A globally and superlinearly convergent primal-dual interior point trust region method for large scale constrained optimization

Hiroshi Yamashita[*], Hiroshi Yabe[†] and Takahito Tanabe[‡]

July, 1997 (revised July, 1998)

Abstract

This paper proposes a primal-dual interior point method for solving large scale nonlinearly constrained optimization problems. To solve large scale problems, we use a trust region method that uses second derivatives of functions for minimizing the barrier-penalty function instead of line search strategies. Global convergence of the proposed method is proved under suitable assumptions. By carefully controlling parameters in the algorithm, superlinear convergence of the iteration is also proved. A nonmonotone strategy is adopted to avoid the Maratos effect as in the nonmonotone SQP method by Yamashita and Yabe. The method is implemented and tested with a variety of problems given by Hock and Schittkowski's book and by CUTE. The results of our numerical experiment show that the given method is efficient for solving large scale nonlinearly constrained optimization problems.

1 Introduction

This paper deals with the following constrained optimization problem:

$$\text{(1.1)} \qquad \begin{array}{ll} \text{minimize} & f(x), \qquad x \in \mathbf{R}^n, \\ \text{subject to} & g(x) = 0, \ \ x \geq 0, \end{array}$$

where we assume that the functions $f : \mathbf{R}^n \rightarrow \mathbf{R}^1$ and $g : \mathbf{R}^n \rightarrow \mathbf{R}^m$ are sufficiently smooth and the number of variable n and the number of equality constraints m may be large.

There are several well known methods for solving the above problem. Well known examples are the augmented Lagrangian method (see, for example [1] and [15]) and the SQP method (see [15]). Recently, variants of classic interior point methods [13] are revived

[*]Mathematical Systems, Inc., 2-4-3, Shinjuku, Shinjuku-ku, Tokyo, Japan. hy@msi.co.jp

[†]Department of Applied Mathematics, Faculty of Science, Science University of Tokyo, 1-3, Kagurazaka, Shinjuku-ku, Tokyo, Japan. yabe@am.kagu.sut.ac.jp

[‡]Mathematical Systems, Inc., 2-4-3, Shinjuku, Shinjuku-ku, Tokyo, Japan. tanabe@msi.co.jp

[25, 4, 12, 6, 7] partly because of the phenomenal success of interior point methods for linear programming problems. In this paper we extend the algorithm developed in [25] to solve large scale nonlinear optimization problems. In [25], the primal-dual framework, minimization of the barrier-penalty function and suitable line search strategy are combined to give a globally convergent efficient algorithm for large scale linear programming and small to medium scale nonlinear programming. The local behavior of this method is analyzed in [28, 24]. However, to solve large scale nonlinear problems, one reasonable way is to resort to the trust region strategy instead of the line search strategy because of the reason explained below. Trust region methods used in the interior point method are studied in [3], [5], [8] and [10].

Let the Lagrangian function of the above problem be defined by

$$L(w) = f(x) - y^t g(x) - z^t x, \tag{1.2}$$

where $w = (x, y, z)^t$, and $y \in \mathbf{R}^m$ and $z \in \mathbf{R}^n$ are the Lagrange multiplier vectors which correspond to the equality and inequality constraints respectively. Then Karush-Kuhn-Tucker (KKT) conditions for optimality of the above problem are given by

$$r_0(w) = \begin{pmatrix} \nabla_x L(w) \\ g(x) \\ XZe \end{pmatrix} = 0, \quad x \geq 0, \; z \geq 0, \tag{1.3}$$

where

$$\begin{aligned} \nabla_x L(w) &= \nabla f(x) - A(x)^t y - z, \\ A(x) &= \begin{pmatrix} \nabla g_1(x)^t \\ \vdots \\ \nabla g_m(x)^t \end{pmatrix}, \\ X &= \operatorname{diag}(x_1, \cdots, x_n), \\ Z &= \operatorname{diag}(z_1, \cdots, z_n), \\ e &= (1, \cdots, 1)^t \in \mathbf{R}^n. \end{aligned}$$

To solve problem (1.1) by an interior point method, we define the following minimization problem for the barrier function [13]:

$$\begin{array}{ll} \text{minimize} & f(x) - \mu \sum_{i=1}^{n} \log(x_i), \quad x \in \mathbf{R}^n_+ \\ \text{subject to} & g(x) = 0, \end{array} \tag{1.4}$$

where $\mu > 0$ is a given constant and $\mathbf{R}^n_+ = \{x \in \mathbf{R}^n \mid x > 0\}$. It is well known that under appropriate assumptions, a solution of the above problem gives a good approximation to a solution of the original problem (1.1) for sufficiently small μ. The optimality conditions of this problem are given by

$$r(w, \mu) = \begin{pmatrix} \nabla_x L(w) \\ g(x) \\ XZe - \mu e \end{pmatrix} = 0, \quad x > 0, \quad z > 0, \tag{1.5}$$

where $y \in \mathbf{R}^m$ is the Lagrange multiplier for the equality constraints and $z \in \mathbf{R}^n$ is introduced to satisfy the third set of equations. In this paper we call conditions (1.5) the barrier KKT conditions, and a point $w(\mu) = (x(\mu), y(\mu), z(\mu))$ that satisfies these conditions is called the barrier KKT point. We note that conditions (1.5) are often called the centrality conditions, and a point $w(\mu)$ that satisfies these conditions is called the center that corresponds to μ in many literatures.

Further, we define the barrier-penalty function which is introduced in [25] by

$$F(x,\mu) = f(x) - \mu \sum_{i=1}^{n} \log(x_i) + \rho \sum_{i=1}^{m} |g_i(x)|, \tag{1.6}$$

for $\mu > 0$ and $\rho > 0$. If ρ is sufficiently large to satisfy $\rho \geq \|y\|_\infty$, then it is easy to show that the optimality condition of the problem

$$\text{minimize} \quad F(x,\mu), \quad x \in \mathbf{R}^n_+ \tag{1.7}$$

coincides with conditions (1.5) (see [25]).

In the following, we consider an interior point method that solves optimality conditions (1.5) with a strictly decreasing sequence $\{\mu_k\}$, $\mu_k \downarrow 0$. Therefore we will assume that the variables x and z always have positive values. Let $\Delta w = (\Delta x, \Delta y, \Delta z)^t$ be defined by a solution of

$$J(w)\Delta w = -r(w,\mu), \tag{1.8}$$

where

$$J(w) = \begin{pmatrix} G & -A(x)^t & -I \\ A(x) & 0 & 0 \\ Z & 0 & X \end{pmatrix}. \tag{1.9}$$

If $G = \nabla_x^2 L(w)$, then Δw becomes Newton's direction for solving (1.5). Unless otherwise stated, G is supposed to be $\nabla_x^2 L(w)$ in the following.

In [25], the above iteration vectors are used to give a globally convergent algorithm that uses the Armijo's rule for reducing the barrier-penalty function when we can assume that the matrix G is positive semi-definite. As examples, we can list linear programs, positive semi-definite quadratic programs and small to medium scale general nonlinear programs. The last class of problems can be included because we can use dense positive definite quasi-Newton approximations to the matrix $\nabla_x^2 L(w)$ in this case. Our interest in this paper is in solving large scale nonlinear programs. In this case, we can no longer use dense positive definite approximations to the Hessian of the Lagrangian. Therefore we use the Hessian of the Lagrangian itself which may not be a positive semi-definite matrix, and employ a trust region method for minimization of the barrier-penalty function. A preliminary version [26] of this paper describes this algorithm and proves its global convergence.

In this paper, we also aim to obtain superlinear convergence of our basic interior point method. It will be shown that by carefully controlling the values of relevant parameters, we can have superlinear convergence of the iterates if we ignore the occurrence of Maratos effect, which may be caused by the use of the l_1-exact penalty function in (1.6). To avoid Maratos effect, we adopt nonmonotone strategy of the iterations which is proposed in [27] for SQP method.

Our method is implemented and tested with a variety of test problems. Test problems from Hock and Schittkowski's book [17] and CUTE [2] are used for our experiment. The results in Section 6 show that the proposed method is very efficient for solving large scale nonlinear problems as well as small ones.

This paper is organized as follows. In Section 2, the basic trust region iteration for finding a barrier KKT point with a fixed barrier parameter is described, and its global convergence is proved. In Section 3, we propose a new method and show its global convergence. In Section 4, superlinear convergence of our method is proved. Section 5 describes a practical way of choosing the trust region step. Section 6 reports our numerical experiment.

In what follows, the subscript k denotes an iteration count. Subscripts i and j denote components of vectors and matrices. For simplicity of description, we assume $\|\cdot\|$ denotes the l_2 norm for vectors and matrices in this paper. Practical but legitimate choices of actual norms will be described in Section 6. Order notations are used in this paper. Let $\{a_k\}$ and $\{b_k\}$ are nonnegative sequences. If there exists a positive constant ξ such that $a_k \leq \xi b_k$, then we write $a_k = O(b_k)$. If there exists a positive sequence $\{\xi_k\}, \xi_k \downarrow 0$ such that $a_k \leq \xi_k b_k$, then we write $a_k = o(b_k)$.

2 Trust region method with fixed barrier parameter

2.1 Algorithm

A first order approximation $F_l(x; s) : \mathbf{R}^n_+ \to \mathbf{R}^1$ to the barrier-penalty function with respect to $s \in \mathbf{R}^n$ at a point $x \in \mathbf{R}^n_+$ is defined by

$$(2.1)\quad F_l(x;s) = F(x,\mu) + (\nabla f(x) - \mu X^{-1}e)^t s + \rho \sum_{i=1}^{m} \left(\left| g_i(x) + \nabla g_i(x)^t s \right| - |g_i(x)| \right).$$

Similarly, we define a second order approximation $F_q(x; s) : \mathbf{R}^n_+ \to \mathbf{R}^1$ to the barrier-penalty function by

$$F_q(x;s) = F_l(x;s) + \frac{1}{2} s^t Q s,$$

where an explicit form of the matrix $Q \in \mathbf{R}^{n \times n}$ will be given in Section 2.3. Define changes of these functions which correspond to the step s by

$$\begin{aligned} \Delta F_l(x;s) &\equiv F_l(x;s) - F_l(x;0) = F_l(x;s) - F(x,\mu), \\ \Delta F_q(x;s) &\equiv F_q(x;s) - F_q(x;0) = F_q(x;s) - F(x,\mu), \\ \Delta F(x;s) &\equiv F(x+s,\mu) - F(x,\mu). \end{aligned}$$

Now we define a reference direction that will be used to form the actual step with Newton's direction, and to obtain the global convergence property of the algorithm by

$$(2.2)\qquad \begin{pmatrix} D & -A(x)^t & -I \\ A(x) & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x_{SD} \\ \Delta y_{SD} \\ \Delta z_{SD} \end{pmatrix} = -r(w,\mu),$$

where D is a positive definite matrix. We call the direction $\Delta w_{SD} = (\Delta x_{SD}, \Delta y_{SD}, \Delta z_{SD})^t$ the steepest descent direction by an analogy with the case in unconstrained optimization.

Lemma 1 *There holds*

$$\Delta F_l(x;\Delta x) \leq -\Delta x^t(G + X^{-1}Z)\Delta x - (\rho - \|y+\Delta y\|_\infty)\sum_{i=1}^m |g_i(x)|\,. \tag{2.3}$$

If $\rho \geq \|y+\Delta y\|_\infty$ *and* G *is positive semi-definite, then* $\Delta F_l(x;\Delta x) \leq 0$, *and* $\Delta F_l(x;\Delta x) = 0$ *yields* $\Delta x = 0$.

Proof. From (1.8) and (2.1) we have

$$\begin{aligned}\Delta F_l(x;\Delta x) &= -\Delta x^t(G+X^{-1}Z)\Delta x + \Delta x^t A(x)\,(y+\Delta y)\\ &\quad +\rho\sum_{i=1}^m \left|g_i(x) + \nabla g_i(x)^t\Delta x\right| - \rho\sum_{i=1}^m |g_i(x)|\\ &= -\Delta x^t(G+X^{-1}Z)\Delta x - (y+\Delta y)^t\,g(x) - \rho\sum_{i=1}^m |g_i(x)|.\end{aligned}$$

This equality gives the desired result (2.3).

A proof of the second statement is easy because two terms in (2.3) are nonpositive by the assumption. □

If G is replaced by D in Lemma 1, then we have

$$\Delta F_l(x;\Delta x_{SD}) \leq -\Delta x_{SD}^t(D+X^{-1}Z)\Delta x_{SD} - (\rho - \|y+\Delta y_{SD}\|_\infty)\sum_{i=1}^m |g_i(x)|\,. \tag{2.4}$$

Now we describe a trust region algorithm that finds a barrier KKT point for a fixed barrier parameter μ. At the iteration k, we are given the trust region radius $\delta_k > 0$ and the vectors Δw_k and Δw_{SDk}. From these two vectors we form the step s_k that satisfies the trust region constraint $\|s_k\| \leq \delta_k$ and strict positivity conditions of the variables. For the latter to be maintained, we force the next trial point to satisfy

$$(1-\gamma)(x_k)_i \leq (x_k+s_k)_i, \;\; i = 1,...,n,$$

where $\gamma \in (0,1)$. Note that by the existence of this condition, the trust region radius need not be unnecessarily small to satisfy positivity conditions. The step s_k should also satisfy

$$\Delta F_q(x_k;s_k) \leq \frac{1}{2}\Delta F_q(x_k;\alpha^*(x_k,\Delta x_{SDk})\Delta x_{SDk}), \tag{2.5}$$

where $\alpha^*(x,d)$ is defined by

$$\alpha^*(x,d) = \arg\min\left\{F_q(x;\alpha d)\;\middle|\;\alpha \leq 1,\;\|\alpha d\| \leq \delta,\;\alpha\in[0,\gamma\bar{\alpha}(x,d)]\right\} \tag{2.6}$$

and

$$\bar{\alpha}(x,d) = \min_i\left\{-\frac{x_i}{d_i}\middle|\, d_i < 0\right\}$$

for $x \in \mathbf{R}^n_+$, $d \in \mathbf{R}^n$. The step size $\bar{\alpha}(x,d)$ gives a step to the boundary composed of the bounds on the variables along the direction d. Thus the step size $\alpha^*(x,d)$ gives a minimum point of the function F_q along the direction d in the interval defined by the trust region radius δ and the feasible step size $\gamma\bar{\alpha}(x,d)$. Therefore condition (2.5) gives a sufficient decrease condition based on the steepest descent step.

Now we present the algorithm of a trust region method for finding a barrier KKT point.

Algorithm TR

Step 0. Select an initial point $w_0 \in \mathbf{R}^n_+ \times \mathbf{R}^m \times \mathbf{R}^n_+$ and positive parameters μ and ρ . Set parameters $\varepsilon > 0$, $\gamma \in (0,1)$, $\delta_0 > 0$ and set $k = 0$.

Step 1. If $\|r(w_k,\mu)\| \leq \varepsilon$, then stop.

Step 2. Calculate the vectors Δw_k and Δw_{SDk} that satisfy (1.8) and (2.2) respectively. If $G_k = \nabla^2_x L(w_k)$ gives a too large vector that does not satisfy (2.10) given below, G_k is modified to satisfy (2.10).

Step 3. Find the direction $s_k \in \mathbf{R}^n$ that satisfies the conditions:

$$\begin{aligned} \|s_k\| &\leq \delta_k, \\ (1-\gamma)(x_k)_i &\leq (x_k + s_k)_i, \;\; i = 1, ..., n, \\ \Delta F_q(x_k; s_k) &\leq \frac{1}{2}\Delta F_q(x_k; \alpha^*(x_k, \Delta x_{SDk})\Delta x_{SDk}). \end{aligned} \tag{2.7}$$

Step 4. δ_{k+1} is defined as follows:

$$\begin{aligned} &\text{If } \Delta F(x_k; s_k) > \frac{1}{4}\Delta F_q(x_k; s_k), \text{ then} && \delta_{k+1} = \frac{1}{2}\delta_k; \\ &\text{If } \Delta F(x_k; s_k) \leq \frac{3}{4}\Delta F_q(x_k; s_k), \text{ then} && \delta_{k+1} = 2\delta_k; \\ &\qquad\qquad\qquad\qquad\qquad \text{Otherwise} && \delta_{k+1} = \delta_k. \end{aligned}$$

Step 5. If $\Delta F(x_k; s_k) \leq 0$, then set $x_{k+1} = x_k + s_k$, compute α_{yk} and α_{zk}, set $y_{k+1} = y_k + \alpha_{yk}\Delta y_k$ and $z_{k+1} = z_k + \alpha_{zk}\Delta z_k$. Otherwise set $w_{k+1} = w_k$.

Step 6. Set $k := k+1$ and go to Step 1. □

In the above algorithm, step sizes for the variables y and z are determined according to the rule proposed in [25]. For the variable z, we prevent z from becoming too small or diverging to infinity by setting the condition

$$(c_{Lk})_i \leq (x_k + s_k)_i (z_k + \alpha_{zk}\Delta z_k)_i \leq (c_{Uk})_i, \quad i = 1, \cdots, n,$$

at the end of each iteration, where the bounds c_{Lk} and c_{Uk} satisfy

$$0 < (c_{Lk})_i < \mu < (c_{Uk})_i, \quad i = 1, \cdots, n.$$

To this end, we let

$$(c_{Lk})_i = \min\left\{\frac{\mu}{M_L}, (x_k+s_k)_i(z_k)_i\right\}, \quad (c_{Uk})_i = \max\left\{M_U\mu, (x_k+s_k)_i(z_k)_i\right\}, \quad i=1,\cdots,n \tag{2.8}$$

where $M_L > 1$ and $M_U > 1$ are given constants. The construction of the above bounds shows that current z satisfies

$$\frac{(c_{Lk})_i}{(x_k+s_k)_i} \le (z_k)_i \le \frac{(c_{Uk})_i}{(x_k+s_k)_i}, \quad i=1,\cdots,n.$$

Thus α_{zk} is determined by

$$\alpha_{zk} = \min\left\{\min_i\left\{\max_{\alpha_i}\left\{\alpha_i \,\middle|\, \frac{(c_{Lk})_i}{(x_k+s_k)_i} \le (z_k+\alpha_i\Delta z_k)_i \le \frac{(c_{Uk})_i}{(x_k+s_k)_i}\right\}\right\}, 1\right\}. \tag{2.9}$$

This rule means that the step size α_{zk} is the maximal allowed step that satisfies the box constraints with the restriction of being not greater than the unit step length. In actual calculation we modify the direction Δz_k by

$$(\Delta z_k')_i = \begin{cases} 0, & \text{if } (z_k)_i = (c_{Lk})_i/(x_k+s_k)_i \text{ and } (\Delta z_k)_i < 0, \\ 0, & \text{if } (z_k)_i = (c_{Uk})_i/(x_k+s_k)_i \text{ and } (\Delta z_k)_i > 0, \\ (\Delta z_k)_i, & \text{otherwise.} \end{cases}$$

This modification means that we project the direction along the boundary of the box constraints if the point z_k is on that boundary and the direction Δz_k points outward of the box. This procedure is adopted because it gives better numerical results. The global convergence result shown in the following is equally valid for both unmodified and modified directions.

Lemma 2 *Suppose that an infinite sequence $\{w_k\}$ is generated by Algorithm TR for fixed $\mu > 0$. Then if $\liminf_{k\to\infty}(x_k)_i > 0$ and $\limsup_{k\to\infty}(x_k)_i < \infty$, then $\liminf_{k\to\infty}(c_{Lk})_i > 0$ and $\limsup_{k\to\infty}(c_{Uk})_i < \infty$ for $i=1,\cdots,n$.*

Proof. Suppose that $(c_{Lk})_i \to 0$ for an i and some subsequence $K \subset \{0,1,2,\cdots\}$. Then by the definition of $(c_{Lk})_i$ in (2.8), $(z_k)_i \to 0, k \in K$.

However, in order for a subsequence of $\{(z_k)_i\}$ to tend to 0, there must be an iteration k at which the lower bound $(c_{Lk})_i/(x_{k+1})_i$ of $(z_k)_i$ is arbitrary small and the value of $(z_k)_i$ at the iteration is strictly larger than that bound, i.e. at the iteration the value of $(z_k)_i$ decreases to a strictly smaller value. This means that at the iteration k, $(c_{Lk})_i = \mu/M_L$ and therefore the value of $(x_{k+1})_i$ must be arbitrary large. This is impossible because of the assumption of the lemma. The proof of the boundedness of $(c_{Uk})_i$ is similar. □

For the variable y, we set

$$\alpha_{yk} = \alpha_{zk}.$$

We note that the step $\alpha_y = 1$ is also a valid step that gives a global convergence result.

2.2 Global convergence

Before proving the global convergence of Algorithm TR, we list the necessary assumptions.

Assumption G

(G1) The functions f and $g_i, i = 1, ..., m$, are twice continuously differentiable.

(G2) The level set of the barrier penalty function at an initial point $x_0 \in \mathbf{R}^n_+$, which is defined by $\left\{x \in \mathbf{R}^n_+ \,|F(x, \mu) \le F(x_0, \mu)\right\}$, is compact for given $\mu > 0$.

(G3) The matrix $A(x)$ is of full rank on the level set defined in (G2).

(G4) The matrix D is uniformly positive definite and uniformly bounded. The matrices Q and G are uniformly bounded.

(G5) There exists a number $M > 0$ such that

$$\|\Delta x_k\| \le M \|\Delta x_{SDk}\|, \quad \|s_k\| \le M \|\Delta x_{SDk}\|, \tag{2.10}$$

for each $k = 0, 1, \cdots$.

(G6) The penalty parameter ρ satisfies $\rho \ge \|y_k + \Delta y_{SDk}\|_\infty$ for each $k = 0, 1, ...$. □

It follows from Assumption G that the linear system of equations (2.2) has a unique solution and that the direction Δx_{SDk} is uniformly bounded on the compact level set defined in (G2). The following lemma shows the basic property of the search directions.

Lemma 3 *(1) If $\Delta w_k = 0$ or $\Delta w_{SDk} = 0$ at an interior point w_k, then the point w_k satisfies the barrier KKT conditions.*
(2) If $\Delta x_k = 0$, then $\Delta x_{SDk} = 0$.
(3) If $\Delta x_{SDk} = 0$, then $\Delta x_k = 0$ and $s_k = 0$.
(4) If $\Delta x_k = 0$, then $\alpha_{zk} = 1$ and $\alpha_{yk} = 1$ are adopted in Algorithm TR, and the point w_{k+1} satisfies the barrier KKT conditions.

Proof. (1) It is clear from (1.8) and (2.2).
(2) Since $(0, \Delta y_k, \Delta z_k)^t$ satisfies (2.2) and the coefficient matrix of (2.2) is nonsingular, the uniqueness of the solution to (2.2) implies $\Delta x_{SDk} = 0$.
(3) This is a direct result from (G5).
(4) We note from (2) and (3) that $\Delta x_k = 0$ yields $s_k = 0$. Thus by (1.8) we have

$$(x_k + s_k)_i(z_k + \Delta z_k)_i = (x_k)_i(z_k + \Delta z_k)_i = \mu.$$

This implies that the stepsize $\alpha_{zk} = 1$ is accepted, and so is $\alpha_{yk} = 1$. Then it follows from (1.8) again that $w_{k+1} = (x_k, y_k + \Delta y_k, z_k + \Delta z_k)$ satisfies the barrier KKT conditions. Therefore the lemma is proved. □

Now we proceed to the analysis of global convergence property of the above algorithm. From the above lemma, we observe that if $\Delta x_{SDk} = 0$ at some iteration k, then the next point w_{k+1} is the barrier KKT point. Therefore we will assume that $\Delta x_{SDk} \ne 0$ for each $k = 0, 1, \cdots$ in the following.

We state the following simple lemma first.

Lemma 4 *If a vector $d \in \mathbf{R}^n$ satisfies*

$$g(x) + A(x)d = 0,$$

then there holds the relation

$$\Delta F_l(x; \alpha d) = \alpha \Delta F_l(x; d), \quad \alpha \in [0, 1]. \tag{2.11}$$

Proof. Since $g_i(x) + \nabla g_i(x)^t d = 0$ for all i, by (2.1) we have

$$\begin{aligned} \Delta F_l(x; \alpha d) &= \alpha(\nabla f(x) - \mu X^{-1} e)^t d + \rho \sum_{i=1}^{m} ((1-\alpha)|g_i(x)| - |g_i(x)|) \\ &= \alpha \left[(\nabla f(x) - \mu X^{-1} e)^t d + \rho \sum_{i=1}^{m} \left(\left| g_i(x) + \nabla g_i(x)^t d \right| - |g_i(x)| \right) \right]. \end{aligned}$$

Thus the proof is complete. □

Lemma 5 *Let $x \in \mathbf{R}^n_+, 0 \neq d \in \mathbf{R}^n$ and $\delta > 0$ be given. Assume that $\Delta F_l(x; d) < 0$, and that*

$$g(x) + A(x)d = 0.$$

Then the step size defined by (2.6) can be expressed as

$$\alpha^*(x, d) = \min \left\{ 1, \frac{\delta}{\|d\|}, \gamma \bar{\alpha}(x, d), -\frac{\Delta F_l(x; d)}{\max\{d^t Q d, 0\}} \right\}, \tag{2.12}$$

where the last term in the braces in the right hand side is assumed to give the value ∞ if the value of the denominator is 0. Further we have

$$\Delta F_q(x; \alpha^*(x, d)d) \leq \frac{1}{2} \alpha^*(x, d) \Delta F_l(x; d). \tag{2.13}$$

Proof. By the definition of the function F_q and Lemma 4, we have

$$F_q(x; \alpha d) = F(x, \mu) + \alpha \Delta F_l(x; d) + \frac{1}{2} \alpha^2 d^t Q d, \quad \alpha \in [0, 1]. \tag{2.14}$$

Suppose that $d^t Q d > 0$ for the moment. Then the unconstrained minimum $\hat{\alpha}$ of the function in the right hand side of the above equality is calculated by

$$\hat{\alpha} = -\frac{\Delta F_l(x; d)}{d^t Q d}.$$

Therefore we obtain

$$\alpha^*(x, d) = \min \left\{ 1, \frac{\delta}{\|d\|}, \gamma \bar{\alpha}(x, d), -\frac{\Delta F_l(x; d)}{d^t Q d} \right\}, \tag{2.15}$$

in this case. From this relation we have

$$d^t Q d \leq -\frac{\Delta F_l(x; d)}{\alpha^*(x, d)}. \tag{2.16}$$

¿From (2.14) and (2.16) we deduce

$$\begin{aligned}\Delta F_q(x;\alpha^*(x,d)d) &= \alpha^*(x,d)\Delta F_l(x;d)+\frac{1}{2}\alpha^*(x,d)^2 d^tQd\\ &\le \alpha^*(x,d)\Delta F_l(x;d)-\frac{1}{2}\alpha^*(x,d)\Delta F_l(x;d)\\ &= \frac{1}{2}\alpha^*(x,d)\Delta F_l(x;d).\end{aligned}$$

If $d^tQd \le 0$, we have

$$\alpha^*(x,d)=\min\left\{\ 1,\ \frac{\delta}{\|d\|},\gamma\bar{\alpha}(x,d)\right\},$$

and

$$\begin{aligned}\Delta F_q(x;\alpha^*(x,d)d) &= \alpha^*(x,d)\Delta F_l(x;d)+\frac{1}{2}\alpha^*(x,d)^2 d^tQd\\ &\le \frac{1}{2}\alpha^*(x,d)\Delta F_l(x;d).\end{aligned}$$

Therefore we proved (2.12) and (2.13). □

Theorem 1 *Let an infinite sequence $\{w_k\}$ be generated by Algorithm TR for fixed $\mu > 0$ and $\rho > 0$. Then there exists an accumulation point that satisfies the barrier KKT conditions (1.5).*

Proof. We first prove that

(2.17) $$\liminf_{k\to\infty}\|\Delta x_{SDk}\|=0.$$

Since the sequence $\{\Delta x_{SDk}\}$ is uniformly bounded and the barrier terms exist in the merit function, any components of x_k do not become arbitrarily small. Thus we have

$$\liminf_{k\to\infty}\bar{\alpha}(x_k,\Delta x_{SDk})>0.$$

By Step 3 of Algorithm TR and Lemma 5, we have

(2.18)
$$\Delta F_q(x_k;s_k)\le \frac{1}{4}\Delta F_l(x_k;\Delta x_{SDk})\min\left\{\ 1,\ \frac{\delta_k}{\|\Delta x_{SDk}\|},\gamma\bar{\alpha}(x_k,\Delta x_{SDk}),-\frac{\Delta F_l(x_k;\Delta x_{SDk})}{\max\{\Delta x_{SDk}^tQ_k\Delta x_{SDk},0\}}\right\}.$$

We define subsequences $K_1\subset\{0,1,\cdots\}$ and $K_2\subset\{0,1,\cdots\}$ that satisfy $K_1\cup K_2=\{0,1,2,\cdots\}$ and $K_1\cap K_2=\emptyset$ by

(2.19) $$\Delta F(x_k;s_k)>\frac{1}{4}\Delta F_q(x_k;s_k),\ \ k\in K_1,$$

(2.20) $$\Delta F(x_k;s_k)\le\frac{1}{4}\Delta F_q(x_k;s_k),\ \ k\in K_2.$$

(i) Suppose that K_1 is an infinite sequence.
(i-a) If $\liminf_{k\to\infty, k\in K_1} \delta_k = 0$, then there exists an infinite set $K_1' \subset K_1$ such that $\delta_k \to 0, k \in K_1'$. Then because $\|s_k\| \le \delta_k$, we have $\|s_k\| \to 0,\ k \in K_1'$. Suppose $\liminf_{k\to\infty} \|\Delta x_{SDk}\| > 0$. Then Assumption (G6) and (2.4) yield

$$\liminf_{k\to\infty, k\in K_1'} |\Delta F_l(x_k; \Delta x_{SDk})| > 0.$$

On the other hand, we have

$$\begin{aligned} \Delta F(x_k; s_k) &= \Delta F_l(x_k; s_k) + \mathrm{O}\left(\|s_k\|^2\right) \\ &= \Delta F_q(x_k; s_k) + \mathrm{O}\left(\|s_k\|^2\right). \end{aligned} \tag{2.21}$$

¿From (2.19) and (2.21), we have

$$-\Delta F_q(x_k; s_k) < \mathrm{O}\left(\|s_k\|^2\right).$$

However this contradicts (2.18), because it gives the relation

$$-\Delta F_q(x_k; s_k) \ge \frac{|\Delta F_l(x_k; \Delta x_{SDk})|}{4\|\Delta x_{SDk}\|} \|s_k\| = \mathrm{O}\left(\|s_k\|\right),$$

for sufficiently large $k \in K_1'$. Thus we obtain $\liminf_{k\to\infty} \|\Delta x_{SDk}\| = 0$ in this case.
(i-b) If $\liminf_{k\to\infty, k\in K_1} \delta_k > 0$, the condition $\Delta F(x_k; s_k) \le \frac{3}{4}\Delta F_q(x_k; s_k)$ must be satisfied infinitely many times for $k \notin K_1$ and this case corresponds to (ii) below.
(ii) Suppose that K_2 is an infinite sequence.
(ii-a) Suppose that there exists an infinite sequence $K_2' \subset K_2$ such that $\liminf_{k\to\infty, k\in K_2'} \delta_k > 0$. Since $\{F(x_k, \mu)\}$ is bounded below and decreasing, and $\Delta F(x_k; s_k) \le 0$ for $k \in K_2$, we have

$$F(x_{k+1}, \mu) - F(x_k, \mu) = \Delta F(x_k; s_k) \to 0, \quad k \in K_2$$

and thus $\Delta F_q(x_k; s_k) \to 0,\ k \in K_2$, from (2.20). Therefore we have $\Delta F_l(x_k; \Delta x_{SDk}) \to 0,\ k \in K_2'$, from (2.18). Then, by (2.4) we obtain $\Delta x_{SDk} \to 0,\ k \in K_2'$, and thus $\liminf_{k\to\infty} \|\Delta x_{SDk}\| = 0$ in this case.
(ii-b) Suppose $\lim_{k\to\infty, k\in K_2} \delta_k = 0$. Then the condition $\Delta F(x_k; s_k) > \frac{1}{4}\Delta F_q(x_k; s_k)$ must be satisfied infinitely many times. This case corresponds to (i) above. If the case (i-a) holds, then (2.17) is proved as above. Otherwise we prove that the case (i-b) does not occur in this case. Suppose that we have the case in which (i-b) occurs. Then $\liminf_{k\to\infty, k\in K_1} \delta_k > 0$ and $\lim_{k\to\infty, k\in K_2} \delta_k = 0$. This is a contradiction because $\delta_{k+1} = \delta_k$, $\frac{1}{2}\delta_k$, or $2\delta_k$ for any k. Therefore the case (i-b) does not occur.

Thus we proved (2.17). By the requirement (2.10), this means that we have

$$\liminf_{k\to\infty} \|\Delta x_k\| = 0.$$

Thus there exists an infinite sequence $K \subset \{0, 1, \cdots\}$ and an accumulation point $\hat{x} \in \mathbf{R}^n_+$ such that

$$x_k \to \hat{x}, \quad s_k \to 0, \quad \Delta x_k \to 0, \quad x_{k+1} \to \hat{x}, \quad k \in K.$$

Since Lemma 2 and Assumption G assure the boundedness of $\left\{X_k^{-1}Z_k\right\}$, we have

$$\lim_{k\to\infty, k\in K} \left\| z_k + \Delta z_k - \mu X_k^{-1} e \right\| = 0$$

from (1.8). If we define $\hat{z} = \mu \hat{X}^{-1} e$ where $\hat{X} = \mathrm{diag}(\hat{x}_1, \cdots, \hat{x}_n)$, then we have

$$z_k + \Delta z_k \to \hat{z}, \quad k \in K.$$

Hence from (2.8) we have

$$(c_{Lk})_i \le \frac{\mu}{M_L} \le (x_k + s_k)_i (z_k + \Delta z_k)_i \le M_U \mu \le (C_{Uk})_i, \quad i = 1, \cdots, n$$

for $k \in K$ sufficiently large, which shows that the point $z_k + \Delta z_k$ is always accepted as z_{k+1} for sufficiently large $k \in K$.

Since $\alpha_{zk} = 1$ is accepted for $k \in K$ sufficiently large, so is $\alpha_{yk} = 1$. Because the matrix $A(\hat{x})$ is of full rank, the sequence $\{y_k + \Delta y_k\}, k \in K$ converges to a point $\hat{y} \in \mathbf{R}^m$ from (1.8). Thus we proved that $(x_{k+1}, y_{k+1}, z_{k+1}) \to (\hat{x}, \hat{y}, \hat{z})$ for $k \in K$ and that

$$\begin{aligned} \nabla f(\hat{x}) - A(\hat{x})^t \hat{y} - \hat{z} &= 0, \\ g(\hat{x}) &= 0, \\ \hat{X}\hat{z} &= \mu e, \ \hat{x} > 0, \hat{z} > 0. \end{aligned}$$

This completes the proof. □

2.3 Quadratic convergence

In the proof of the above global convergence theorem, the explicit form of the matrix Q is arbitrary. However it is better to have a good form of Q that gives a fast local convergence to the barrier KKT point for fixed μ. For this purpose we set

(2.22) $$Q = \nabla_x^2 L(w) + X^{-1} Z.$$

In the following theorem we show that it is possible to prove that under Assumption G and additional assumptions, the sequence generated by Algorithm TR converges to a barrier KKT point quadratically.

Theorem 2 *Let $w(\mu) = (x(\mu), y(\mu), z(\mu))$ be a solution to the barrier KKT conditions (1.5) and let an infinite sequence $\{w_k\}$ be generated by Algorithm TR for fixed $\mu > 0$ and $\rho > 0$. Suppose the following assumptions in addition to Assumption G:*

(Q1) The sequence $\{w_k\}$ converges to $w(\mu)$.

(Q2) The second order sufficient condition for optimality of problem (1.4) holds at $w(\mu)$.

(Q3) If Δx_k satisfies conditions (2.7), then s_k is set to be Δx_k in Algorithm TR.

(Q4) Δx_k *satisfies (2.10) without modifying the matrix* G_k *in Step 2 of Algorithm TR for* k *sufficiently large.*

(Q5) The penalty parameter ρ *satisfies*

$$\rho \geq \|y_k\|_\infty + \zeta \tag{2.23}$$

for each k*, where* ζ *is a positive constant.*

(Q6) The Hessian matrices of the constraint functions are sufficiently small to satisfy

$$\rho \sum_{i=1}^{m} |s_k^t \nabla^2 g_i(x_k) s_k| < -\frac{1}{2} \Delta F_q(x_k; s_k) \tag{2.24}$$

for sufficiently large k*.*

Then the sequence $\{w_k\}$ *converges quadratically to* $w(\mu)$*.*

Proof. In the following, we assume that k is sufficiently large. We note that the second order sufficient condition for optimality of problem (1.4) implies that there exist positive constants β_0' and β_1' such that

$$v^t \left(\nabla_x^2 L(w(\mu)) + X(\mu)^{-1} Z(\mu) + \beta_0' A(x(\mu)) A(x(\mu))^t \right) v \geq \beta_1' \|v\|^2$$

for all $v \in \mathbf{R}^n$. Hence there exist positive constants β_0 and β_1 such that

$$v^t (Q_k + \beta_0 A(x_k) A(x_k)^t) v \geq \beta_1 \|v\|^2 \quad \text{for all} \quad v \in \mathbf{R}^n. \tag{2.25}$$

We first prove that if $\|\Delta x_k\| \leq \delta_k$, then Δx_k satisfies conditions (2.7). The first condition of (2.7) clearly holds. Since $\lim_{k\to\infty} \gamma x_k = \gamma x(\mu) > 0$ and $\lim_{k\to\infty} \Delta x_k = 0$, we have $-\gamma(x_k)_i < (\Delta x_k)_i$ for $i = 1, \cdots, n$, which implies the second condition of (2.7). The Newton direction Δw_k satisfies

$$\begin{aligned} \nabla f(x_k) + Q_k \Delta x_k - \mu X_k^{-1} e - A(x_k)^t (y_k + \Delta y_k) &= 0, \\ g(x_k) + A(x_k) \Delta x_k &= 0. \end{aligned}$$

Since the above equations are the first order necessary conditions for optimality of minimizing $\Delta F_q(x_k; s)$ by assumption (Q5) (see [25]), Δx_k becomes a minimizer of $\Delta F_q(x_k; s)$ by the second order sufficient condition for optimality of problem (1.4). Then we have

$$\begin{aligned} \Delta F_q(x_k; \Delta x_k) &\leq \Delta F_q(x_k; \alpha^*(x_k, \Delta x_{SDk}) \Delta x_{SDk}) \\ &\leq \frac{1}{2} \Delta F_q(x_k; \alpha^*(x_k, \Delta x_{SDk}) \Delta x_{SDk}), \end{aligned}$$

which implies the third condition of (2.7). Therefore

$$s_k = \Delta x_k \tag{2.26}$$

is accepted.

Secondly we show that

$$\Delta F_q(x_k; s_k) \le -\beta_2 \|s_k\|^2 \tag{2.27}$$

for a positive constant β_2.
(i) Suppose that $\|\Delta x_k\| \le \delta_k$. Since $s_k = \Delta x_k$ is accepted by (2.26), equations (2.3), (2.23) and (2.25) yield

$$\begin{aligned}\Delta F_q(x_k; s_k) &= \Delta F_l(x_k; \Delta x_k) + \frac{1}{2}\Delta x_k^t Q_k \Delta x_k \\ &\le -\frac{1}{2}\Delta x_k^t Q_k \Delta x_k - (\rho - \|y_k + \Delta y_k\|_\infty)\sum_{i=1}^m |g_i(x_k)| \\ &\le -\frac{1}{2}\beta_1 \|\Delta x_k\|^2 - \frac{1}{2}\zeta \sum_{i=1}^m |g_i(x_k)| + \mathrm{O}(\|g(x_k)\|^2) \\ &\le -\frac{1}{2}\beta_1 \|s_k\|^2.\end{aligned}$$

(ii) Suppose that $\|\Delta x_k\| > \delta_k$. In the same way as the proof of Case (i), equations (2.7), (2.13), (2.4) and the uniformly positive definiteness of the matrix $D_k + X_k^{-1} Z_k$ yield

$$\begin{aligned}\Delta F_q(x_k; s_k) &\le \frac{1}{2}\Delta F_q(x_k; \alpha^*(x_k, \Delta x_{SDk})\Delta x_{SDk}) \\ &\le \frac{1}{4}\alpha^*(x_k, \Delta x_{SDk})\Delta F_l(x_k; \Delta x_{SDk}) \\ &\le -\frac{1}{4}\beta_3 (\alpha^*(x_k, \Delta x_{SDk}))^2 \|\Delta x_{SDk}\|^2, \end{aligned} \tag{2.28}$$

where β_3 is a positive constant and $\alpha^*(x_k, \Delta x_{SDk})$ is given by (2.12). Since $x_k \to x(\mu)$ and $\Delta x_{SDk} \to 0$, we have

$$\gamma \bar{\alpha}(x_k, \Delta x_{SDk}) > 1.$$

Thus we only consider the following three cases.
(ii-a) If $\alpha^*(x_k, \Delta x_{SDk}) = 1$, then (2.28) and (G5) yield

$$\Delta F_q(x_k; s_k) \le -\frac{1}{4}\beta_3 \|\Delta x_{SDk}\|^2 \le -\frac{\beta_3}{4M^2}\|s_k\|^2.$$

(ii-b) If $\alpha^*(x_k, \Delta x_{SDk}) = \dfrac{\delta_k}{\|\Delta x_{SDk}\|}$, then (2.28) and $\|s_k\| \le \delta_k$ yield

$$\Delta F_q(x_k; s_k) \le -\frac{1}{4}\beta_3 \delta_k^2 \le -\frac{1}{4}\beta_3 \|s_k\|^2.$$

(ii-c) If $\alpha^*(x_k, \Delta x_{SDk}) = -\dfrac{\Delta F_l(x_k; \Delta x_{SDk})}{\Delta x_{SDk}^t Q_k \Delta x_{SDk}}$, then (2.4) yields

$$\alpha^*(x_k, \Delta x_{SDk}) \ge \frac{\Delta x_{SDk}^t (D_k + X_k^{-1} Z_k)\Delta x_{SDk} + (\rho - \|y_k + \Delta y_{SDk}\|_\infty)\sum_{i=1}^m |g_i(x_k)|}{\Delta x_{SDk}^t Q_k \Delta x_{SDk}}.$$

Since there exist positive constants β_4 and β_5 such that

$$\Delta x_{SDk}^t (D_k + X_k^{-1} Z_k)\Delta x_{SDk} \ge \beta_4 \|\Delta x_{SDk}\|^2 \quad \text{and} \quad \Delta x_{SDk}^t Q_k \Delta x_{SDk} \le \beta_5 \|\Delta x_{SDk}\|^2,$$

we have

$$\alpha^*(x_k, \Delta x_{SDk}) \geq \frac{\beta_4}{\beta_5} > 0.$$

Hence it follows from (2.28) and (G5) that

$$\Delta F_q(x_k; s_k) \leq -\frac{1}{4}\beta_3 \left(\frac{\beta_4}{\beta_5}\right)^2 \|\Delta x_{SDk}\|^2 \leq -\frac{\beta_3\beta_4^2}{4M^2\beta_5^2}\|s_k\|^2.$$

Therefore by (i) and (ii), we obtain (2.27).

We thirdly prove that

$$\Delta F(x_k; s_k) \leq \Delta F_q(x_k; s_k) + \rho \sum_{i=1}^m \left| s_k^t \nabla^2 g_i(x_k) s_k \right| + \mathrm{o}(\|s_k\|^2). \tag{2.29}$$

Since

$$\mu \sum_{i=1}^n \log\left(1 + \frac{(s_k)_i}{(x_k)_i}\right) = \mu s_k^t X_k^{-1} e - \frac{1}{2}\mu s_k^t X_k^{-2} s_k + \mathrm{o}(\|s_k\|^2),$$

we have

$$\begin{aligned}
F(x_k + s_k, \mu) &= f(x_k) + \nabla f(x_k)^t s_k + \frac{1}{2} s_k^t \nabla^2 f(x_k) s_k + \mathrm{o}(\|s_k\|^2) \\
&\quad -\mu \sum_{i=1}^n \log(x_k)_i - \mu \sum_{i=1}^n \log\left(1 + \frac{(s_k)_i}{(x_k)_i}\right) \\
&\quad +\rho \sum_{i=1}^m \left| g_i(x_k) + \nabla g_i(x_k)^t s_k + \frac{1}{2} s_k^t \nabla^2 g_i(x_k) s_k \right| \\
&\leq F(x_k, \mu) + \Delta F_l(x_k; s_k) + \frac{1}{2} s_k^t (\nabla_x^2 L(w_k) + X_k^{-1} Z_k) s_k \\
&\quad +\frac{1}{2}\rho \sum_{i=1}^m \left| s_k^t \nabla^2 g_i(x_k) s_k \right| + \frac{1}{2} \sum_{i=1}^m (y_k)_i s_k^t \nabla^2 g_i(x_k) s_k + \mathrm{o}(\|s_k\|^2) \\
&\leq F(x_k, \mu) + \Delta F_q(x_k; s_k) + \rho \sum_{i=1}^m \left| s_k^t \nabla^2 g_i(x_k) s_k \right| + \mathrm{o}(\|s_k\|^2).
\end{aligned}$$

Thus we obtain (2.29).

By (2.29), (2.24) and (2.27), we have

$$\begin{aligned}
\Delta F(x_k; s_k) &< \frac{1}{2}\Delta F_q(x_k; s_k) + \mathrm{o}(\|s_k\|^2) \\
&= \frac{1}{4}\Delta F_q(x_k; s_k) + \left(\frac{1}{4}\Delta F_q(x_k; s_k) + \mathrm{o}(\|s_k\|^2)\right) \\
&\leq \frac{1}{4}\Delta F_q(x_k; s_k) + \left(-\frac{1}{4}\beta_2\|s_k\|^2 + \mathrm{o}(\|s_k\|^2)\right) \\
&< \frac{1}{4}\Delta F_q(x_k; s_k).
\end{aligned}$$

Step 4 of Algorithm TR implies

$$\delta_{k+1} = 2\delta_k \quad \text{or} \quad \delta_{k+1} = \delta_k,$$

and then we have

$$\liminf_{k\to\infty} \delta_k > 0.$$

Since $\Delta x_k \to 0$ yields $\|\Delta x_k\| < \delta_k$, by (2.26), $s_k = \Delta x_k$ is accepted. Therefore since assumptions guarantee the nonsingularity of $J(w_k)$ and the pure Newton step Δw_k is chosen, we obtain the quadratic rate of convergence of Algorithm TR. □

We note that (2.24) is a condition for avoiding the Maratos effect and that this is not so unrealistic condition for some problems. In fact, if the magnitude of $\nabla^2 g_i(x_k)$ is sufficiently small for $i = 1, \cdots, m$, equation (2.27) guarantees (2.24). Instead of assuming the above conditions, we could have a method for avoiding Maratos effect, an example of such method is shown below, even in the iterations for fixed μ. We do not include this kind of strategy in Algorithm TR mainly because of simplicity of the given algorithm. Also we note that in the algorithm given below a search with fixed μ is terminated when a point that approximately satisfies the barrier KKT conditions is obtained. Therefore Maratos effect does not occur actually in the inner loop of Algorithm IPTR given below.

3 Our method and its global convergence

In the previous section, we showed global convergence of Algorithm TR for finding a barrier KKT point. Since KKT conditions (1.3) are obtained by letting $\mu \to 0$ in barrier KKT conditions (1.5), we can expect that the sequence $\{w_k\}$ that consists of approximations to barrier KKT points obtained by Algorithm TR with $\{\mu_k\}, \mu_k \downarrow 0$ converges to a KKT point. On the other hand, in [28], local behavior of the primal-dual interior point methods is studied and superlinear convergence property is proved. Hence our aim in this paper is to propose a globally and superlinearly convergent primal-dual interior point method. To avoid the Maratos effect, we adopt a nonmonotone strategy which is similar to the one used in [27] for the SQP method.

In this section, we present our new method, *a primal-dual interior point trust region method*, and show its global convergence. In the next section, we will analyze local behavior of our method and show that the Maratos effect does not occur, hence superlinear convergence of our method.

In the following algorithm, iterates consist of points $w_{k+1}, k = 0, 1, ...$ that satisfy the condition

$$\|r(w_{k+1}, \mu_k)\| \le M_c \mu_k, \tag{3.1}$$

where M_c is a given positive constant. This condition means that the point w_{k+1} approximately satisfies the barrier KKT conditions for the barrier parameter μ_k. Therefore we call condition (3.1) the approximate barrier KKT condition. By using Algorithm TR we can obtain such a point as shown above. Therefore it is easy to have a globally convergent algorithm for obtaining a KKT point with the decreasing sequence $\{\mu_k\}$ that converges to 0. However, we have to expect that Maratos effect may occur at the final stage of the iteration because of the use of l_1 type penalty function in our algorithm.

To avoid this effect, we include the nonmonotone procedure in Step 2 of Algorithm IPTR below. We could use a sort of second order correction steps to avoid the Maratos effect (see for example [14] and [20]). However we think such extra steps may complicate the algorithm and necessitates extra computations. On the other hand, the Maratos effect itself is somewhat an artificial one because it comes from the use of the l_1 merit function for attaining global convergence. Therefore we adopt a strategy which uses the original Newton direction only. The nonmonotone step which will be described below is just Newton direction for the barrier KKT conditions, and we try to adopt the step even if it raises the value of the barrier penalty function. We note that our nonmonotone step is different from the one proposed in [16].

In the nonmonotone step which is tried at the initial step with a newly updated barrier parameter (an approximate barrier KKT point for the previous barrier parameter value), we test the quality of the direction by using the parameter λ_k. The parameter λ_k is called the bounding parameter for the nonmonotone step. If a nonmonotone step gives a merit function value that is not less than λ_k, we discard the point and resort to usual trust region strategy (Algorithm TR). Otherwise we adopt the point as a closer approximation to a barrier KKT point even if the merit function value does not decrease. If the point obtained by the nonmonotone step satisfies the above approximate barrier KKT condition, we are ready to reduce the barrier parameter. If not, we resort to Algorithm TR to obtain such a point.

In our algorithm below values of λ_k have some flexibility. Initially we set $\lambda_0 = F(x_0, \mu_{-1})$, i.e., the merit function value at the initial point. This value acts as the largest allowable value of the merit function in the iterations hereafter. Let a point $x_k + \alpha_{xk}\Delta x_k$ be given by the nonmonotone step at x_k where $\alpha_{xk} > 0$ is a step size which will be defined below. If the point $x_k + \alpha_{xk}\Delta x_k$ is discarded because $F(x_k + \alpha_{xk}\Delta x_k, \mu_k) \geq \lambda_k$, then we have $\lambda_{k+1} = \lambda_k$. Otherwise we have $\lambda_{k+1} \in [\max\{F(x_k, \mu_k), F(x_k + \alpha_{xk}\Delta x_k, \mu_k)\}, F(x_0, \mu_{-1})]$. We will prove that the Maratos effect can be avoided with these parameter values.

Now we give the algorithm as follows:

Algorithm IPTR

Step 0. (Initialize)

Choose parameters $\rho > 0$, $M_c > 0$ and $\varepsilon > 0$. Select an initial point $w_0 \in \mathbf{R}^n_+ \times \mathbf{R}^m \times \mathbf{R}^n_+$ and a positive parameter μ_{-1} such that $\|r(w_0, \mu_{-1})\| \leq M_c\mu_{-1}$. Set $\lambda_0 = F(x_0, \mu_{-1})$ and $k = 0$.

Step 1. (Termination)

If $\|r_0(w_k)\| \leq \varepsilon$, then stop. Otherwise choose $\mu_k \in (0, \mu_{k-1})$.

Step 2. (Nonmonotone Procedure)

Step 2.1 Compute a search direction Δw_k by solving

$$J(w_k)\Delta w_k = -r(w_k, \mu_k). \tag{3.2}$$

If $J(w_k)$ is singular, then set $\lambda_{k+1} = \lambda_k$ and go to Step 3.

Step 2.2 Compute $\Lambda_k = \mathrm{diag}(\alpha_{xk}I_n, \alpha_{yk}I_m, \alpha_{zk}I_n) > 0$ such that $x_k + \alpha_{xk}\Delta x_k > 0$ and $z_k + \alpha_{zk}\Delta z_k > 0$, where I_n and I_m are n-th and m-th order identity matrices respectively.

Step 2.3 If $F(x_k + \alpha_{xk}\Delta x_k, \mu_k) \geq \lambda_k$, then set $\lambda_{k+1} = \lambda_k$ and go to Step 3.

Step 2.4 Set $\lambda_{k+1} \in [\max\{F(x_k, \mu_k), F(x_k + \alpha_{xk}\Delta x_k, \mu_k)\}, F(x_0, \mu_{-1})]$.

Step 2.5 If $\|r(w_k + \Lambda_k \Delta w_k, \mu_k)\| \leq M_c \mu_k$, then set $w_{k+1} = w_k + \Lambda_k \Delta w_k$ and go to Step 4. Otherwise go to Step 3.

Step 3. (Trust Region Procedure)

Find a new point w_{k+1} that satisfies the approximate barrier KKT condition (3.1) by Algorithm TR.

Step 4. Set $k := k + 1$ and go to Step 1. □

As described in Section 2, we can find a point that satisfies (3.1) by Algorithm TR from any starting point. On the other hand, in order for the interval of λ_{k+1} in Step 2.4 to be well defined, we need to use a starting point x which satisfies $F(x, \mu_k) \leq \lambda_k$ in Step 3. In fact, the point w_k or $w_k + \Lambda_k \Delta w_k$ is used as a starting point in Step 3. Specifically, we may use w_k if a jump from Step 2.1 or Step 2.3 occurs, and $w_k + \Lambda_k \Delta w_k$ if a jump from Step 2.5 occurs. Though in Step 0 of Algorithm IPTR, we could start from any initial point w_0, we start from w_0 such that $\|r(w_0, \mu_{-1})\| \leq M_c \mu_{-1}$ for simplicity.

By assumption (G2), there exists a constant $\bar{x}_i$ such that

$$0 < \frac{(x_k)_i}{\bar{x}_i} < 1, \quad i = 1, 2, ..., n$$

at each k, and the merit function (1.6) could be replaced by

$$F(x, \mu) = f(x) - \mu \sum_{i=1}^{n} \log\left(\frac{x_i}{\bar{x}_i}\right) + \rho \sum_{i=1}^{m} |g_i(x)|,$$

if necessary. Therefore, in proving global convergence of our method, we can assume that

$$0 < (x_k)_i < 1, \quad i = 1, 2, ..., n$$

for all k, without loss of generality. This guarantees the monotone decreasing of the barrier term with respect to μ, i.e.

$$-\mu_k \sum_{i=1}^{n} \log(x_{k-1})_i \leq -\mu_{k-1} \sum_{i=1}^{n} \log(x_{k-1})_i$$

for $i = 1, ..., n$.

In the following theorem, we show the global convergence property of Algorithm IPTR.

Theorem 3 *Let $\{w_k\}$ be an infinite sequence generated by Algorithm IPTR with $\{\mu_k\}$, $\mu_k \downarrow 0$. Suppose that the sequences $\{x_k\}$ and $\{y_k\}$ are bounded. Then $\{z_k\}$ is bounded, and any accumulation point of $\{w_k\}$ satisfies KKT conditions (1.3) of problem (1.1).*

Proof. Assume that there exists an i such that $(z_k)_i \to \infty$. Equation (3.1) yields

$$\left| \frac{(\nabla f(x_k) - A(x_k)^t y_k)_i}{(z_k)_i} - 1 \right| \le M_c \frac{\mu_{k-1}}{(z_k)_i},$$

which is a contradiction because of the boundedness of $\{x_k\}$ and $\{y_k\}$. Thus the sequence $\{z_k\}$ is bounded.

Let $\hat{w}$ be any accumulation point of $\{w_k\}$. Since the sequences $\{w_k\}$ and $\{\mu_k\}$ satisfy (3.1) for each k and μ_k approaches zero, $r_0(\hat{w}) = 0$ follows from the definition of $r(w, \mu)$. Therefore the proof is complete. □

4 Superlinear Convergence

In this section, we discuss the convergence rate of Algorithm IPTR. In order to obtain fast convergence, we must choose suitable step sizes. Following the analysis by Yamashita and Yabe [28], we define the step sizes by the rule:

$$\alpha_{xk} = \min \left\{ 1,\ \gamma_k \min_i \left\{ -\frac{(x_k)_i}{(\Delta x_k)_i} \middle| (\Delta x_k)_i < 0 \right\} \right\}, \tag{4.1}$$

$$\alpha_{zk} = \min \left\{ 1,\ \gamma_k \min_i \left\{ -\frac{(z_k)_i}{(\Delta z_k)_i} \middle| (\Delta z_k)_i < 0 \right\} \right\} \tag{4.2}$$

and

$$\alpha_{yk} = 1, \quad \text{or} \quad \alpha_{xk}, \quad \text{or} \quad \alpha_{zk},$$

where $\gamma_k \in (0, 1)$.

Let $w^* = (x^*, y^*, z^*)^t$ be a KKT point of (1.1). In the following, we assume that k is sufficiently large and μ_k is sufficiently close to 0. In order to prove superlinear convergence, we need Assumption L.

Assumption L

(L1) The sequence $\{w_k\}$ converges to w^*.

(L2) The second derivatives of the functions f and g are Lipschitz continuous at x^*.

(L3) The linear independence of active constraint gradients, the second order sufficient condition for optimality and the strict complementarity condition hold at w^*.

(L4) $\rho \ge \|y_k\|_\infty + \zeta$ for all k, where ζ is a positive constant.

(L5) μ_k and γ_k are updated by the rules

$$\mu_k = \xi_k \|r_0(w_k)\|^{1+\tau_1} \quad \text{and} \quad 1 - \gamma_k = \kappa \xi_k \|r_0(w_k)\|^{\tau_2}$$

for positive constants τ_1, τ_2 and κ such that $\min(1, \tau_2) > \tau_1$ and $0 < \kappa < 1$, and for a positive number ξ_k such that $\frac{1}{M'} \le \xi_k \le M'$, where M' is a positive constant.

(L6) $0 < M_c < \sqrt{n}$.

□

By (L1), (L2) and (L3), the Jacobian matrix $\nabla r(w_k, \mu_k)$ is nonsingular and

$$\|\nabla r(w_k, \mu_k)^{-1}\| \leq \nu$$

holds for a positive constant ν. Thus the linear system of equations (3.2) has a unique solution and Step 2.2 is always performed.

First we give the following theorem, which plays an important role in showing superlinear convergence property of Algorithm IPTR.

Theorem 4 *(1) If a point $\hat{w} \in \mathbf{R}^n_+ \times \mathbf{R}^m \times \mathbf{R}^n_+$ satisfies $\|r(\hat{w}, \mu_k)\| \leq M_c\mu_k$, then*

$$\nu_1\|r_0(w_k)\|^{1+\tau_1} \leq \|r_0(\hat{w})\| \leq \nu_2\|r_0(w_k)\|^{1+\tau_1} \tag{4.3}$$

for positive constants ν_1 and ν_2.
(2) $\Lambda_k = I$.
(3) There holds

$$\|r(w_k + \Delta w_k, \mu_k)\| \leq M_c\mu_k. \tag{4.4}$$

Proof. (1) Since $\|r(\hat{w}, \mu_k)\| \leq M_c\mu_k$, we have

$$\|r_0(\hat{w})\| = \left\| r(\hat{w}, \mu_k) + \mu_k \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \right\| = \mathrm{O}(\mu_k) = \mathrm{O}(\|r_0(w_k)\|^{1+\tau_1}).$$

Furthermore we obtain

$$\begin{aligned} \|r_0(\hat{w})\| &= \left\| r(\hat{w}, \mu_k) + \mu_k \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \right\| \geq \mu_k \left\| \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \right\| - \|r(\hat{w}, \mu_k)\| \\ &\geq (\sqrt{n} - M_c)\mu_k \geq \frac{\sqrt{n} - M_c}{M'}\|r_0(w_k)\|^{1+\tau_1}. \end{aligned}$$

(2) We will show that

$$\alpha_{xk} = \min\left\{ 1,\ \gamma_k \min_i \left\{ -\frac{(x_k)_i}{(\Delta x_k)_i} \middle| (\Delta x_k)_i < 0 \right\} \right\} = 1.$$

For i such that $(x^*)_i > 0$, it follows from $(\Delta x_k)_i \to 0$ and $\gamma_k \to 1$ that

$$-\gamma_k \frac{(x_k)_i}{(\Delta x_k)_i} > 1 \quad ((\Delta x_k)_i < 0). \tag{4.5}$$

Now we consider an index i such that $(x^*)_i = 0$. In this case we note that $(z^*)_i > 0$ by Assumption (L3). By the Newton equaion (3.2),

$$(x_k)_i(\Delta z_k)_i + (z_k)_i(\Delta x_k)_i = \mu_k - (x_k)_i(z_k)_i,$$

and then we have

$$(x_k)_i + (\Delta x_k)_i = \frac{\mu_k}{(z_k)_i} - \frac{(x_k)_i(\Delta z_k)_i}{(z_k)_i}. \tag{4.6}$$

Since $\|r(w_k, \mu_{k-1})\| \le M_c\mu_{k-1}$, we have

$$\mu_k = \mathrm{O}(\|r_0(w_k)\|^{1+\tau_1}) = \mathrm{O}(\|r_0(w_{k-1})\|^{(1+\tau_1)^2}) \tag{4.7}$$

by result (1), and

$$|(x_k)_i(z_k)_i - \mu_{k-1}| \le M_c\mu_{k-1}.$$

The latter yields

$$(x_k)_i \le \frac{(1+M_c)\mu_{k-1}}{(z_k)_i} = \frac{1+M_c}{(z_k)_i}\xi_k\|r_0(w_{k-1})\|^{1+\tau_1}.$$

Since

$$(\Delta z_k)_i \le \|\Delta w_k\| = \mathrm{O}(\|r(w_k,\mu_k)\|) = \mathrm{O}(\|r_0(w_k)\|) = \mathrm{O}(\|r_0(w_{k-1})\|^{1+\tau_1}),$$

we have

$$(x_k)_i(\Delta z_k)_i = \mathrm{O}\left(\|r_0(w_{k-1})\|^{2(1+\tau_1)}\right). \tag{4.8}$$

Assumption (L5) implies $(1+\tau_1)^2 < 2(1+\tau_1)$. Thus it follows from (4.6), (4.7) and (4.8) that

$$(x_k)_i + (\Delta x_k)_i > \kappa\frac{\mu_k}{(z_k)_i}, \tag{4.9}$$

where κ is given by (L5). Since $(x_k)_i(z_k)_i \le \|r_0(w_k)\|$, Assumption (L5) guarantees

$$\begin{aligned}\frac{\mu_k}{(z_k)_i} &= \frac{\xi_k\|r_0(w_k)\|^{1+\tau_1}}{(z_k)_i} \ge \xi_k(x_k)_i\|r_0(w_k)\|^{\tau_1}\\ &\ge \xi_k(x_k)_i\|r_0(w_k)\|^{\tau_2} = \frac{1}{\kappa}(x_k)_i(1-\gamma_k),\end{aligned}$$

then we have

$$\kappa\frac{\mu_k}{(z_k)_i} \ge (x_k)_i(1-\gamma_k). \tag{4.10}$$

Thus by (4.9) and (4.10) we obtain

$$(x_k)_i + (\Delta x_k)_i > (1-\gamma_k)(x_k)_i,$$

which implies

$$\gamma_k\left(-\frac{(x_k)_i}{(\Delta x_k)_i}\right) > 1 \quad \text{for} \quad (\Delta x_k)_i < 0.$$

Hence (4.5) holds for any i such that $(\Delta x_k)_i < 0$, and we have $\alpha_{xk} = 1$.

In the same way as above, we can prove that

$$\alpha_{zk} = \min\left\{\, 1,\ \gamma_k \min_i\left\{-\frac{(z_k)_i}{(\Delta z_k)_i}\,\middle|\, (\Delta z_k)_i < 0\right\}\right\} = 1.$$

Therefore the result follows.
(3) ¿From the Newton equation (3.2) and Assumption (L5), we directly obtain

$$\begin{aligned}\|r(w_k+\Delta w_k,\mu_k)\| &= \|r(w_k,\mu_k)+J(w_k)\Delta w_k+\mathrm{O}(\|\Delta w_k\|^2)\| \\ &= \mathrm{O}(\|\Delta w_k\|^2) \\ &= \mathrm{O}(\|r(w_k,\mu_k)\|^2) \\ &= \mathrm{O}(\|r_0(w_k)\|^2) \\ &= \mathrm{o}(\|r_0(w_k)\|^{1+\tau_1}) \\ &= \mathrm{o}(\mu_k) \\ &\le M_c\mu_k.\end{aligned}$$

This proves (4.4).

Therefore the proof of this theorem is complete. □

The preceding theorem shows that $w_k + \Delta w_k$ satisfies the approximate barrier KKT condition in Step 2.5, therefore if we accept this point in Step 2.3 for each k, then we obtain superlinear convergence of Algorithm IPTR from (4.3).

Theorem 5 *Let η be a positive constant. If $\lambda_k \ge F(x_{k_0},\mu_{k_0})+\eta$ for sufficiently large k_0 and each $k \ge k_0$, then Algorithm IPTR sets $w_{k+1}=w_k+\Delta w_k$ for all $k\ge k_0$ and gives a superlinear rate of convergence of $\{w_k\}$.*

Proof. Since Assumption L implies that

$$\lim_{k\to\infty} F(x_k+\Delta x_k,\mu_k) = \lim_{k\to\infty} F(x_k,\mu_k) = f(x^*),$$

we have

$$|F(x_k+\Delta x_k,\mu_k)-F(x_{k_0},\mu_{k_0})| < \eta,$$

for sufficiently large k_0 and all $k \ge k_0$. Thus the assumption of the theorem yields

$$\lambda_k > F(x_k+\Delta x_k,\mu_k),$$

and by (3) of Theorem 4, $w_{k+1} = w_k+\Delta w_k$ is accepted in Step 2.5. Therefore the superlinear convergence property follows from (1) of Theorem 4. □

In what follows, even if we do not assume the condition in the above theorem, we show that it is possible to prove that Step 2.4 and Step 2.5 are performed and Step 3 is skipped at each iteration, and that superlinear convergence property is obtained. To this end, we need additional assumptions as follows:

(L7) There exists an integer $\hat{\imath}$ such that $(x^*)_{\hat{\imath}}=0$.

(L8) $0<M_c<1$.

(L9) τ_1 and τ_2 given in (L5) satisfy $\tau_1>\sqrt{2}-1$ and $\tau_2\ge 1$. □

We note that (L7) means $x_k \neq x^*$ for all k, because x_k is always positive. Since $\|r(w_k, \mu_{k-1})\| \leq M_c \mu_{k-1}$ is satisfied for all k, (L8) yields

$$(x_k)_i (z_k)_i \geq (1 - M_c)\mu_{k-1} > 0, \quad i = 1, ..., n. \tag{4.11}$$

Theorem 6 *There hold*

(1) $\|w_k - w^*\| = \mathrm{O}(\|x_k - x^*\|)$,

(2) $\nu_3 \|\Delta x_k\| \leq \|x_k - x^*\| \leq \nu_4 \|\Delta x_k\|$ *for positive constants* ν_3 *and* ν_4,

(3) $\|\Delta w_k\| = \mathrm{O}(\|\Delta x_k\|)$,

(4) $\|\Delta x_k\| = \mathrm{o}(\|\Delta x_{k-1}\|)$.

Proof. (1) Since

$$\begin{aligned} \|w_k - w^*\| &= \mathrm{O}(\|r_0(w_k) - r_0(w^*)\| \\ &= \mathrm{O}\left(\left\| r(w_k, \mu_{k-1}) + \mu_{k-1} \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \right\| \right), \end{aligned}$$

we have

$$\|w_k - w^*\| \leq \nu_5 \mu_{k-1}, \tag{4.12}$$

where ν_5 is a positive constant. ¿From (4.11), there exists a positive constant ν_6 such that $(x_k)_i \geq \nu_6 \mu_{k-1}$ for $i = 1, ..., n$. It follows from (L7) that

$$|(x_k)_{\hat{i}} - (x^*)_{\hat{i}}| \geq \nu_6 \mu_{k-1}.$$

This implies

$$\|x_k - x^*\| \geq \nu_6 \mu_{k-1}. \tag{4.13}$$

Therefore equations (4.12) and (4.13) yield

$$\|w_k - w^*\| \leq \nu_7 \|x_k - x^*\| \tag{4.14}$$

for a positive constant ν_7.

(2) By (4.14), we have

$$\begin{aligned} \left| \frac{\|\Delta x_k\|}{\|x_k - x^*\|} - 1 \right| &= \frac{|\|x_k - x^*\| - \|\Delta x_k\||}{\|x_k - x^*\|} \\ &\leq \frac{\|x_k + \Delta x_k - x^*\|}{\|x_k - x^*\|} \\ &\leq \frac{\nu_7 \|w_k + \Delta w_k - w^*\|}{\|w_k - w^*\|}. \end{aligned}$$

Since

$$\begin{aligned}
\|w_k + \Delta w_k - w^*\| &= \mathrm{O}(\|r_0(w_k + \Delta w_k) - r_0(w^*)\|) \\
&= \mathrm{O}(\|r(w_k + \Delta w_k, \mu_k) + \mathrm{O}(\mu_k)\|) \\
&= \mathrm{O}(\|r(w_k, \mu_k) + J(w_k)\Delta w_k + \mathrm{O}(\|\Delta w_k\|^2) + \mathrm{O}(\mu_k)\|) \\
&= \mathrm{O}(\|r_0(w_k)\|^2) + \mathrm{O}(\|r_0(w_k)\|^{1+\tau_1}) \\
&= \mathrm{o}(\|r_0(w_k)\|) \\
&= \mathrm{o}(\|w_k - w^*\|),
\end{aligned}$$

we obtain

$$\left| \frac{\|\Delta x_k\|}{\|x_k - x^*\|} - 1 \right| = \mathrm{o}(1).$$

(3) By using results (1) and (2), we have

$$\begin{aligned}
\|\Delta w_k\| &= O(\|r(w_k, \mu_k)\|) = O\left(\left\| r_0(w_k) - \mu_k \begin{pmatrix} 0 \\ 0 \\ e \end{pmatrix} \right\| \right) \\
&= O(\|r_0(w_k)\|) = O(\|w_k - w^*\|) \\
&= O(\|x_k - x^*\|) = O(\|\Delta x_k\|).
\end{aligned}$$

(4) By Theorem 4 we have

$$\begin{aligned}
\|\Delta x_k\| &\le \|\Delta w_k\| = \mathrm{O}(\|r(w_k, \mu_k)\|) = \mathrm{O}(\|r_0(w_k)\|) \\
&= \mathrm{O}(\|r_0(w_{k-1})\|^{1+\tau_1}) = \mathrm{o}(\|r_0(w_{k-1})\|) = \mathrm{o}(\|w_{k-1} - w^*\|).
\end{aligned}$$

Thus (1) and (2) yield

$$\|\Delta x_k\| = \mathrm{o}(\|x_{k-1} - x^*\|) = \mathrm{o}(\|\Delta x_{k-1}\|).$$

Therefore the theorem is proved. □

Since $x_k \neq x^*$, we should note that $\Delta x_k \neq 0$ for all k from (2).

The following assumption is stated temporarily for use in Corollary 1.

($L7'$) There exists an integer $\hat{i}$ such that $(z^*)_{\hat{i}} = 0$. □

Corollary 1 *If Assumption (L7) is replaced by* ($L7'$)*, then*

(1) $\|w_k - w^*\| = \mathrm{O}(\|z_k - z^*\|)$,

(2) $\nu_3\|\Delta z_k\| \le \|z_k - z^*\| \le \nu_4\|\Delta z_k\|$ *for positive constants* ν_3 *and* ν_4,

(3) $\|\Delta w_k\| = \mathrm{O}(\|\Delta z_k\|)$,

(4) $\|\Delta z_k\| = \mathrm{o}(\|\Delta z_{k-1}\|)$.

Proof. Proof of the corollary is same as Theorem 6. □

Lemma 6 *If* $w_k = w_{k-1} + \Delta w_{k-1}$, *then*

$$\begin{aligned} -\mu_k \sum_{i=1}^n \log(x_k + \Delta x_k)_i &= -\mu_k \sum_{i=1}^n \log(x_{k-1})_i + \mathrm{o}(\|\Delta x_{k-1}\|^2) \\ &< -\mu_{k-1} \sum_{i=1}^n \log(x_{k-1})_i + \mathrm{o}(\|\Delta x_{k-1}\|^2). \end{aligned}$$

Proof. First we note that

$$\log\left(\frac{(x_{k-1})_i + (\Delta x_{k-1})_i}{(x_{k-1})_i}\right) \geq \log(1 - \gamma_{k-1}). \tag{4.15}$$

Since Theorem 4 yields

$$\|r_0(w_k)\| \leq \nu_2 \|r_0(w_{k-1})\|^{1+\tau_1},$$

Assumption (L5) yields

$$\begin{aligned} (4.16)\quad & -\mu_k \log(1 - \gamma_{k-1}) \\ &= -\xi_k \|r_0(w_k)\|^{1+\tau_1} (\log \kappa \xi_{k-1} + \log \|r_0(w_{k-1})\|^{\tau_2}) \\ &\leq -M' \nu_2^{1+\tau_1} \|r_0(w_{k-1})\|^{(1+\tau_1)^2} \left(\log \frac{\kappa}{M'} + \log \|r_0(w_{k-1})\|^{\tau_2}\right) \\ &= -M' \nu_2^{1+\tau_1} \|r_0(w_{k-1})\|^2 \left\{ \|r_0(w_{k-1})\|^{\tau_1^2 + 2\tau_1 - 1} \left(\log \frac{\kappa}{M'} + \tau_2 \log \|r_0(w_{k-1})\|\right)\right\}. \end{aligned}$$

Since $\tau_1 > \sqrt{2} - 1$ guarantees $\tau_1^2 + 2\tau_1 - 1 > 0$, it follows from (4.15), (4.16) and Theorem 6 that

$$\begin{aligned} -\mu_k \sum_{i=1}^n \log(1 + (X_{k-1}^{-1} \Delta x_{k-1})_i) &\leq -n\mu_k \log(1 - \gamma_{k-1}) \\ &= \mathrm{o}(\|r_0(w_{k-1})\|^2) \\ &= \mathrm{o}(\|w_{k-1} - w^*\|^2) \\ &= \mathrm{o}(\|x_{k-1} - x^*\|^2) \\ &= \mathrm{o}(\|\Delta x_{k-1}\|^2). \end{aligned} \tag{4.17}$$

In the same way as the above and by using Theorem 4, we have

$$\begin{aligned} (4.18)\quad & -\mu_k \sum_{i=1}^n \log(1 + (X_k^{-1} \Delta x_k)_i) \\ &\leq -n\mu_k \log(1 - \gamma_k) \\ &= -n\xi_k \|r_0(w_k)\|^{1+\tau_1} \log(\kappa \xi_k \|r_0(w_k)\|^{\tau_2}) \\ &\leq -nM' \nu_2^{1+\tau_1} \|r_0(w_{k-1})\|^2 \left\{ \|r_0(w_{k-1})\|^{\tau_1^2 + 2\tau_1 - 1} \left(\log \frac{\kappa}{M'} + \tau_2 \log \nu_1 \right.\right. \\ &\qquad \left.\left. + \tau_2(1 + \tau_1) \log \|r_0(w_{k-1})\|\right)\right\} \\ &= o(\|\Delta x_{k-1}\|^2). \end{aligned}$$

We also see that

$$(4.19)\qquad \begin{aligned}\sum_{i=1}^n \log(x_k+\Delta x_k)_i &= \sum_{i=1}^n \log(x_k)_i + \sum_{i=1}^n \log(1+(X_k^{-1}\Delta x_k)_i) \\ &= \sum_{i=1}^n \log(x_{k-1})_i + \sum_{i=1}^n \log(1+(X_{k-1}^{-1}\Delta x_{k-1})_i) \\ &\quad + \sum_{i=1}^n \log(1+(X_k^{-1}\Delta x_k)_i).\end{aligned}$$

Since Assumption (L7) implies $\sum_{i=1}^n \log(x_{k-1})_i < 0$, we have

$$(4.20)\qquad -\mu_k \sum_{i=1}^n \log(x_{k-1})_i < -\mu_{k-1}\sum_{i=1}^n \log(x_{k-1})_i.$$

Thus by expressions (4.17), (4.18), (4.19) and (4.20), we obtain the desired results. □

Let $I^* = \{i|(x^*)_i = 0\}$ and $\tilde{I}$ be a $|I^*| \times n$ matrix whose row consists of e_i^t, $i \in I^*$, where e_i denotes the i-th column vector of the identity matrix. We define

$$\tilde{A}(x) = \begin{pmatrix} A(x) \\ \tilde{I} \end{pmatrix} \in \mathbf{R}^{(m+|I^*|)\times n}.$$

Assumption (L3) implies that an augmented matrix

$$\bar{G}_k = \nabla_x^2 L(w_k) + \beta_0 \tilde{A}(x_k)^t \tilde{A}(x_k)$$

is uniformly positive definite for a sufficiently large positive constant β_0, i.e. there exists a positive constant β such that the matrix $\bar{G}_k$ satisfies

$$(4.21)\qquad v^t \bar{G}_k v \geq \beta\|v\|^2 \quad \text{for any } v \in \mathbf{R}^n.$$

Lemma 7 *There hold*

(1)
$$\begin{aligned}&-\Delta x_k^t(\nabla_x^2 L(w_k) + X_k^{-1}Z_k)\Delta x_k \\ &\quad\leq -\beta\|\Delta x_k\|^2 - \sum_{i\in I^*}\left(\frac{(z_k)_i}{(x_k)_i} - \beta_0\right)(\Delta x_k)_i^2 + \mathrm{O}(\|g(x_k)\|^2) \\ &\quad\leq -\beta\|\Delta x_k\|^2 + \mathrm{O}(\|g(x_k)\|^2),\end{aligned}$$

(2) $e^t X_k^{-1}\Delta x_k < 0$.

Proof. (1) By (4.21), we have

$$\begin{aligned}&-\Delta x_k^t(\nabla_x^2 L(w_k) + X_k^{-1}Z_k)\Delta x_k \\ &\quad= -\Delta x_k^t(\bar{G}_k - \beta_0\tilde{A}(x_k)^t\tilde{A}(x_k) + X_k^{-1}Z_k)\Delta x_k \\ &\quad\leq -\beta\|\Delta x_k\|^2 + \beta_0(g(x_k)^t g(x_k) + \sum_{i\in I^*}(\Delta x_k)_i^2) - \sum_{i=1}^n \frac{(z_k)_i}{(x_k)_i}(\Delta x_k)_i^2 \\ &\quad\leq -\beta\|\Delta x_k\|^2 - \sum_{i\in I^*}\left(\frac{(z_k)_i}{(x_k)_i} - \beta_0\right)(\Delta x_k)_i^2 + \mathrm{O}(\|g(x_k)\|^2).\end{aligned}$$

The second inequality follows from $\dfrac{(z_k)_i}{(x_k)_i} - \beta_0 > 0, \; i \in I^*$.

(2) Since Lemma 3 in [28] yields

$$\frac{(\Delta x_k)_i}{(x_k)_i} \le -1 + \frac{\mu_k}{(x_k)_i (z_k)_i} + \mathrm{O}(\|\Delta w_k\|)$$

for i such that $(x^*)_i = 0$, and

$$\frac{(\Delta x_k)_i}{(x_k)_i} = \mathrm{O}(\|\Delta w_k\|)$$

for i such that $(x^*)_i > 0$, we have

$$e^t X_k^{-1} \Delta x_k = \sum_{i=1}^n \frac{(\Delta x_k)_i}{(x_k)_i} \le -1 + \sum_{i=1}^n \frac{\mu_k}{(x_k)_i (z_k)_i} + \mathrm{O}(\|\Delta w_k\|).$$

Since (1) of Theorem 4 and (4.11) yield

$$\frac{\mu_{k-1}}{(x_k)_i (z_k)_i} \le \frac{1}{1 - M_c} \quad \text{and} \quad \mu_k = \mathrm{O}(\mu_{k-1}^{1+\tau_1}),$$

we see that

$$\frac{\mu_k}{(x_k)_i (z_k)_i} \le \nu_9 \frac{\mu_{k-1}^{1+\tau_1}}{(x_k)_i (z_k)_i} \le \frac{\nu_9}{1 - M_c} \mu_{k-1}^{\tau_1},$$

where ν_9 is a positive constant. Then we have

$$e^t X_k^{-1} \Delta x_k \le -1 + \mathrm{O}(\mu_{k-1}^{\tau_1}) + \mathrm{O}(\|\Delta w_k\|) < 0. \quad \square$$

Define

$$L_0(x_k, y_k) = f(x_k) - y_k^t g(x_k) \quad \text{and} \quad \tilde{y}_k = y_k + \Delta y_k.$$

¿From Newton's equations (3.2), we have

$$\begin{aligned} \nabla_x^2 L(w_k) \Delta x_k &= -\nabla f(x_k) + A(x_k)^t (y_k + \Delta y_k) + (z_k + \Delta z_k) \\ &= -\nabla f(x_k) + A(x_k)^t \tilde{y}_k - X_k^{-1} Z_k \Delta x_k + \mu_k X_k^{-1} e \end{aligned} \tag{4.22}$$

and therefore

$$\nabla_x L_0(x_k, \tilde{y}_k) = -\nabla_x^2 L_0(x_k, y_k) \Delta x_k - X_k^{-1} Z_k \Delta x_k + \mu_k X_k^{-1} e. \tag{4.23}$$

Theorem 7 *If $w_k = w_{k-1} + \Delta w_{k-1}$, then*

$$F(x_k + \Delta x_k, \mu_k) < F(x_{k-1}, \mu_{k-1}).$$

Proof. From equation (4.22), we have

$$\begin{aligned}
F(x_k+\Delta x_k,\mu_k) &= f(x_k+\Delta x_k)-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i \\
&\quad +\rho\sum_{i=1}^{m}|g_i(x_k+\Delta x_k)| \\
&= f(x_k)+\nabla f(x_k)^t\Delta x_k-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i \\
&\quad +\rho\sum_{i=1}^{m}|g_i(x_k)+\nabla g_i(x_k)^t\Delta x_k|+\mathrm{O}(\|\Delta x_k\|^2) \\
&= f(x_k)-\tilde{y}_k^t g(x_k)-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i \\
&\quad -\Delta x_k^t(\nabla_x^2 L(w_k)+X_k^{-1}Z_k)\Delta x_k \\
&\quad +\mu_k e^t X_k^{-1}\Delta x_k+\mathrm{O}(\|\Delta x_k\|^2).
\end{aligned}$$

Hence by Lemma 7 and Theorem 6-(4), we have

$$\begin{aligned}
F(x_k+\Delta x_k,\mu_k) &< L_0(x_k,\tilde{y}_k)-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i+\mathrm{O}(\|\Delta x_k\|^2) \\
&= L_0(x_{k-1},\tilde{y}_k)+\nabla_x L_0(x_{k-1},\tilde{y}_k)^t\Delta x_{k-1} \\
&\quad +\frac{1}{2}\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},\tilde{y}_k)\Delta x_{k-1} \\
&\quad -\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i+\mathrm{o}(\|\Delta x_{k-1}\|^2).
\end{aligned}$$

Since equation (4.23) yields

$$\begin{aligned}
\nabla_x L_0(x_{k-1},\tilde{y}_k)^t\Delta x_{k-1} &= \nabla_x L_0(x_{k-1},\tilde{y}_{k-1})^t\Delta x_{k-1}-(\tilde{y}_{k-1}-\tilde{y}_k)^t g(x_{k-1}) \\
&= -\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},y_{k-1})\Delta x_{k-1}-\Delta x_{k-1}^t X_{k-1}^{-1}Z_{k-1}\Delta x_{k-1} \\
&\quad +\mu_{k-1}\Delta x_{k-1}^t X_{k-1}^{-1}e-(\tilde{y}_{k-1}-\tilde{y}_k)^t g(x_{k-1}),
\end{aligned}$$

we have

$$\begin{aligned}
&F(x_k+\Delta x_k,\mu_k) \\
&\quad < L_0(x_{k-1},\tilde{y}_k)+\{-\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},y_{k-1})\Delta x_{k-1} \\
&\quad -\Delta x_{k-1}^t X_{k-1}^{-1}Z_{k-1}\Delta x_{k-1}+\mu_{k-1}e^t X_{k-1}^{-1}\Delta x_{k-1}-(\tilde{y}_{k-1}-\tilde{y}_k)^t g(x_{k-1})\} \\
&\quad +\frac{1}{2}\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},\tilde{y}_k)\Delta x_{k-1}-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i+\mathrm{o}(\|\Delta x_{k-1}\|^2) \\
&\quad < L_0(x_{k-1},\tilde{y}_k)-\frac{1}{2}\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},y_{k-1})\Delta x_{k-1} \\
&\quad -\frac{1}{2}\Delta x_{k-1}^t\{\nabla_x^2 L_0(x_{k-1},y_{k-1})-\nabla_x^2 L_0(x_{k-1},\tilde{y}_k)\}\Delta x_{k-1} \\
&\quad -\Delta x_{k-1}^t X_{k-1}^{-1}Z_{k-1}\Delta x_{k-1}-\mu_k\sum_{i=1}^{n}\log(x_k+\Delta x_k)_i \\
&\quad +\mathrm{o}(\|\Delta x_{k-1}\|^2)+\mathrm{o}(\|g(x_{k-1})\|).
\end{aligned}$$

Hence Lemmas 6 and 7 yield

$$\begin{aligned}
F(x_k+\Delta x_k,\mu_k) &< F(x_{k-1},\mu_{k-1})-\frac{1}{2}\alpha_{x,k-1}\Delta x_{k-1}^t\nabla_x^2 L_0(x_{k-1},y_{k-1})\Delta x_{k-1} \\
&\quad -y_{k+1}^t g(x_{k-1})-\rho\sum_{i=1}^m |g_i(x_{k-1})|-\Delta x_{k-1}^t X_{k-1}^{-1}Z_{k-1}\Delta x_{k-1} \\
&\quad +\mathrm{o}(\|\Delta x_{k-1}\|^2)+\mathrm{o}(\|g(x_{k-1})\|) \\
&< F(x_{k-1},\mu_{k-1})-(\rho-\|y_{k+1}\|_\infty)\sum_{i=1}^m |g_i(x_{k-1})|+o(\|g(x_{k-1})\|) \\
&\quad -\frac{1}{2}\Delta x_{k-1}^t(\nabla_x^2 L(w_{k-1})+X_{k-1}^{-1}Z_{k-1})\Delta x_{k-1}+\mathrm{o}(\|\Delta x_{k-1}\|^2) \\
&< F(x_{k-1},\mu_{k-1})-\zeta\sum_{i=1}^m |g_i(x_{k-1})|+\mathrm{o}(\|g(x_{k-1})\|) \\
&\quad -\frac{1}{2}\beta\|\Delta x_{k-1}\|^2+\mathrm{o}(\|\Delta x_{k-1}\|^2).
\end{aligned}$$

This implies

$$F(x_k+\Delta x_k,\mu_k)<F(x_{k-1},\mu_{k-1}).$$

The theorem is proved. □

Now we present a main result for superlinear convergence.

Theorem 8 *Algorithm IPTR sets $w_{k+1}=w_k+\Delta w_k$ for all k sufficiently large and gives a superlinear rate of convergence of $\{w_k\}$ and $\{x_k\}$.*

Proof. We first show the nonmonotone procedure in Step 2 of Algorithm IPTR is accepted at some iteration. To this end, we assume that the trust region procedure in Step 3 is performed for all k suffciently large. Since assumption (L1) implies $-\log(x_{k+1})_i>0$ for k sufficiently large, we have

$$\lambda_k\geq F(x_{k+1},\mu_k)>F(x_{k+1},\mu_{k+1})\geq F(x_{k+2},\mu_{k+1}).$$

Thus the facts that λ_k is constant for sufficiently large k and $\Delta w_k\to 0$ guarantee that there exists a sufficiently large k such that

$$F(x_k+\Delta x_k,\mu_k)<\lambda_k$$

in Step 2.3, and then Step 2.4 is performed, which is a contradiction. Since Theorem 4 implies $\|r(w_k+\Delta w_k,\mu_k)\|\leq M_c\mu_k$, we have $w_{k+1}=w_k+\Delta w_k$ in Step 2.5.

At the $(k+1)$-st iteration, Theorem 7 and the updating rule of λ_k imply

$$F(x_{k+1}+\Delta x_{k+1},\mu_{k+1})<F(x_k,\mu_k)\leq\max\{F(x_k,\mu_k),F(x_{k+1},\mu_k)\}\leq\lambda_{k+1}.$$

Thus Step 2.3 and Step 2.4 are performed, and we obtain $w_{k+2}=w_{k+1}+\Delta w_{k+1}$ in Step 2.5 because $\|r(w_{k+1}+\Delta w_{k+1},\mu_{k+1})\|\leq M_c\mu_{k+1}$ holds by Theorem 4.

Therefore nonmonotone steps (Step 2.5) are adopted hereafter, and Theorems 4 and 6 guarantee the superlinear convergence properties of $\{w_k\}$ and $\{x_k\}$, which completes the proof. □

5 Actual step

In this section, we describe how to perform the trust region iterations practically. We calculate the vector s based on the following two sets of equations:

$$(5.1)\qquad \begin{pmatrix} G & -A(x)^t & -I \\ A(x) & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} = -r(w,\mu), \quad G = \nabla_x^2 L(w)$$

and

$$(5.2)\qquad \begin{pmatrix} D & -A(x)^t & -I \\ A(x) & 0 & 0 \\ Z & 0 & X \end{pmatrix} \begin{pmatrix} \Delta x_{SD} \\ \Delta y_{SD} \\ \Delta z_{SD} \end{pmatrix} = -r(w,\mu).$$

To satisfy condition (2.10), the matrix G is added a positive diagonal matrix if the current matrix $G = \nabla_x^2 L(w)$ gives a singular or nearly singular coefficient matrix, i.e., if condition (2.10) is not satisfied. From these two sets of vectors, we calculate the vector $\overline{s}$ by

$$(5.3)\qquad \overline{s} = \nu \Delta x_{SD} + (1-\nu)\Delta x,$$

where the parameter $\nu \in [0,1]$ is determined to satisfy condition (2.7). For a given $\nu \in [0,1]$, we calculate $\alpha^*(x,\overline{s})$ and check if condition (2.7) is satisfied by $s = \alpha^*(x,\overline{s})\overline{s}$. The calculation of the step $\alpha^*(x,\overline{s})$ is easy because the function involved is quadratic. If $\nu = 1$, condition (2.7) is obviously satisfied by $s = \alpha^*(x,\overline{s})\overline{s}$. If $\nu = 0$, the resulting iteration vector s coincides with the Newton iteration vector Δx. Therefore, we try the value $\nu = 0$ first, and increase the value of ν by 0.1 until condition (2.7) is satisfied by $s = \alpha^*(x,\overline{s})\overline{s}$.

6 Implementation and numerical results

The algorithm of this paper is implemented and tested with various problems from Hock and Schittkowski's book [17] and CUTE [2]. The program is named as NUOPT 3.0. In this section, the implementation of NUOPT 3.0 and its numerical performance are described in order. All experiments are done on Pentium Pro 200MHz PC with 96MB main memory which runs under BSD/OS. Programming languages used are Fortran 77, C and C++.

6.1 problem input

The problem is specified with an objective function, upper and/or lower bounds on variables, linear equality constraints, nonlinear equality constraints, linear inequality constraints with upper and/or lower bounds and nonlinear inequality constraints with upper and/or lower bounds. Inequality constraints are converted to equality constraints and slack variables with bound(s). Our implementation can deal with upper and lower bounds on variables by modifying the algorithm of this paper.

6.2 Solution of linear equation

Our algorithm has to solve two sets of possibly large sparse linear equations (5.1) and (5.2) at each trust region iteration. The solution method of these equation is a critical point of the performance of the program. These two sets of systems may be large sparse indefinite one in general. Therefore we have to consider not only increase of fill-in factors, but also numerical stability in the course of pivotings. NUOPT 3.0 uses the supernodal right-looking method for solving these linear equations [23].

6.3 Miscellaneous details

6.3.1 Initial values of variables and parameters

Initial values of primal variables are designated by each problem. If a specified value violates a bound, then a value that satisfies a bound strictly is set in the program. Initial values of dual variables and various parameters are determined by the following rules:

$$\begin{aligned}
(z_0)_i &= \max(\|\nabla f(x_0)\|_1, 1), i = 1, \cdots, n, \\
\mu_0 &= \max(1, (x_0)^t z_0 / \max(1, n)), \\
\eta_1 &= 1, \\
\eta_2 &= 1/\max(1, n, \|\nabla f\|_1), \\
\eta_3 &= 1/\max(1, m, \|g(x_0)\|_1).
\end{aligned}$$

Other parameters include $\tau_1 = 0.6$ and $\gamma_0 = 0.99$.

6.3.2 Scaling of functions

All the functions involved are scaled as follows:

$$\begin{aligned}
f &:= f/\max(1, \|\nabla f(x_0)\|_1/n), \\
g_i &:= g_i/\max(1, \|g(x_0)\|_1/m), i = 1, \cdots, m.
\end{aligned}$$

6.3.3 Parameters

The barrier parameter μ is updated when the following condition is satisfied

$$\phi(w_k, \mu_k) \leq M_c \mu_k,$$

where

$$\phi(w, \mu) = \max\left\{\frac{\|\nabla_x L(w)\|_1}{\max(n, \|\nabla f\|_1)}, \frac{\|g(x)\|_1}{\max(1, m, \|g(x_0)\|_1)}, \frac{\|Xz - \mu e\|_1}{\max(1, n, \|x\|_1 + \|z\|_1)}\right\},$$

and $M_c = 30 \times \phi(w_0, \mu_0)$.

Convergence of primal-dual iterations is judged by :

$$\max\left\{\frac{\|\nabla_x L(w)\|_1}{\max(n, \|\nabla f\|_1)}, \frac{\|g(x)\|_1}{\max(1, m, \|g(x_0)\|_1)}, \frac{x^t z}{\max(1, n, \|x\|_1 + \|z\|_1)}\right\} < \epsilon,$$

where

$$\epsilon \equiv \sqrt{\epsilon_{mch}} \cdot 10^2 \simeq 1.4 \cdot 10^{-6}.$$

6.4 Hock & Schittkowski problems

In this subsection, we report the results for Hock and Schittkowski problems [17]. NUOPT 3.0 could solve all 114 problems with the same set of parameters.

Total number of problems = 114
Failed problem = 0
Total number of iterations = 1296
Total number of function evaluations = 2321
Total number of factorizations = 2091

- Hock & Schittkowski problems

problem	n	m	obj	res	itr	neval	nfact	time(s)	
HS1	2	1	1.8074e-13	3.9e-10	20	35	34	0.02	
HS2	2	1	4.9412	3.2e-09	8	10	10	0.02	
HS3	2	1	4.5238e-09	4.5e-09	4	6	5	0.00	
HS4	2	1	2.6667	5.0e-07	4	6	4	0.00	
HS5	2	1	-1.9132	5.1e-07	5	9	8	0.00	
HS6	2	2	0	1.9e-17	2	4	3	0.00	
HS7	2	2	-1.7321	1.2e-08	7	17	10	0.02	
HS8	2	3	-1	6.9e-12	5	7	6	0.00	
HS9	2	2	-0.5	1.2e-09	5	7	7	0.00	
HS10	2	2	-1	3.7e-11	11	18	20	0.02	
HS11	2	2	-8.4984	1.1e-06	6	8	6	0.00	
HS12	2	2	-30	3.5e-08	10	12	14	0.02	
HS14	2	3	1.3935	6.3e-07	6	8	6	0.00	
HS15	2	3	306.51	1.4e-06	8	14	9	0.02	
HS16	2	3	0.25001	9.4e-08	26	35	45	0.03	
HS17	2	3	1.0002	1.1e-06	14	28	20	0.03	
HS18	2	3	5	1.0e-09	11	15	15	0.03	
HS19	2	3	-6961.8	1.1e-07	7	15	9	0.00	
HS20	2	4	40.199	1.3e-07	6	8	6	0.00	
HS21	2	2	-99.96	1.8e-08	8	17	15	0.00	
HS22	2	3	1	4.0e-07	6	8	6	0.00	
HS23	2	6	2	1.5e-08	10	23	19	0.03	
HS24	2	4	-0.99999	1.1e-06	23	41	42	0.02	
HS25	3	1	3.3565e-13	2.3e-09	23	42	45	0.48	
HS26	3	2	4.232e-12	1.2e-06	17	23	25	0.02	
HS27	3	2	0.04	2.5e-09	23	49	43	0.03	
HS28	3	2	6.163e-32	1.1e-16	1	3	1	0.00	
HS29	3	2	-22.627	5.5e-08	7	9	11	0.02	
HS30	3	2	1	3.3e-07	6	13	11	0.02	
HS31	3	2	6	6.1e-08	6	13	11	0.02	
HS32	3	3	1	6.2e-07	9	11	10	0.00	
HS33	3	3	-4.5858	2.0e-07	19	32	33	0.03	
HS34	3	3	-0.83402	1.5e-06	9	24	18	0.02	
HS35	3	2	0.11111	4.6e-08	7	9	7	0.00	
HS36	3	2	-3300	3.7e-07	6	8	6	0.02	
HS37	3	3	-3456	2.0e-07	6	12	10	0.02	
HS38	4	1	8.5679e-10	1.4e-08	37	58	73	0.03	
HS39	4	3	-1	1.2e-10	10	27	21	0.03	
HS40	4	4	-0.25	2.6e-12	4	6	4	0.02	
HS41	4	2	1.9259	1.3e-08	7	9	7	0.02	
HS42	4	3	13.858	3.3e-11	5	8	6	0.00	
HS43	4	4	-44	3.2e-08	7	13	8	0.02	
HS44	4	7	-15	2.2e-07	9	12	13	0.02	
HS45	5	1	1	1.0e-07	7	9	8	0.00	
HS46	5	3	1.044e-10	1.3e-06	16	18	21	0.02	
HS47	5	4	2.732e-09	9.6e-07	15	28	21	0.05	
HS48	5	3	4.9304e-32	1.7e-16	1	3	1	0.00	
HS49	5	3	4.5732e-06	1.5e-06	11	13	12	0.00	
HS50	5	4	6.3837e-13	2.0e-09	8	10	8	0.00	
HS51	5	4	2.9582e-31	4.5e-16	1	3	1	0.00	
HS52	5	4	5.3266	1.4e-16	1	3	1	0.00	
HS53	5	4	4.093	5.6e-10	5	9	8	0.02	
HS54	6	2	-0.90807	2.4e-08	15	52	30	0.02	
HS55	6	7	6.6667	1.0e-06	6	8	6	0.02	
HS56	7	5	-3.456	9.3e-12	37	59	72	0.08	
HS57	2	2	0.030662	5.0e-09	26	35	48	0.07	

problem	n	m	obj	res	itr	neval	nfact	time(s)	
HS59	2	4	-7.8028	1.1e-07	11	20	15	0.02	
HS60	3	2	0.032568	3.0e-09	5	11	9	0.00	
HS61	3	3	-143.65	1.1e-06	4	9	6	0.02	
HS62	3	2	-26272	3.0e-07	6	11	7	0.00	
HS63	3	3	961.72	1.6e-07	23	64	41	0.05	
HS64	3	2	6299.8	4.7e-08	15	17	17	0.03	
HS65	3	2	0.95353	1.1e-07	9	11	9	0.00	
HS66	3	3	0.51816	9.2e-08	9	24	18	0.02	
HS67	3	15	-1162.1	2.8e-07	8	10	8	0.00	
HS68	4	3	-0.92043	1.1e-10	15	35	30	0.03	
HS69	4	3	-956.71	6.4e-07	15	36	30	0.05	
HS70	4	2	0.0074985	9.9e-07	18	38	36	0.53	
HS71	4	3	17.014	7.8e-07	7	9	7	0.02	
HS72	4	3	727.68	4.9e-07	15	53	30	0.03	
HS73	4	4	29.894	2.8e-08	8	10	8	0.02	
HS74	4	6	5126.5	5.9e-07	7	14	12	0.00	
HS75	4	6	5174.4	1.0e-11	8	19	14	0.02	
HS76	4	4	-4.6818	1.4e-06	6	8	6	0.00	
HS77	5	3	0.2415	1.7e-08	11	13	13	0.02	
HS78	5	4	-2.9197	1.9e-10	4	6	4	0.00	
HS79	5	4	0.078777	1.1e-09	4	6	4	0.03	
HS80	5	4	0.05395	9.9e-07	5	10	9	0.00	
HS81	5	4	0.05395	2.4e-08	7	14	13	0.03	
HS83	5	4	-30666	4.0e-07	7	9	8	0.02	
HS84	5	4	-5.2803e+06	1.0e-06	11	30	22	0.02	
HS85	5	22	-2.2147	8.3e-07	17	22	24	0.08	
HS86	5	11	-32.349	8.4e-08	10	12	11	0.02	
HS87	6	5	8927.6	6.2e-09	12	25	24	0.03	
HS88	2	2	1.3627	5.2e-11	13	25	24	0.12	
HS89	3	2	1.3627	9.9e-09	30	54	55	0.40	
HS90	4	2	1.3627	1.0e-06	21	33	40	0.38	
HS91	5	2	1.3627	1.3e-08	17	27	28	0.52	
HS92	6	2	1.3627	9.6e-07	21	38	40	0.83	
HS93	6	3	135.08	6.9e-08	8	20	16	0.05	
HS95	6	5	0.015627	4.1e-08	10	16	13	0.03	
HS96	6	5	0.015672	2.9e-07	9	16	11	0.02	
HS97	6	5	4.0713	6.5e-07	13	25	19	0.03	
HS98	6	5	4.6452	1.2e-07	12	20	17	0.03	
HS99	7	3	-8.3108e+08	1.8e-07	5	7	5	0.02	
HS100	7	5	680.63	1.2e-07	7	17	8	0.02	
HS101	7	6	1809.8	2.2e-07	16	27	23	0.08	
HS102	7	6	911.88	9.1e-07	15	22	18	0.08	
HS103	7	6	543.67	5.3e-07	20	38	30	0.12	
HS104	8	6	3.9512	1.2e-08	11	20	20	0.05	
HS105	8	2	1044.6	4.7e-07	10	21	19	0.58	
HS106	8	7	7049.3	2.9e-08	15	32	29	0.03	
HS107	9	7	5055	2.4e-08	8	10	9	0.03	
HS108	9	14	-0.67498	5.5e-07	46	72	85	0.20	
HS109	9	11	5362.1	3.0e-10	11	21	20	0.07	
HS110	10	1	-45.779	1.2e-06	6	13	11	0.03	
HS111	10	4	-47.761	1.2e-06	16	36	32	0.10	
HS112	10	4	-47.761	2.7e-07	12	14	14	0.07	
HS113	10	9	24.306	1.0e-06	9	11	9	0.00	
HS114	10	12	-1768.8	1.8e-08	15	39	30	0.07	
HS116	13	15	97.588	5.2e-09	26	56	51	0.13	
HS117	15	6	32.349	3.9e-07	11	13	11	0.03	

problem	n	m	obj	res	itr	neval	nfact	time(s)	
HS118	15	18	664.82	4.8e-08	15	27	29	0.08	
HS119	16	9	244.9	9.4e-07	12	23	21	0.12	
TOTAL (114)					1296	2321	2091	6.77	
AVERAGE	4	4		3.4e-07	11.4	20.4	18.3	0.06	

6.5 CUTE problems

In this subsection, we report the results for CUTE problems [2]. Our version of CUTE problems is the one obtained in December 8 1994. We choose those problems which have more than 20 variables, more than 20 constraints and analytic second derivatives. If the problem size is variable, we choose the maximum size specified basically. The problem LHAIFAM is excluded because the CUTE interface subroutine behaves abnormally. The problem GROUPING is excluded because the number of equality constraints exceeds that of variables. This selection leaves 164 problems for us. In the following table the mark t means that the problem needed parameter tuning to solve it.

Summary of the results is as follows:

Total number of problems = 164
Total number of succeeded problems = 150
Total number of problems that needed parameter tuning = 18
Total number of failed problems = 14
Average number of variables = 3830
Average number of constraints = 2522
Total number of iterations = 3092

- CUTE problems

problem	n	m	obj	res	itr	neval	nfact	time(s)	
AGG	163	489	-3.5991e+07	7.7e-07	26	28	40	2.7	
AIRPORT	84	43	47953	2.2e-11	25	33	49	2.3	
AUG2D	20200	10001	1.6874e+06	1.5e-07	6	8	18	127.0	
AUG2DC	20200	10001	1.8184e+06	1.1e-13	1	3	1	51.7	
AUG2DCQP	20200	10001	6.4981e+06	2.2e-08	34	41	66	482.0	*t
AUG2DQP	20200	10001	6.237e+06	2.8e-07	36	43	70	525.0	*t
AUG3D	3873	1001	554.07	3.5e-07	4	6	12	7.6	
AUG3DC	3873	1001	771.26	1.4e-15	1	3	1	3.1	
AUG3DCQP	3873	1001	993.36	1.4e-07	14	16	21	14.9	
AUG3DQP	3873	1001	675.24	8.5e-07	14	16	20	13.1	
BIGGSB1	1000	1	0.015323	5.7e-07	15	17	18	1.9	
BLOCKQP1	2005	1002	2.5042	1.1e-06	8	10	9	4.3	
BLOCKQP2	2005	1002	2.5043	4.1e-07	9	11	10	4.6	
BLOCKQP3	2005	1002	2.5013	9.0e-07	8	10	9	4.2	
BLOCKQP4	2005	1002	2.5021	3.5e-09	15	17	20	7.1	
BLOCKQP5	2005	1002	2.5021	1.1e-06	8	10	9	4.2	
CHENHARK	1000	1	-2	1.3e-06	12	14	15	1.6	
CLNLBEAM^{*s}	1503	1001	346.5	1.1e-06	94	174	186	32.9	
CORKSCRW	9006	7001	90.69	1.1e-08	19	33	32	87.5	
COSHFUN	61	21	-0.77326	2.5e-07	18	23	32	0.2	
DALLASL	906	668	-2.026e+05	1.6e-07	19	23	28	6.0	
DALLASM	196	152	-48198	1.3e-07	13	15	15	0.8	
DALLASS	46	32	-32393	1.5e-06	14	17	19	0.2	
DISC2	29	24	1.5625	5.4e-07	23	54	43	0.2	
DITTERT^{*s}	105	71	-1.9846	1.1e-08	7	9	8	0.2	
DIXCHLNV	100	51	8.0923e-27	2.9e-08	36	48	68	4.3	
DTOC1L	14995	9991	125.34	1.3e-07	5	7	6	68.6	
DTOC1NA	1495	991	12.702	9.8e-08	5	7	6	5.7	
DTOC1NB	1495	991	15.938	2.1e-09	5	7	6	5.6	
DTOC1NC	1495	991	24.97	3.9e-07	72	97	140	64.3	
DTOC1ND^{*s}	745	491	13.374	4.9e-07	93	151	183	43.6	*t
DTOC2	5998	3997	0.50865	4.0e-07	5	10	8	17.6	
DTOC3	14999	9999	235.26	1.8e-13	1	3	1	37.5	
DTOC4	14999	9999	2.8685	8.9e-10	3	5	3	49.1	
DTOC5	9999	5000	1.5351	5.4e-09	3	5	4	20.6	
DTOC6	10001	5001	1.3485e+05	4.0e-10	11	17	21	60.4	
EG3	1001	2001	0.22677	1.3e-06	37	57	73	26.1	*t
EIGENA2	110	56	0	1.0e-17	2	6	3	0.4	
EIGENACO	110	56	8.9141e-29	2.9e-16	2	6	3	1.6	
EIGENB2	110	56	18	5.0e-15	2	4	2	0.4	
EIGENBCO	110	56	9	3.8e-16	2	4	2	1.6	
EIGENC2^{*s}	462	232	6.8846	3.8e-09	37	61	69	140.2	
EIGENCCO^{*s}	30	16	0.38828	1.0e-07	30	47	57	0.6	
EIGMAXA	101	102	-1	1.1e-06	13	15	24	0.3	
EIGMAXB	101	102	-0.00096743	7.1e-08	9	11	13	0.2	
EIGMAXC	22	23	-1	3.1e-07	10	12	16	0.1	
EIGMINA	101	102	1	1.1e-06	13	15	24	0.3	
EIGMINB	101	102	0.00096743	1.4e-07	9	11	13	0.2	
EIGMINC	22	23	1	4.8e-07	11	13	18	0.1	
EXPLIN	120	1	-7.2352e+05	1.0e-06	15	17	21	0.1	
EXPLIN2	120	1	-7.2446e+05	1.7e-07	14	16	19	0.1	
EXPQUAD	120	1	-3.626e+06	1.1e-06	9	11	10	0.1	
GAUSSELM	1496	3691	-0.99999	9.9e-07	12	18	15	10.6	
GOFFIN	51	51	6.551e-05	1.0e-07	13	15	31	1.5	
GOULDQP2	699	350	0.00018798	2.6e-07	8	10	14	0.9	
GOULDQP3	699	350	2.0278	1.3e-07	8	14	14	1.4	

problem	n	m	obj	res	itr	neval	nfact	time(s)	
GRIDNETA	13284	6725	304.98	1.3e-06	20	22	31	81.0	
GRIDNETB	13284	6725	143.32	2.1e-14	1	3	1	23.9	
GRIDNETC	7564	3845	161.87	8.2e-07	27	29	44	74.9	
GRIDNETD	7564	3845	570.71	7.3e-07	16	19	25	21.0	
GRIDNETE	7564	3845	206.48	4.2e-07	2	4	2	14.5	
GRIDNETF	7564	3845	243.54	8.3e-08	26	28	41	100.7	
GRIDNETG	60	37	73.449	4.7e-08	8	10	11	0.3	
GRIDNETH	60	37	39.609	1.6e-07	4	6	4	0.2	
GRIDNETI	60	37	40.223	1.0e-07	9	11	11	0.4	
HADAMALS	100	1	813.35	1.4e-06	12	14	18	1.5	
HAGER1	10001	5001	0.88078	9.0e-07	3	5	3	22.9	
HAGER2	10001	5001	0.43208	5.8e-15	1	3	1	15.7	
HAGER3	10001	5001	0.14096	1.1e-14	1	3	1	20.5	
HAGER4	10001	5001	2.7955	2.6e-07	9	11	11	41.3	
HANGING^{*s}	300	181	-620.18	4.0e-08	19	21	33	2.0	
HARKERP2	100	1	43.892	1.1e-06	13	15	16	25.8	
HELSBY	1408	1400	31.97	4.2e-07	24	36	42	7.9	
HS99EXP	31	22	-8.6883e-23	7.4e-08	8	10	10	0.1	
HVYCRASH^{*s}	204	151	2.5536e-07	1.3e-06	51	69	96	2.4	
HYDROELL	1009	1009	-3.5852e+06	1.1e-06	16	18	25	4.3	
HYDROELM	505	505	-3.5818e+06	1.1e-06	16	18	25	1.9	
HYDROELS	169	169	-3.5822e+06	8.5e-07	15	17	24	0.6	
KSIP	20	1002	6.0968e+15	1.2e-06	14	16	20	5.3	*t
LAUNCH	25	29	9.0052	2.3e-07	29	45	51	0.5	
LEAKNET	156	154	8.0464	1.3e-06	24	38	42	0.8	
LINSPANH	97	34	-77	4.2e-07	5	7	5	0.0	
LISWET1	10002	10001	475.19	1.1e-07	16	18	31	92.3	*t
LISWET10	10002	10001	526.3	6.0e-08	17	19	33	91.7	*t
LISWET11	10002	10001	372.49	1.7e-07	15	17	29	81.8	*t
LISWET12	10002	10001	2177.6	8.3e-08	18	20	35	96.5	*t
LISWET2	10002	10001	25.001	1.5e-06	7	9	9	39.8	
LISWET3	10002	10001	25	5.9e-07	9	11	13	50.2	
LISWET4	10002	10001	25	1.7e-10	22	24	39	117.6	
LISWET5	10002	10001	25	3.4e-07	9	11	12	48.5	
LISWET6	10002	10001	25.001	1.4e-06	7	9	9	39.8	
LISWET7	10002	10001	1276.8	1.7e-07	17	19	33	97.5	*t
LISWET8	10002	10001	1237.8	1.4e-07	17	19	33	97.6	*t
LISWET9	10002	10001	2512	5.5e-08	19	21	37	107.8	*t
MADSSCHJ	81	159	-797.28	3.8e-07	13	22	22	9.6	
MAKELA3	21	21	3.3825e-08	8.7e-08	8	10	11	0.1	
MAKELA4	21	41	2.6483e-06	6.6e-08	5	7	5	0.0	
MANNE	1095	731	-0.94019	1.1e-06	94	141	179	26.6	
MINC44^{*s}	311	263	0.002573	2.9e-07	14	19	20	5.1	
MODEL	1542	39	0	8.7e-10	5	7	6	0.2	
MOSARQP1	2500	701	-952.88	1.0e-06	12	14	12	5.7	
MOSARQP2	900	601	-1597.5	6.3e-07	10	12	11	2.2	
NGONE^{*s}	100	1274	-0.63764	8.5e-09	51	67	100	24.2	*t
OPTCNTRL	32	21	550	9.9e-09	13	15	20	0.0	
OPTCTRL3	122	81	2047.8	7.9e-07	3	5	3	0.1	
OPTCTRL6	122	81	2047.8	7.9e-07	3	5	3	0.1	
OPTMASS^{*s}	70	56	-9.669e-13	4.4e-07	3	5	6	0.1	
ORTHREGA	517	257	1664.8	8.4e-08	70	93	139	8.9	
POWELL20^{*s}	1000	1001	5.2146e+07	1.6e-09	42	44	80	12.9	*t
PRODPL0	60	30	58.79	1.0e-09	22	29	37	0.2	
PRODPL1	60	30	35.739	1.4e-07	49	94	90	0.5	
QPCBLEND	83	75	-0.0078415	5.3e-07	18	20	28	0.4	

problem	n	m	obj	res	itr	neval	nfact	time(s)	
QPCBOEI1	384	352	1.1504e+07	7.7e-07	23	25	34	2.8	
QPCBOEI2	143	167	8.1966e+06	1.6e-08	39	41	67	1.8	
QPCSTAIR	467	357	6.2044e+06	9.4e-07	63	65	115	9.6	
QPNBLEND	83	75	-0.0091333	2.2e-08	21	23	34	0.4	
QPNBOEI1	384	352	6.7789e+06	1.3e-06	27	29	43	3.4	
QPNBOEI2	143	167	1.3835e+06	3.9e-08	34	36	56	1.5	
QPNSTAIR	467	357	5.146e+06	7.0e-07	69	71	125	10.4	
QR3DLS	155	1	2.4812e-15	5.3e-09	51	124	99	13.1	
QUDLIN	50	1	-1.25e+05	6.4e-07	13	15	18	0.1	
READING1	10002	5001	-0.15517	1.3e-06	17	27	32	88.9	
READING2	15003	10001	-0.012576	1.5e-06	29	34	57	219.6	
READING3	10002	5002	-0.15255	3.3e-08	19	27	36	98.9	
READING4^{*s}	501	501	-0.28928	5.9e-08	32	48	62	6.3	*t
READING5	5001	5001	0	7.1e-09	7	9	12	19.2	
S368	100	1	-123.05	4.8e-07	32	58	60	17.8	
SINROSNB	1000	1000	201.41	2.2e-07	11	15	15	2.8	
SMBANK	117	65	-7.1293e+06	2.9e-07	16	18	24	0.3	
SMMPSF	720	264	1.0329e+06	3.7e-07	43	65	74	13.3	
SPANHYD	97	34	335.08	4.7e-07	7	9	8	0.1	
SREADIN3	10002	5002	-0.15249	7.7e-07	23	32	44	117.8	
SSEBLIN	194	73	1.6172e+07	2.8e-07	10	12	11	0.2	
SSEBNLN	194	97	1.6171e+07	4.1e-11	19	43	36	0.7	*t
SSNLBEAM^{*s}	33	21	337.77	1.2e-06	51	83	100	0.3	
STATIC3	434	97	-1529.8	1.4e-06	26	29	45	1.7	
STEENBRA	432	109	16958	6.5e-08	12	14	17	6.0	
STEENBRB	468	109	9076.1	5.7e-08	51	53	92	30.2	
STEENBRC	540	127	28482	2.7e-07	42	53	73	36.2	
STEENBRD	468	109	9030.6	5.9e-07	79	143	149	48.7	
STEENBRE	540	127	28529	1.5e-06	33	42	58	29.2	
STEENBRF	468	109	8995.3	5.8e-07	50	64	90	29.6	
STEENBRG	540	127	28268	1.0e-07	53	57	95	46.0	
SVANBERG	5000	5001	8361.4	1.7e-07	23	25	39	76.7	*t
SWOPF	83	93	0.06786	9.5e-08	13	15	19	0.2	
TRAINF	20008	10003	3.1056	7.2e-08	31	33	55	390.1	
TRAINH	20008	10003	12.316	1.3e-07	64	66	119	801.5	
UBH1	18009	12001	1.116	4.9e-10	4	6	5	68.5	
UBH5	20010	14001	1.116	6.4e-08	4	6	4	89.5	
ZIGZAG^{*s}	64	51	3.1618	4.2e-07	32	51	56	0.4	*t
TOTAL (150)					3092	4204	5394	5761.6	
AVERAGE	3830	2522		4.3e-07	20.6	28.0	36.0	38.4	

- failed CUTE problems

problem	n	m
BINSTAR1	257	251
BINSTAR2	157	151
CATENARY^{*s}	99	33
DISCS	36	67
DRUGDIS	3004	2001
DRUGDISE	63	51
HADAMARD^{*s}	65	165
HAIFAL	343	8959
HAIFAM	99	151
HUESTIS	10000	3
JUNKTURN	7000	10010
LUBRIF	500	751
MINPERM	1113	1034
ROTDISC	905	1082
TOTAL (14)		
AVERAGE	1688	1764

References

[1] D.P.Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[2] I.Bongartz, A.R.Conn, N.Gould, and Ph.L.Toint, *CUTE:Constrained and Unconstrained Testing Environment*, Research Report RC 18860, IBM T.J. Watson Research Center, Yorktown , USA,1993.

[3] J.F.Bonnans and C.Pola, *A trust region interior point algorithm for linearly constrained optimization*, Technical Report 1948, INRIA, 1993.

[4] M.G.Breitfield and D.F.Shanno, Preliminary computational experience with modified log-barrier functions for large-scale nonlinear programming, in *Large Scale Optimization*, Kluwer academic publishers, Dordrecht, Boston, London, 1994.

[5] R.H.Byrd, J.C.Gilbert and J.Nocedal, *A trust region method based on interior point techniques for nonlinear programming*, Technical Report OTC 96/02, Optimization Technology Center, Argonne National Laboratory, June, 1996.

[6] R.H.Byrd, M.E.Hribar and J.Nocedal, *An interior point algorithm for large scale nonlinear programming*, Technical Report OTC 97/05, Optimization Technology Center, Argonne National Laboratory, August, 1997.

[7] R.H.Byrd, G.Liu and J.Nocedal, On the local behaviour of an interior point method for nonlinear programming, in *Numerical analysis 1997*, D.F.Griffiths, D.J.Higham and G.A.Watson eds., Longman (1998), pp.37-56.

[8] T.F.Coleman and Y.Li, An interior trust region approach for nonlinear minimization subject to bounds, *SIAM J. on Optimization*, 6 (1996) pp.418-445.

[9] A.R.Conn, N.I.M.Gould and Ph.L.Toint, `LANCELOT`: a Fortran package for large-scale nonlinear optimization (Release A). Springer Verlag, Heiderberg, Berlin, New York, 1992.

[10] J.E.Dennis, Jr., M.Heinkenschloss and L.N.Vicente, *Trust-region interior-point SQP algorithms for a class of nonlinear programming problems*, TR94-45, Dept. of Computational and Applied Mathematics, Rice University, Houston, Texas, USA, 1994 (revised November 1995).

[11] I.S.Duff, and J.K.Reid, *The Multifrontal solution of indefinite sparse symmetric linear systems.*, ACM Transaction on Mathematical Software.Vol.9,No.3 ,302-325,1983.

[12] A.S.El-Bakry, R.A.Tapia, T.Tsuchiya and Y.Zhang, On the formulation and theory of the Newton interior-point method for nonlinear programming, *Journal of Optimization Theory and Applications*, 89 (1996) pp.507-541.

[13] A.V.Fiacco and G.P.McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, SIAM, Philadelphia, 1990.

[14] R.Fletcher, Second order corrections for nondifferentiable optimization, in *Numerical Analysis – Dundee 1981*, G.A.Watson, ed., Lecture Notes in Mathematics 912, Springer-Verlag, Berlin, 1982, pp.85-114.

[15] R.Fletcher, *Practical Methods of Optimization*, Second Edition, John Wiley & Sons, New York, 1987.

[16] L.Grippo, F.Lampariello and S.Lucidi, A nonmonotone line search technique for Newton's method, *SIAM J. on Numerical Analysis*, 23 (1986), pp.707-716.

[17] W.Hock and K.Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Economics and Mathematical Systems 187, Springer-Verlag, Berlin, 1981.

[18] N.Maratos, *Exact penalty function algorithms for finite dimensional and control optimization problems*, Ph.D.Thesis, Imperial College of Science and Technology, University of London, London, U.K., 1978.

[19] H.J.Martinez, Z.Parada and R.A.Tapia, On the characterization of Q-superlinear convergence of quasi-Newton interior-point methods for nonlinear programming, *Bol. Soc. Mat. Mexicana*, Vol.1 (1995), pp.137-148.

[20] D.Q.Mayne and E.Polak, A superlinearly convergent algorithm for constrained optimization problems, *Mathematical Programming Study*, 16 (1982), pp.45-61.

[21] W.Murray, Sequential quadratic programming methods for large-scale problems, *Computational Optimization and Applications*, 7 (1997) pp.127-142.

[22] E.R.Panier and A.L.Tits, Avoiding the Maratos effect by means of a nonmonotone line search I: General constrained problems, *SIAM J. on Numerical Analysis*, 28 (1991), pp.1183-1195.

[23] E.Rothberg and A.Gupta, Efficient sparse matrix factorization on high-peformance workstations-Exploiting the memory hierarchy, *ACM Transactions on Mathematical Software*, 17, No3, (1991), pp313-334.

primal-dual Optimization

[24] H.Yabe and H.Yamashita, Q-superlinear convergence of primal-dual interior point quasi-Newton methods for constrained optimization, *Journal of the Operations Research Society of Japan*, 40 (1997), pp.415-436.

[25] H.Yamashita, *A globally convergent primal-dual interior point method for constrained optimization*, Technical Report, Mathematical Systems, Inc., Tokyo, Japan, April 1992 (revised May 1995).

[26] H.Yamashita and T.Tanabe, A primal-dual interior point trust region method for large scale constrained optimization, *Optimization – Modeling and Algorithms 6*, Cooperative Research Report 73, The Institute of Statistical Mathematics, March (1995), pp.1-25.

[27] H.Yamashita and H.Yabe, A nonmonotone SQP method with global and superlinear convergence properties, *Optimization – Modeling and Algorithms 8*, Cooperative Research Report 84, The Institute of Statistical Mathematics, March (1996), pp.10-29.

[28] H.Yamashita and H.Yabe, Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization, *Mathematical Programming*, 75 (1996), pp.377-397.