finite} \\ \nabla f(x)_i \frac{d}{dx_i}(x_i - l_i) & \text{if } \nabla f(x)_i \geq 0 \text{ and } l_i \text{ is finite} \\ 0 & \text{if } \nabla f(x)_i < 0 \text{ and } u_i = \infty \\ & \text{or if } \nabla f(x)_i \geq 0 \text{ and } l_i = -\infty. \end{cases}$$

3 Perturbations of the First-Order Conditions

In the following subsections we present justifications for perturbing the KKT and CL conditions. We also present the perturbed KKT (PKKT) conditions and the perturbed CL (PCL) conditions for problems having only lower bounds. Then we discuss the difficulties encountered when trying to derive a perturbation of the CL conditions for problems with general bounds.

3.1 Nonconvergence: Sticking to the Boundary

The iterates of Newton's method based on the KKT formulation may fail to converge if they are close to the boundary of the feasible set and unable to move away, i.e., they "stick" to the boundary. This behavior was first observed by Tapia (Ref. [11] and is a result of the linearized complementarity equations in (3).

Suppose that $x_i = l_i$ for some i. Newton's method replaces complementarity: $y_i(x_i - l_i) = 0$, with linearized complementary:

$$y_i \Delta x_i + (x_i - l_i)\Delta y_i = -y_i(x_i - l_i).$$

If $x_i = l_i$ and $y_i > 0$, then necessarily Δx_i equals zero, and Newton's method cannot change the value of x_i. In practice, if x_i is nearly equal to l_i, then Δx_i can be very small. Thus, if an iterate approaches the boundary of the feasible set too quickly, it will not be able to move away from it; and it may be the "wrong" boundary that does not contain any solution.

Now let us investigate the behavior of Newton's method based on the CL formulation (6). Again suppose $x_i = l_i$ and, additionally, $\nabla f(x)_i > 0$. Then the

ith row of $D(x)\nabla^2 f(x)$ is equal to the zero vector, and the ith row of (6) becomes:

$$\nabla f(x)_i \Delta x_i = 0.$$

Again, $\Delta x_i = 0$ and Newton's method will not be able to change the iterate. If $x_i = l_i$ and $\nabla f(x)_i < 0$, the iterate will not necessarily stick to the boundary. We obtain the same result if we suppose $x_i = u_i$ and $\nabla f(x)_i < 0$. So, even if the ith constraint is not active at the solution, it may be responsible for nonconvergence by causing the iterates to stick to that part of the boundary.

3.2 The Perturbed KKT Conditions

In order to prevent the behavior of sticking to boundaries, we can modify the equations in the KKT conditions (2) slightly by perturbing the right-hand sides. The following motivation of the perturbation closely mirrors that given by Tapia (Ref. [12]).

The perturbed KKT (PKKT) conditions are:

$$\begin{aligned} \nabla f(x) - y + z &= 0 && \text{(7a)} \\ Y(x - l) &= \mu e && \text{(7b)} \\ Z(u - x) &= \mu e && \text{(7c)} \\ x - l, u - x, y, z &\geq 0, && \text{(7d)} \end{aligned}$$

where $\mu > 0$ and $e \in \Re^n$ is a vector of ones. Newton's method applied to the system of equations in (7) yields

$$\begin{bmatrix} \nabla^2 f(x) & -I & I \\ Y & X - L & 0 \\ -Z & 0 & U - X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = - \begin{bmatrix} \nabla f(x) - y + z \\ Y(x - l) - \mu e \\ Z(u - x) - \mu e \end{bmatrix}. \qquad (8)$$

The PKKT conditions can be motivated by the fact that perturbing linearized complementarity allows the iterates to move away from the boundary (Ref. [12]). The traditional motivation for using these conditions as a framework is the notion of a central path parameterized by μ. It is known that under suitable conditions, (7) has a unique solution for all sufficiently small $\mu > 0$, and as $\mu \to 0$, the corresponding solution path converges to the solution. This path in a sense remains central in the feasible set and hence is called the central path.

Let the system of equations in (7) be denoted $F_\mu(x, y, z; \mu) = 0$, and let x^*_μ be the solution of that system. Then the central path C_P is defined by

$$C_P = \{x^*_\mu : \ F(x^*_\mu ; \ \mu) = 0; \quad x - l, u - x, y, z \geq 0; \quad 0 < \mu < \hat{\mu}\},$$

for some $\hat{\mu} > 0$. For existence and smoothness of the central path see Fiacco and McCormick (Ref. [13]).

3.3 Perturbing the CL Condition: Lower Bounds Only

In this subsection, we introduce and justify a perturbation scheme of the CL conditions for problems with one-sided bounds. Without loss of generality, we will consider problems with only lower bounds:

$$\begin{array}{rl} \min & f(x) \\ \text{subject to} & x - l \geq 0, \end{array} \tag{9}$$

where $f(x) : \Re^n \to \Re$ is twice continuously differentiable. The following development follows a line similar to that presented in (Ref. [14]) where the inequality constraints are nonnegativity only.

The KKT conditions for (9) are $\nabla f(x) - y = 0, \ Y(x - l) = 0$, and $x - l, y \geq 0$, leading to the following KKT system:

$$\begin{bmatrix} \nabla^2 f(x) & -I \\ Y & X - L \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = - \begin{bmatrix} \nabla f(x) - y \\ Y(x - l) \end{bmatrix}. \tag{10}$$

The CL conditions for (9) are $D(x) \nabla f(x) = 0$, and $x - l \geq 0$, leading to the following CL system:

$$\left[D(x) \nabla^2 f(x) + \text{Diag}\left(\nabla f(x)\right) J^d(x) \right] \Delta x = -D(x) \nabla f(x), \tag{11}$$

where the diagonal matrix $\text{Diag}(\nabla f(x)) \, J^d(x)$ is defined as:

$$\left[\left(\text{Diag}\left(\nabla f(x)\right) J^d(x)\right]_{ii} \right. \ = \begin{cases} \nabla f(x)_i & \text{if } \nabla f(x)_i > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Supposing $d(x) > 0$ and multiplying the CL system by $D^{-1}(x)$, we obtain

$$\left[\nabla^2 f(x) + D^{-1}(x) \text{Diag}(\nabla f(x)) J^d(x) \right] \Delta x = -\nabla f(x). \tag{12}$$

On the other hand, if we solve Δy from the second equation in (10) and substitute it into the first equation, we arrive at the equation below corresponding to (12):

$$\left[\nabla^2 f(x) + (X - L)^{-1} Y\right] \Delta x = -\nabla f(x). \tag{13}$$

Notice that the two equalities differ only in the diagonal matrices added to $\nabla^2 f(x)$. The following proposition allows us to relate these two matrices. To simplify the notation we will write $D(x^*)$ as D^*, $f(x^*)$ as f^*, and so on.

Proposition 3.1 *Let x^* be a solution of* (9) *with associated Lagrange multipliers y^*. Then (i) $D^* = X^* - L$; and (ii) $Diag(\nabla f^*) J^d(x) = Y^*$.*

Proof:

(i) Assume that x^* is a solution of (9). Then either $x_i^* = l_i$ or $x_i^* > l_i$. If $x_i^* = l_i$, then $\nabla f_i^* \geq 0$. If $x_i^* > l_i$, then $\nabla f_i^* = 0$. Thus, from definition (5), $d_i^* = x_i^* - l_i$, i.e., $D^* \equiv X^* - L$.

(ii) If $x_i^* = l_i$, then $\nabla f_i^* \geq 0$. Suppose $\nabla f_i^* > 0$. Then $(\mathrm{Diag}(\nabla f^*) J^d(x^*))_{ii} = \nabla f_i^*$. Since $\nabla_x \ell^* = 0$, $\nabla f_i^* = y_i^*$. Now suppose that $\nabla f_i^* = 0$. Then $[\mathrm{Diag}(\nabla f^*) J^d(x^*)]_{ii} = 0$. Again since $\nabla_x \ell^* = 0$ and $\nabla f_i^* = 0$, we have $y_i^* = 0$. Thus $[\mathrm{Diag}(\nabla f^*) J^d(x^*)]_{ii} = y_i^* = 0$. Now, if $x_i^* > l_i$, then $\nabla f_i^* = 0$ and $[\mathrm{Diag}(\nabla f^*) J^d(x^*)]_{ii} = y_i^*$ from the same argument. Hence in all cases $\mathrm{Diag}(\nabla f^*) J^d(x^*) \equiv Y^*$.

The proposition says that $D(x)$ and $\mathrm{Diag}(\nabla f(x)) J^d(x)$ can be viewed as approximations to $X - L$ and Y, respectively, at least for x close to a solution x^*.

In a similar way, if we write the PKKT system associated with problem (9) and then solve for Δx, we obtain:

$$\left[\nabla^2 f(x) + (X - L)^{-1} Y\right] \Delta x = -\nabla f(x) + \mu (X - L)^{-1} e. \tag{14}$$

Comparing (13) and (14), we see that a perturbation term $\mu(X - L)^{-1} e$ has been added to the right-hand side of the equation. Since $D(x)$ can be viewed as an approximation to $X - L$, (14) leads us to consider the following perturbation of (12):

$$\left[\nabla^2 f(x) + D^{-1}(x)\ \mathrm{Diag}(\nabla f(x)) J^d(x)\right] \Delta x = -\nabla f(x) + \mu D^{-1}(x) e. \tag{15}$$

Multiplying (15) by $D(x)$, we obtain:

$$\left[D(x)\nabla^2 f(x) + \mathrm{Diag}(\nabla f(x))J^d(x)\right]\Delta x = -D(x)\nabla f(x) + \mu e, \tag{16}$$

which can be also be derived by applying Newton's method to

$$D(x)\nabla f(x) = \mu e. \tag{17}$$

Naturally, the perturbed CL conditions for problems with only lower bounds should be the following:

$$\begin{aligned} D(x)\,\nabla f(x) &= \mu e, & \text{(18a)} \\ x - l &\geq 0. & \text{(18b)} \end{aligned}$$

3.4 Perturbing the CL Conditions: General Bounds

Now we consider the more general problem (1) where both lower and upper bounds can be present. We will demonstrate the difficulties encountered when trying to similarly extend the perturbation to the general case. For simplicity, we will restrict our attention to the case in which all the bounds are finite.

Since the equalites in the CL conditions have the identical form $D(x)\nabla f(x) = 0$ for the cases of one-sided and general bounds, one would think that the same perturbation scheme (17) should work for both cases. Hence, the perturbed CL conditions for problems with both lower and upper bounds should be:

$$\begin{aligned} D(x)\,\nabla f(x) &= \mu e, & \text{(19a)} \\ x - l, u - x &\geq 0. & \text{(19b)} \end{aligned}$$

We now show, however, that the relationship between the perturbed KKT conditions and the perturbed CL conditions that exists in the case of one-sided bounds no longer exists in the case of general bounds. Moreover, as we will show in our numerical experiments, on a set of test problems the perturbed CL conditions (18) work well for the case of one-sided bounds, but (19) do not work well at all for the case of general bounds.

We begin by solving the second and third equations in the KKT system (3) for Δy and Δz in terms of Δx, respectively, obtaining

$$\begin{aligned} \Delta y &= -(X-L)^{-1}Y\Delta x - y, \\ \Delta z &= (U-X)^{-1}Z\Delta x + z. \end{aligned}$$

Substituting these expressions into the first equation, we obtain:

$$\left[\nabla^2 f(x) + [(X-L)^{-1}Y + (U-X)^{-1}Z]\right]\Delta x = -\nabla f(x). \tag{20}$$

Multiplying the CL system (6) by $D^{-1}(x)$ (assuming its existence at x), we obtain:

$$\left[\nabla^2 f(x) + D^{-1}(x)\,\mathrm{Diag}(|\nabla f(x)|)\right]\Delta x = -\nabla f(x). \tag{21}$$

Again, the two expressions differ only in the diagonal matrices added to $\nabla^2 f(x)$. In the case of one-sided bounds we were able to relate these two matrices by considering properties that hold at the solution. We were helped by the fact that each of those diagonal matrices was expressed as the product of two diagonal matrices. Notice that this is no longer the case for the diagonal matrix added to $\nabla^2 f(x)$ in (20) which is now a sum of two matrices.

We can similarly derive an equation analogous to (20) for the PKKT system:

$$\begin{aligned} &\left[\nabla^2 f(x) + (X-L)^{-1}Y + (U-X)^{-1}Z\right]\Delta x = \\ &\qquad -\nabla f(x) + \left[(X-L)^{-1} + (U-X)^{-1}\right]\mu e. \end{aligned}$$

Comparing the right-hand side of the above with that of equation (15), we see that the perturbation terms for the two systems are, respectively, $[(X-L)^{-1} + (U-X)^{-1}]\mu e$ and $D^{-1}(x)\mu e$. That is, in the two equations, the term $D^{-1}(x)$ corresponds to the term $[(X-L)^{-1} + (U-X)^{-1}]$.

In the previous derivation of the perturbation for the case of one-sided bounds, we showed that $D^{-1}(x)$ and $\mathrm{Diag}(\nabla f(x))J^d(x)$ could serve as approximations for $(X-L)^{-1}$ and Y, respectively. In the present case, however, it is not clear that $D^{-1}(x)$ can serve as an approximation to $(X-L)^{-1} + (U-X)^{-1}$. Furthermore, even if one insisted that $D^{-1}(x) \approx (X-L)^{-1} + (U-X)^{-1}$ and

$$D^{-1}(x)\mathrm{Diag}(|\nabla f(x)|) \approx (X-L)^{-1}Y + (U-X)^{-1}Z,$$

then one would be asking that

$$\text{Diag}(|\nabla f(x)|) \approx [(X-L)^{-1} + (U-X)^{-1}][(X-L)^{-1}Y + (U-X)^{-1}Z],$$

which does not seem plausible. Hence, there does not seem to be an analogous justification for the perturbation scheme (17) in the case of general bounds.

4 Interior-Point Newton Algorithms

We now describe four interior-point Newton algorithms, based on the KKT, PKKT, CL and PCL conditions (see (2), (7), (4), (19)), for solving optimization problems with simple bounds. Each algorithm starts with a strictly feasible initial iterate, takes a Newton step at each iteration, and damps the step whenever necessary to keep the iterates strictly feasible.

These algorithms are simple and not sophisticated. Indeed, they are not equipped with mechanisms for enhancing global convergence. Our goal, however, is to compare the behavior, particularly the robustness, of the damped Newton, interior-point method on the four sets of conditions – the perturbed and unperturbed KKT and CL conditions. For this purpose, we feel that simpler algorithms are more revealing.

Let α be the steplength assigned to the Newton step, calculated by multiplying the maximum allowable steplength, $\hat{\alpha}$, to the boundary of the feasible set by a damping parameter $\tau \in (0,1)$, that is, $\alpha = \tau\hat{\alpha}$. We also require α to be less than or equal to one. When multiplied by 100, the parameter τ, which is chosen at the start of the algorithm and remains constant throughout, represents the percentage of the steplength that would take us to the boundary. The larger the value of τ is, the more the iterates are allowed to approach the boundary.

Let $R(x,y,z)$ be the residual vector of the equalities in the KKT conditions (2) evaluated at (x,y,z). Let $\epsilon > 0$, $\epsilon_\alpha > 0$ and $\tau \in (0,1)$ be given. The damped, interior-point Newton algorithm for the KKT conditions (2) can be described as follows.

> **KKT Algorithm**: Given initial point (x,y,z) such that $l < x < u$, $y > 0$ and $z > 0$.

Step 1 If $\|R(x,y,z)\| < \epsilon$, stop, else go to Step 2.

Step 2 Solve the linear system (3) for $(\Delta x, \Delta y, \Delta z)$.

Step 3 Compute $\beta = \min((X-L)^{-1}\Delta x, (X-U)^{-1}\Delta x, Y^{-1}\Delta y, Z^{-1}\Delta z)$, where the minimum is taken over the components of all involved vectors, $\hat{\alpha} = -1/\min(\beta, -0.5)$ and $\alpha = \min(1, \tau\hat{\alpha})$.

Step 4 If $\alpha < \epsilon_\alpha$, stop, else go to Step 5.

Step 5 Let $(x,y,z) = (x,y,z) + \alpha(\Delta x, \Delta y, \Delta z)$ and go to Step 1.

Similarly, the damped, interior-point Newton algorithm for the CL conditions (4) can be described as follows.

CL Algorithm: Give x such that $l < x < u$.

Step 1 If $\|D(x)\,\nabla f(x)\| < \epsilon$, stop, else go to Step 2.

Step 2 Solve the linear system (6) for Δx.

Step 3 Compute $\beta = \min((X-L)^{-1}\Delta x, (X-U)^{-1}\Delta x)$, $\hat{\alpha} = -1/\min(\beta, -0.5)$ and $\alpha = \min(1, \tau\alpha)$.

Step 4 If $\alpha < \epsilon_\alpha$, stop, else go to Step 5.

Step 5 Let $x = x + \alpha\Delta x$ and go to Step 1.

We now present the damped, interior-point Newton algorithms for the PKKT (7) and PCL (19) conditions. In these algorithms the parameter, $\mu > 0$, must be updated at each iteration so that $\mu \to 0$ asymptotically as the iterates converge. Except for this change, the algorithms are essentially the same as those for the unperturbed conditions.

PKKT Algorithm: The same as the KKT Algorithm except at Step 2 the linear system (8) is solved with $\mu = \min(0.2, 20\nu) * \nu$, where $\nu = ((x-l)^T y + (u-x)^T z)/2n$.

PCL Algorithm: The same as the CL Algorithm except at Step 2 the linear system (16) is solved with $\mu = \min(0.2, 20\nu) * \nu$, where $\nu = d(x)^T|\nabla f(x)|/n$.

We note that the ν value in the PCL algorithm is nothing but the normalized 1-norm of the residual vector $D(x)\nabla f(x)$.

5 Numerical Experiments

The main purpose of our numerical experiments is to see how the KKT conditions and the CL conditions, both in perturbed and unperturbed formulations, compare as bases for constructing interior-point Newton algorithms. It is well-know that for the KKT conditions, the perturbed formulation possesses better global convergence properties than the unperturbed ones. For the CL conditions, it is worthwhile to determine whether or not the perturbed formulation is also preferable to the unperturbed ones.

In Section 5.1, we report numerical results obtained from applying each algorithm to a two-dimensional problem. In Section 5.2, we report numerical results for convex quadratic programs subject to either lower bounds only, or both lower and upper bounds.

We start each algorithm from the same set of starting points and report the percentage of runs corresponding to nonconvergence, which in all instances occurs when α becomes less than ϵ_α, i.e., the iterate was sticking to the boundary. For each algorithm, we also report the average number of iterations required to reach the convergence criterion for common successful runs; that is, the average is taken over the runs where all the algorithms successfully terminated upon the satisfaction of their corresponding convergence criterion.

We have implemented the four algorithms presented in the last section in Matlab. In all experiments, we chose $\tau = .85$, $\epsilon_\alpha = 10^{-5}$, and $\epsilon = 10^{-5}$.

5.1 A Two-Dimensional Problem

We tested the four algorithms on a simple two-dimensional problem: minimize $f(x, y) \equiv xy(x^2 - y^2)/(x^2 + y^2)$, subject to the box constraints $0.25 \leq x, y \leq 3.75$, because the results for this problem can be easily visualized and provide an indication of the regions of convergence. Given the box constraints, the minimum of this problem occurs at $(x^*, y^*) \approx (1.82, 3.75)$ on the boundary.

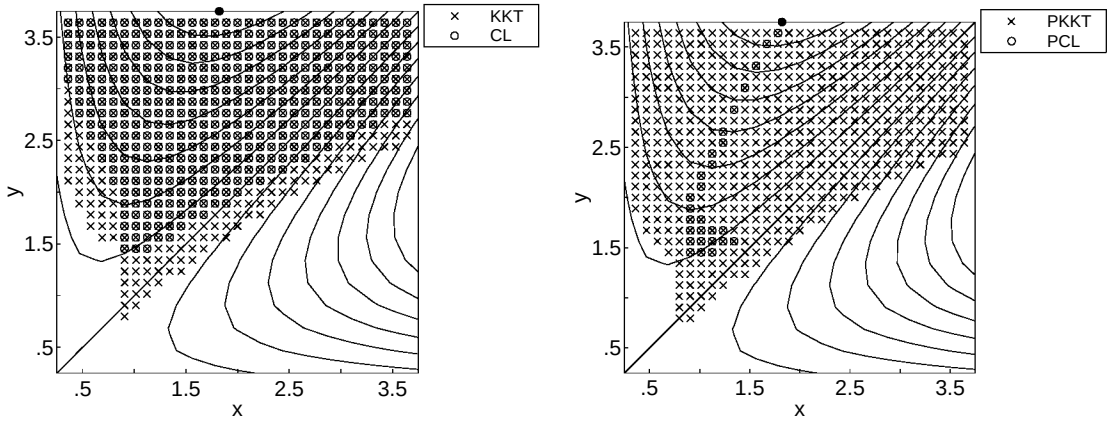

Figure 1: Approximate regions of convergence for the 2-dimensional problem.

The starting points are chosen to be the grid points on a 31 by 31 uniformly spaced mesh, totaling 961 points, as is shown in Figure 1. The positions of the "x" and "o" symbols indicate that the respective algorithms converged starting from those locations. Also plotted in Figure 1 are the contour lines of the function $f(x, y)$. The pictures in Figure 1 are self-explanatory. The most striking fact presented in the pictures is that the region of convergence for the PCL algorithm is extremely small for this problem.

In Table 1, we report the average number of iterations over common successful runs and the number of nonconvergence cases for each algorithm out of the 961 runs. We note that the number of common successful runs is small because of the large number (940) of nonconvergence cases encountered by the PCL algorithm. Clearly, for this problem with general bounds, the perturbed CL formulation did not offer any advantage over the unperturbed one. It fact, it was detrimental. On the other hand, in those a few cases where all the algorithms converged, the CL and PCL algorithms took one or two less iterations on average than their counterparts.

Table 1: Results on a 2-dimensional problem

Algorithm	Average Number of Iterations	Number of cases of Nonconvergence
KKT	8.4	412
CL	6.7	522
PKKT	7.8	398
PCL	6.5	940

5.2 Convex Quadratic Programs

In this subsection we report numerical results obtained from minimizing convex quadratic functions of the form:

$$q(x) = \frac{1}{2} x^T A^T A x - b^T x,$$

subject to either lower bounds only or both lower and upper bounds, where $A \in \Re^{n \times n}$ and $b \in \Re^n$ are randomly generated using the Matlab `rand` function. In our tests, we only used $n = 10$ and $n = 20$ because for problems of these sizes, we can already observe significantly different algorithmic behavior.

We again report the average number of iterations over successful runs and the percentage of nonconvergence. As mentioned before, from the definitions of the algorithms, nonconvergence occurs if and only only if $\alpha < \epsilon_\alpha$, i.e., the iterate is sticking to the boundary.

For $n = 10$, we generated 10 convex quadratic programs and a set of 200 starting points for each problem. For $n = 20$, we generated 5 convex quadratic programs and a set of 100 starting points for each problem.

The bounds l and u for test problems with both lower and upper bounds were generated as follows. For each objective function $q(x)$, we first solved the unconstrained minimization problem to find the location of the unique minimizer (the generated A matrices always turned out to be full rank). Then we selected the bounds in a rather ad hoc way manner except that we ensured that at least one bound, either upper or lower, was active. Mostly the bounds were chosen to be integers between -10 and 10.

For each problem with both lower and upper bounds, we generated another problem with only lower bounds by converting any active upper bound into a lower one and then deleting all upper bounds afterwards. Suppose that a problem has the solution x^* so that $x_i^* = u_i$, i.e., the i-th upper bound is active. The conversion is accomplished as follows. We first change the sign of the i-th column of A and of b_i to get $\hat{A}$ and $\hat{b}$. Then the new objective function,

$$\hat{q}(x) \equiv \frac{1}{2} x^T \hat{A}^T \hat{A} x - \hat{b}^T x,$$

has the unique minimizer $\hat{x}^* \equiv x^* - 2\, x_i^* e$ (i.e., the sign of x_i^* is changed), where e is the vector of all ones. As such, $\hat{x}_i^*$ satisfies the bounds $-u_i \leq \hat{x}_i \leq -l_i$, and the new lower bound is active. So we define $\hat{l_i} = -u_i$ to be the new lower bound. This procedure is obviously applicable to each active upper bound. Since there is no active upper bound left after the conversion, we discard all upper bounds and obtain a problem with only lower bounds.

The starting points for problems with both lower and upper bounds are generated as follows. We begin by randomly generating a vector $w \in \Re^n$ with components -1, 0, or 1. Then we define a point x_b as follows. For $i = 1, 2, \cdots, n$,

$$(x_b)_i = \begin{cases} l_i, & \text{if } w_i = -1, \\ (l_i + u_i)/2, & \text{if } w_i = 0, \\ u_i, & \text{if } w_i = 1. \end{cases}$$

Obviously, such a x_b lies on the boundary of the feasible set $\{x \in \Re^n : l \leq x \leq u\}$. We also compute the center of the feasible set: $x_c = (l + u)/2$. Then we define a starting point with respect to the random vector w and a parameter $\gamma \in (0, 1)$ to be

$$x_0 = x_c + \gamma(x_b - x_c).$$

The parameter γ determines the closeness of the starting point to the boundary. Clearly, the closer γ is to one, the closer it is to x_b on the boundary. We vary the closeness of x_0 by varying the values of γ. It is expected that the closer x_0 is to the boundary, the more likely the subsequent iterates will stick to the boundary.

For problems with only lower bounds, we make sure to use vectors w that have at least one component equal to -1 and repeat the above process, except that we replace u with $l + 5e$ in the computations involving u, where e is a vector of ones.

For the KKT and PKKT formulations, we must also choose the initial values for the multipliers y in the case of lower bounds only, and for y and z in the case of general bounds. In view of the first equation in the KKT conditions (2), we choose $y_0 = |\nabla f(x_0)|$ for the case of lower bounds only. We choose $y_0 = z_0 = |\nabla f(x_0)|$ for the case general bounds. These choices for the case of general bounds are somewhat arbitrary, but we did not want to use more deliberate choices in order to avoid giving undue advantages to the KKT and PKKT algorithms.

In next two subsections, we present our numerical results on convex quadratic programs in the form of bar charts, where the height of the bars in the left plot represents the percentage of failures out of all starting points for each algorithm, and the height of the bars in the right plot represents the average number of iterations over the successful runs in which all algorithms converged. In all the bar charts, the horizontal axis represents the values of the parameter γ, indicating how close the starting points are to the boundary.

5.2.1 Numerical Results: Lower Bounds

Figures 2 and 3 summarize the numerical results on problems with only lower bounds and of dimensions $n = 10$ and $n = 20$, respectively. Recall that we ran each algorithm from a total of 2000 starting points for $n = 10$ and 500 starting points for $n = 20$.

The left bar charts in Figures 2 and 3 indicate that as the starting points moved closer to the boundary of the feasible set (i.e., as γ approached 1), the iterates of the CL and KKT algorithms stuck to the boundary more frequently. Moreover, the iterates of the CL algorithm stuck more frequently than those of the KKT algorithm for each γ value tested.

Overall, the perturbations worked well in preventing the iterates from sticking to the boundary, with the exception that when starting points were too close to the boundary ($\gamma = 0.999$), the PCL algorithm failed to converge 7 times out of 2000 for $n = 10$, and 10 times out of 500 for $n = 20$. On the other hand, the PKKT algorithm never failed once to converge.

We also note from Figures 2 and 3 that for each γ nonconvergence occured more frequently when $n = 20$ than when $n = 10$. In addition, in terms of the

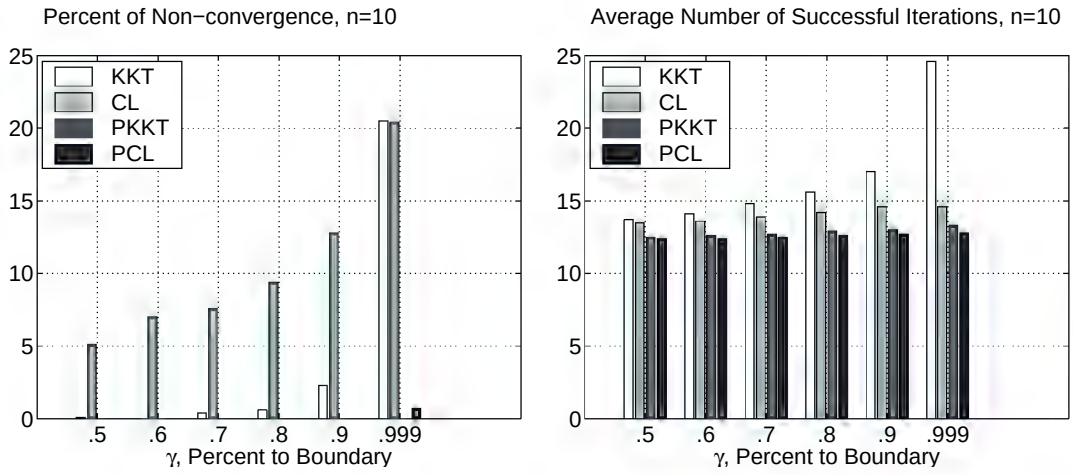

Figure 2: Results for Convex Quadratic Programs: Lower bounds, $n = 10$.

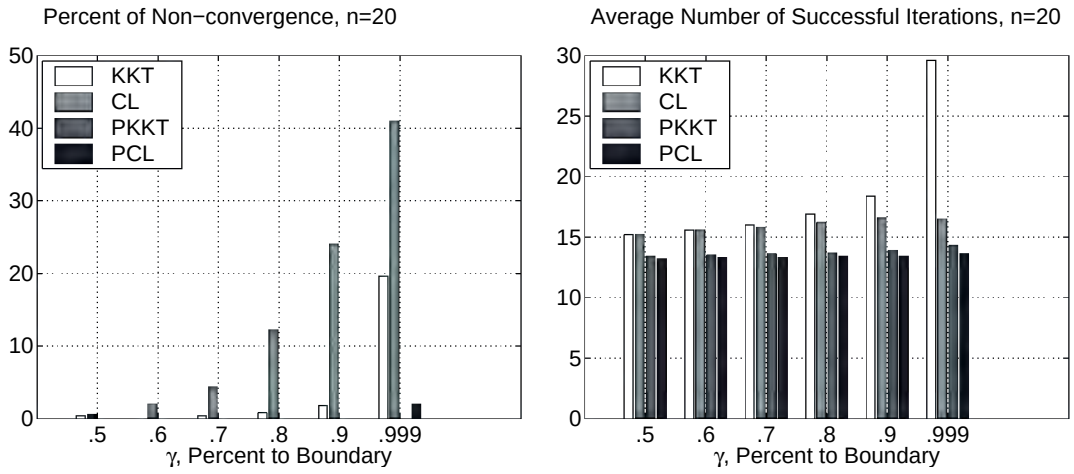

Figure 3: Results for Convex Quadratic Programs: Lower bounds, $n = 20$.

average number of iterations over the successful runs, the CL, PKKT, and PCL algorithms are comparable while the KKT algorithms required more iterations when the starting points are close to the boundary.

5.2.2 Numerical Results: General Bounds

For problems with both lower and upper bounds, the performance of the PCL algorithm is much worse than that of the CL algorithm, as can be seen from Figure 4. The failure rate is over 80% even when the starting points are only 50% of the way to the boundary of the feasible set (i.e., $\gamma = 0.5$). This fact indicates that the perturbation to the CL conditions is detrimental when applied to problems with both upper and lower bounds. However, on the small number of staring points from which the PCL algorithm did converge, it took less iterations on average than the PKKT algorithm.

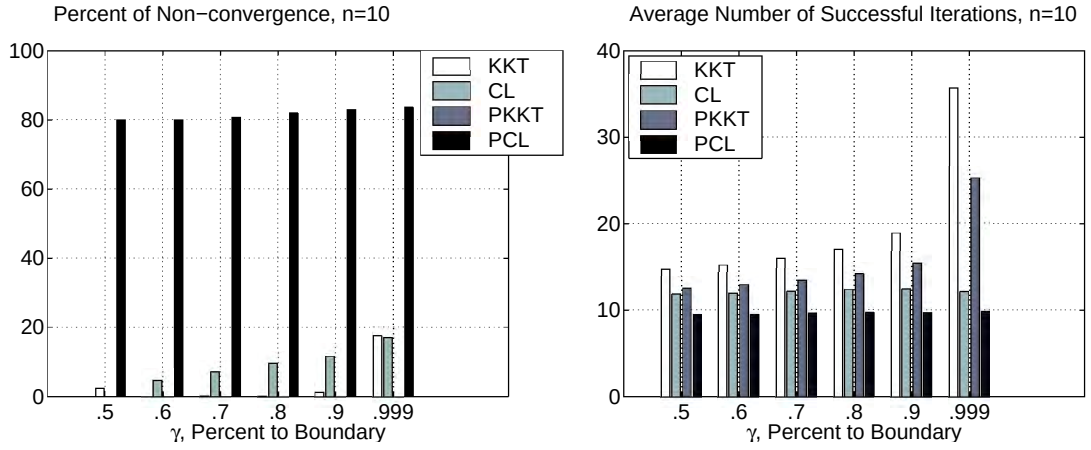

Figure 4: Results for Convex Quadratic Programs: General bounds, $n = 10$.

Since the PCL algorithm was clearly not as robust as the other algorithms on problems with both lower and upper bounds, we did not include it in the comparison for $n = 20$ in Figure 5.

In Figures 4 and 5 we again see that as the starting points moved closer to the boundary of the feasible set (i.e., as $\gamma \to 1$), the iterates of the CL and KKT algorithms stuck to the boundary more frequently, with the iterates of the CL

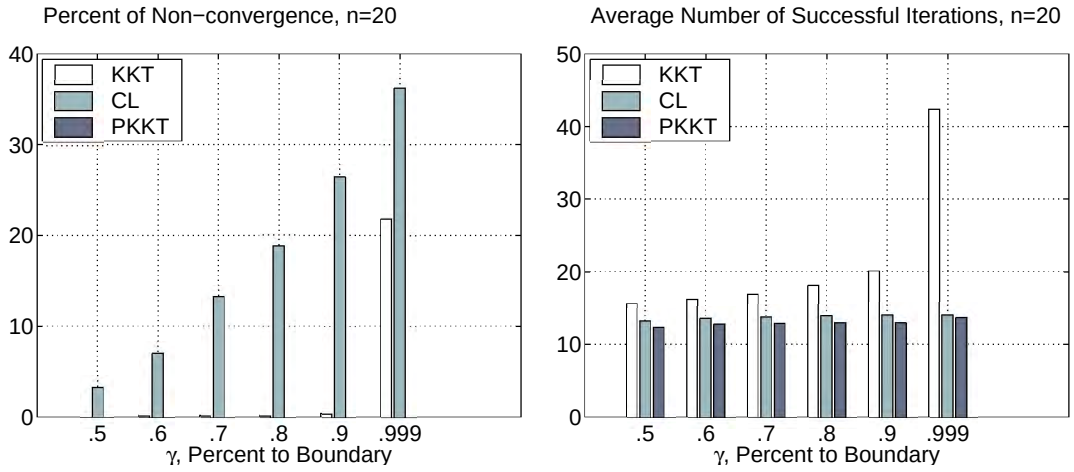

Figure 5: Results for Convex Quadratic Programs: General bounds, $n = 20$.

algorithm sticking more frequently than those of the KKT algorithm.

In addition, when problem size was increased from $n = 10$ to $n = 20$, the percentage of nonconvergence cases also increased for each γ value considered. Once again, the PKKT algorithm converged for all γ values considered.

6 Concluding Remarks

It is not surprising that without the perturbations interior-point Newton algorithms based on both the KKT and the CL conditions tend to stick to the boundary when they are started close to the boundary. It appears, though, that the KKT conditions may be slightly less prone to this difficulty. For problems with only one-sided bounds, the perturbation to the CL conditions enhances global convergence, as in the case of the KKT conditions. However, for problems with general bounds, the same perturbation scheme does not work. On the sets of test problems used in this paper, the PKKT algorithm is the most robust of the interior-point Newton algorithms.

References

[1] VILLALOBOS, C., TAPIA, R. and ZHANG, Y. Behavior of the Newton method on two equivalent systems from linear programming. To appear in *Journal of Optimization Theory and Applications*, 2002.

[2] VILLALOBOS, C., TAPIA, R. and ZHANG, Y. The sphere of convergence of Newton's method on two equivalent systems from nonlinear programming. Technical report TR99-15, Department of Computational and Applied Mathematics, Rice University, May 1999.

[3] VICENTE, L. An analysis of Newton's method for equivalent Karush-Kuhn-Tucker systems. *Investigación Operativa*, 7:17–28, 1999.

[4] VICENTE, L. Local convergence of the affine scaling interior-point algorithm for nonlinear programming. *Computational Optimization and Applications*, 17:23-35, 2000.

[5] COLEMAN, T. and LI, Y. A reflective Newton method for minimizing a quadratic function subject to bounds on the variables. SIAM Journal on Optimization 6 (1996) pp. 418-445.

[6] KARUSH, W. Minima of functions of several variables with inequalities as side conditions. Master's thesis, Department of Mathematics, University of Chicago, 1939.

[7] KUHN, H. W. and TUCKER, A. W. Non-linear programming. In J. Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–493, Berkley, California, 1951. University of California Press.

[8] NASH, S. G. and A. SOFER, A. *Linear and Nonlinear Programming.* McGraw-Hill Companies, Inc., New York, New York, 1996.

[9] COLEMAN, T. and LI, Y. On the convergence of interior-reflective Newton methods for nonlinear minimization subject to bounds. *Mathematical Programming*, 67:189–224, 1994.

[10] COLEMAN, T. and LI, Y. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal of Optimization*, 6:418–445, 1996.

[11] TAPIA, R. On the role of slack variables in quasi-Newton methods for constrained optimization. In L.C. W. Dixon and G. P. Szego, editors, *Numerical Optimization of Dynamic Systems*. North-Holland, Amsterdam, Holland, 1980.

[12] TAPIA, R. On the fundamental role of interior-point methodology in constrained optimization. Technical report TR97-09, Department of Computational and Applied Math, Rice University, April 1997.

[13] FIACCO, A. V. and MCCORMICK, G. P. *Nonlinear Programming, Sequential Unconstrained Minimization Techniques*. John Wiley and Sons, New York, New York, 1968. Reprinted by SIAM Publications, 1990.

[14] VICENTE, L. On interior-point Newton algorithms for discretized optimal control problems with state constraints. Manuscript, Departamento de Matemática, Universidade de Coimbra, 3000 Coimbra, Portugal. 2000.