A primal-dual interior-point relaxation method
for nonlinear programs

Xin-Wei Liu∗ and Yu-Hong Dai†

Abstract. We prove that the classic logarithmic barrier problem is equivalent to a particular logarithmic
barrier positive relaxation problem with barrier and scaling parameters. Based on the equivalence, a line-
search primal-dual interior-point relaxation method for nonlinear programs is presented. Our method
does not require any primal or dual iterates to be interior-points, which is prominently different from the
existing interior-point methods in the literature. A new logarithmic barrier penalty function dependent
on both primal and dual variables is used to prompt the global convergence of the method, where the
penalty parameter is updated adaptively. Without assuming any regularity condition, it is proved that
our method will terminate at an approximate KKT point of the original problem provided the barrier
parameter tends zero. Otherwise, either an approximate infeasible stationary point or an approximate
singular stationary point of the original problem will be found. Some preliminary numerical results are
reported, including the results for a well-posed problem for which many line-search interior-point methods
were demonstrated not to be globally convergent, a feasible problem for which the LICQ and the MFCQ
fail to hold at the solution and an infeasible problem, and for some standard test problems of the CUTE
collection. These results show that our algorithm is not only efficient for well-posed feasible problems,
but also is applicable for some ill-posed feasible problems and some even infeasible problems.

Key words: nonlinear programming, constrained optimization, interior-point method, logarithmic bar-
rier problem, global convergence

AMS subject classifications. 90C26, 90C30, 90C51

∗Institute of Applied Mathematics, Hebei University of Technology, Tianjin 300401, China. E-mail: math-
lxw@hebut.edu.cn. The research is supported by the Chinese NSF grants (nos. 11671116 and 11271107), the
Major Research Plan of the National Natural Science Foundation of China (no. 91630202), and a grant (no.
A2015202365) from Hebei Natural Science Foundation of China.

†Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China. This
author is supported by the Chinese NSF grants (nos. 11631013, 11331012 and 81173633) and the National Key
Basic Research Program of China (no. 2015CB856000).
1. Introduction

We consider the general nonlinear programs with equality and inequality constraints

\[
\begin{align*}
\text{minimize} \quad & (\min) \quad f(x) \\
\text{subject to} \quad & (\text{s.t.}) \quad h_i(x) = 0, \ i = 1, \ldots, m_e, \\
& c_j(x) \leq 0, \ j = 1, \ldots, m,
\end{align*}
\]

where \(m_e\) and \(m\) are integer numbers, \(x \in \mathbb{R}^n\), \(f, h_i(i = 1, \ldots, m_e)\), and \(c_j(j = 1, \ldots, m)\) are twice continuously differentiable real-valued functions defined on \(\mathbb{R}^n\). Interior-point methods have been among those most efficient methods for nonlinear programs (for example, see \[6, 9, 13, 23, 25\]). Given parameter \(\mu > 0\), interior-point methods for nonlinear program (1.1)–(1.3) usually need to approximately solve the following logarithmic barrier problem

\[
\begin{align*}
\text{min} \quad & f(x) - \mu \sum_{j=1}^m \ln t_j \\
\text{s.t.} \quad & h_i(x) = 0, \ i = 1, \ldots, m_e, \\
& c_j(x) + t_j = 0, \ j = 1, \ldots, m,
\end{align*}
\]

where \(t_j(j = 1, \ldots, m)\) are slack variables for inequality constraints. Some other interior-point methods approximately solve the KKT system of the above logarithmic barrier problem as follows,

\[
\begin{align*}
\nabla f(x) + \sum_{i=1}^{m_e} \lambda_i \nabla h_i(x) + \sum_{j=1}^m s_j \nabla c_j(x) &= 0, \\
t_j > 0, \ s_j > 0, \ t_j s_j - \mu &= 0, \ j = 1, \ldots, m, \\
h_i(x) = 0, \ i = 1, \ldots, m_e; \ c_j(x) + t_j &= 0, \ j = 1, \ldots, m,
\end{align*}
\]

where \(\lambda = (\lambda_i, i = 1, \ldots, m_e) \in \mathbb{R}^{m_e}\), \(s = (s_j, j = 1, \ldots, m) \in \mathbb{R}^m\) are respectively the Lagrangian multiplier vector associated with constraints in (1.5) and (1.6), \(\mu > 0\) is the barrier parameter which can be any number of a decreasing sequence with the limit 0.

During the iterative processes for the logarithmic barrier problem (1.4)–(1.6) and the system (1.7)–(1.9), all iterates must be interior points. That is, there should always have \(t = (t_j, j = 1, \ldots, m) > 0\) and \(s > 0\). This guarantee for iterates to be interior-points may result in the truncation of the step and can impact on global performance of many existing line-search interior-point methods. An analytical counterexample presented in Wächter and Biegler [24] demonstrated that interior-point methods using linearized constraints and interior-point guarantee may fail in converging to any feasible point of a well-posed problem. As \(\mu > 0\) is small enough, an approximate KKT point of problem (1.4)–(1.6) is usually thought to be an approximate KKT point of the original problem.

There are already many interior-point methods in the literature which do not suffer from the failure in [24]. These methods either use trust region techniques for new iterates (such as [6]) or change the system (1.7)–(1.9) and/or its linearized system (e.g. [9, 11, 13, 15, 17, 22, 25]).
Very recently, with help of the augmented Lagrangian function, Dai, Liu and Sun [10] presented a new primal-dual interior-point method for nonlinear programs. The method can produce interior-point iterates without truncation of the step, which is entirely different from the current primal-dual interior-point methods in the literature. It is proved that the method converges to a KKT point of the original problem as the barrier parameter tends to zero. Otherwise, the penalty parameter tends to zero, and the method converges to either an infeasible stationary point or a singular stationary point of the original problem. In particular, the method has the capability to rapidly detect the infeasibility of the problem. It is a very useful property in practice.

In this paper, we firstly prove that the logarithmic barrier problem (1.4)–(1.6) is equivalent to a particular logarithmic barrier positive relaxation problem with barrier and scaling parameters (see problem (2.7)–(2.10) in the next section). That is, we can derive a KKT point of problem (1.4)–(1.6) by solving its equivalent problem (2.7)–(2.10). Based on the equivalence, a primal-dual interior-point relaxation method for nonlinear programs (1.1)–(1.3) is then presented. Our method does not require any primal or dual iterates to be interior-points, which is prominently different from the existing interior-point methods in the literature. This characteristic of our method makes us avert the truncation of any step for guarantee of interior-point iterates.

Our method has some similarity to that in [10]. However, since our method is proposed without using the augmented Lagrangian function, it has more flexibility to the selection of penalty parameter. Moreover, our method in this paper solves general optimization problems with inequality and equality constraints. It is known that the Lagrangian multipliers are dependent on the scaling of constraints. Thus, a scaling parameter is incorporated into the method. The incorporation of this parameter makes our method be robust for some degenerate and difficult problems. Without assuming any regularity condition, it is proved that our method will terminate at an approximate KKT point of the original problem provided the barrier parameter tends to zero. Otherwise, either an approximate infeasible stationary point or an approximate singular stationary point of the original problem will be found. Some preliminary numerical results are reported, including the results for a well-posed problem for which many line-search interior-point methods were demonstrated not to be globally convergent, a feasible problem for which the LICQ and the MFCQ fail to hold at the solution and an infeasible problem, and for some standard test problems of the CUTE collection. These results show that our algorithm is not only efficient for well-posed feasible problems, but also is applicable for some ill-posed feasible problems and some even infeasible problems.

This article is organized as follows. In section 2, we describe a particular logarithmic barrier positive relaxation problem, and prove its equivalence to the logarithmic barrier problem (1.4)–(1.6). Our primal-dual interior-point relaxation method for nonlinear program (1.1)–(1.3) is presented in section 3. In section 4, we show the global convergence results of our method. Some preliminary numerical results are reported in section 5. We conclude the paper in section 6.

Throughout the article, we use standard notations from the literature. A letter with subscript \( k \) (\( l \)) is related to the \( k \)th (\( l \)th) iteration, the subscript \( j \) (\( i \)) indicates the \( j \)th (\( i \)th) component of
Proof. Due to (2.1), and \( \sqrt{\mu} \) depend on the parameters \( \mu > 0 \). For given Lemma 2.1, \( z \) are column vectors, and \( t, s \in \mathbb{R}^n \), and \( \theta_k \) is a monotonically decreasing function on \( \mu > 0 \). Thus, there exists a constant \( M > 0 \) such that \( |\theta_k| \leq M |\mu| \) for all \( k \) large enough, and \( \theta_k = o(t_k) \) indicates that \( |\theta_k| \leq \epsilon_k |t_k| \) for all \( k \) large enough with \( \lim_{k \to 0} \epsilon_k = 0 \). If it is not specified, \( I \) is an identity matrix whose order may be showed in the subscript or be clear in the context, \( \| \cdot \| \) is the Euclidean norm. Some unspecified notations may be identified from the context.

2. A logarithmic barrier positive relaxation problem

Suppose that \( \mu > 0 \) and \( \tau > 0 \) are fixed constants. Let us consider problem (1.4)–(1.6) and its KKT system (1.7)–(1.9). For \( x \in \mathbb{R}^n \), \( t \in \mathbb{R}^m \), and \( s \in \mathbb{R}^m \), define \( z \in \mathbb{R}^m \) and \( y \in \mathbb{R}^m \) by components

\[
\begin{align*}
    z_j &= (\sqrt{(\tau s_j - t_j)^2 + 4\tau \mu} - (\tau s_j - t_j))/2, \\
    y_j &= (\sqrt{(\tau s_j - t_j)^2 + 4\tau \mu + (\tau s_j - t_j)})/2,
\end{align*}
\]

where \( j = 1, \ldots, m \). That is, both \( z : \mathbb{R}^{2m} \to \mathbb{R}^m \) and \( y : \mathbb{R}^{2m} \to \mathbb{R}^m \) are functions on \( (t, s) \) and depend on the parameters \( \mu \) and \( \tau \). Obviously, \( z_j y_j = \tau \mu \) for \( j = 1, \ldots, m \). Moreover, we have following simple but important results on \( z \) and \( y \).

**Lemma 2.1** For given \( \mu > 0 \) and \( \tau > 0 \), \( z_j \) and \( y_j \) are defined by (2.7). Then

\[
t_j > 0, \ s_j > 0, \ t_j s_j = \mu \quad \text{if and only if} \quad z_j - t_j = 0, \ y_j - \tau s_j = 0.
\]

**Proof.** Due to (2.1), \( z_j > 0 \) and \( y_j > 0 \) for any \( t \in \mathbb{R}^m \) and \( s \in \mathbb{R}^m \). If \( z_j - t_j = 0 \), then \( t_j > 0 \) and

\[
\sqrt{(\tau s_j - t_j)^2 + 4\tau \mu} = t_j + \tau s_j.
\]

Thus, \( t_j s_j = \mu \), which implies \( s_j > 0 \). Similarly, one can prove that the results hold provided \( y_j - \tau s_j = 0 \).

Conversely, suppose \( t_j > 0 \) and \( s_j > 0 \). If either \( z_j - t_j \neq 0 \) or \( y_j - \tau s_j \neq 0 \), then \( t_j s_j \neq \mu \), which shows that \( t_j s_j = \mu \) implies \( z_j - t_j = 0 \) and \( y_j - \tau s_j = 0 \). \( \square \)

**Lemma 2.2** For \( j = 1, \ldots, m \), let \( z_j \) and \( y_j \) be defined by (2.1). (1) \( z_j \) and \( y_j \) are differentiable on \( (t, s) \), and

\[
\begin{align*}
    \nabla_t z_j &= \frac{z_j}{z_j + y_j} e_j, & \nabla_t y_j &= -\frac{y_j}{z_j + y_j} e_j, \\
    \nabla_s z_j &= -\tau \frac{z_j}{z_j + y_j} e_j, & \nabla_s y_j &= \frac{y_j}{z_j + y_j} e_j,
\end{align*}
\]

where \( e_j \in \mathbb{R}^m \) is the \( j \)-th coordinate vector.

(2) \( z_j \) is a monotonically increasing function on \( t_j \), and is a monotonically decreasing function on \( s_j \).

(3) \( y_j \) is a monotonically decreasing function on \( t_j \), and is a monotonically increasing function on \( s_j \).

(4) Both \( z_j \) and \( y_j \) are decreasing as \( \mu > 0 \) becomes smaller.
Proof. (1) Due to $z_jy_j = \tau \mu,$
\[
y_j \nabla t z_j + z_j \nabla t y_j = 0.
\] (2.5)
Note that $z_j - y_j = t_j - \tau s_j.$ Thus, $\nabla t z_j - \nabla t y_j = e_j.$ Using (2.5), one has (2.3) immediately.

We can prove (2.4) similarly.

(2) By (1), $\frac{\partial z_j}{\partial t_j} > 0$ and $\frac{\partial z_j}{\partial s_j} < 0.$ The results can be derived straightforward.

(3) The results follows immediately since $\frac{\partial y_j}{\partial t_j} < 0$ and $\frac{\partial y_j}{\partial s_j} > 0.$

(4) Due to $z_jy_j = \tau \mu$ and $z_j - y_j = t_j - \tau s_j,$ one has
\[
y_j \frac{dz_j}{d\mu} + z_j \frac{dy_j}{d\mu} = \tau, \quad \text{and} \quad \frac{dz_j}{d\mu} - \frac{dy_j}{d\mu} = 0.
\]
Thus,
\[
\frac{dz_j}{d\mu} = \frac{dy_j}{d\mu} = \frac{\tau}{z_j + y_j} > 0,
\] (2.6)
which shows that both $z_j$ and $y_j$ are monotonically increasing on $\mu.$

Throughout this article, we denote $z := z(t, s; \mu, \tau)$ and $y := y(t, s; \mu, \tau)$ which are defined by (2.1) being functions on $(t, s).$ The next theorem provides a firm foundation for the development of our method.

Theorem 2.3 Let $((x^*, t^*), (\lambda^*, s^*))$ be the KKT pair of problem (1.4)–(1.6) and $(x^*, t^*, \lambda^*, s^*)$ satisfy the system (1.7)–(1.9), where $\lambda^* \in \mathbb{R}^{m_{ce}}$ and $s^* \in \mathbb{R}^m$ are respectively the Lagrangian multipliers of constraints (1.5) and (1.6). Then $((x^*, t^*, s^*), (\lambda^*, s^*))$ are the KKT pair of the following problem

\[
\min_{x, t, s} \quad f(x) - \mu \sum_{j=1}^m \ln z_j
\] (2.7)
\[
s.t. \quad h_i(x) = 0, \quad i = 1, \ldots, m_c,
\] (2.8)
\[
c_j(x) + t_j = 0, \quad j = 1, \ldots, m,
\] (2.9)
\[
z_j - t_j = 0, \quad j = 1, \ldots, m.
\] (2.10)

That is, $\lambda^* \in \mathbb{R}^{m_{ce}}, s^* \in \mathbb{R}^m$ and $s^* \in \mathbb{R}^m$ are the corresponding Lagrangian multipliers of constraints (2.8), (2.9) and (2.10), respectively.

Conversely, if $((x^*, t^*, s^*), (\lambda^*, \beta^*, \nu^*))$ are the KKT pair of the problem (2.7), (2.10), where $\lambda^* \in \mathbb{R}^{m_{ce}}, \beta^* \in \mathbb{R}^m$ and $\nu^* \in \mathbb{R}^m$ are the associated Lagrangian multipliers of constraints (2.8), (2.9) and (2.10), then $\beta^* = \nu^* = y^*(t^*, s^*; \mu, \tau) = s^*$ and $(x^*, t^*, \lambda^*, s^*)$ satisfies the system (1.7)–(1.9). Thus, $((x^*, t^*), (\lambda^*, s^*))$ is the KKT pair of problem (1.4)–(1.6).

Proof. Due to Lemma 2.1, the KKT conditions of problem (1.4)–(1.6) can be written as follows:
\[
\nabla f(x^*) + \sum_{i=1}^{m_c} \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^m s_j^* \nabla c_j(x^*) = 0,
\] (2.11)
\[
h_i(x^*) = 0, \quad i = 1, \ldots, m_c;
\] (2.12)
\[
c_j(x^*) + t_j^* = 0, \quad j = 1, \ldots, m,
\] (2.13)
where \( z^* = z(t^*, s^*; \mu, \tau) \) and \( y^* = y(t^*, s^*; \mu, \tau) \).

Using Lemma 2.2, we can derive the following KKT conditions of problem (2.7)–(2.10):

\[
\nabla f(x^*) + \sum_{i=1}^{m_e} \lambda^*_i \nabla h_i(x^*) + \sum_{j=1}^{m} \beta^*_j \nabla c_j(x^*) = 0, \tag{2.14}
\]

\[
\beta^* - \sum_{j=1}^{m} \frac{\mu + y^*_j \nu^*_j}{z_j^* + y_j^*} e_j = 0, \tag{2.15}
\]

\[
\sum_{j=1}^{m} \tau \frac{\mu - z^*_j \nu^*_j}{z_j^* + y_j^*} e_j = 0, \tag{2.16}
\]

\[
h_i(x^*) = 0, \ i = 1, \ldots, m_e, \tag{2.17}
\]

\[
c_j(x^*) + t^*_j = 0, \ j = 1, \ldots, m, \tag{2.18}
\]

\[
z^*_j - t^*_j = 0, \ j = 1, \ldots, m, \tag{2.19}
\]

where \( \lambda^*_i (i = 1, \ldots, m_e) \), \( \beta^*_j (j = 1, \ldots, m) \) and \( \nu^*_j (j = 1, \ldots, m) \) are the Lagrangian multipliers of constraints (2.8), (2.9) and (2.10), respectively.

For \( j = 1, \ldots, m, \) \( z^*_j \nu^*_j = \mu \) and \( t^*_j s^*_j = \mu \) due to (2.16) and (2.19). Thus, \( \nu^*_j = s^*_j \) for all \( j \) and \( \beta^* = \nu^* \) by (2.15). Since \( z^*_j y^*_j = \tau \mu, \) \( y^*_j = y^*_j / \tau \) for \( j = 1, \ldots, m. \) Thus, any solution of the system (2.14)–(2.19) satisfies the system (2.11)–(2.13). Similarly, we can prove that any solution of the system (2.11)–(2.13) also solves the system (2.14)–(2.19). Hence, the result follows immediately from the equivalence between the systems (2.11)–(2.13) and (2.14)–(2.19).

Since \( z \) is a function on \((t, s)\), problem (2.7)–(2.10) is a nonlinear programming problem on \((x, t, s)\) with equality constraints. Although the logarithmic barrier positive relaxation problem (2.7)–(2.10) has a similar form to the classic logarithmic barrier problem (1.4)–(1.6), it is essentially distinguished from problem (1.4)–(1.6) in that the relaxation problem does not require \( t \) and \( s \) to be positive. This important characteristic makes us free to truncate the step for guarantee of interior-point iterates, a technique generally used by the existing interior-point methods in the literature.

3. A primal-dual interior-point relaxation algorithm

Our algorithm consists of the inner algorithm and the outer algorithm. In the inner algorithm, we attempt to find an approximate KKT point of the logarithmic barrier positive relaxation problem (2.7)–(2.10) for any given \( \mu > 0 \) and \( \tau > 0. \) In the outer algorithm, we update parameters \( \mu \) and \( \tau \) and select the initial value of penalty parameter \( \rho \) of the merit function according to the information of the solution derived from the inner algorithm.

Throughout the article, we denote \( v := (x, t, s) \in \mathbb{R}^{n+2m} \) and functions \( F : \mathbb{R}^{n+2m} \to \mathbb{R}, \)
\[
C : \mathbb{R}^{n+2m} \to \mathbb{R}^{m_e+2m},
\]
\[
F(v) := f(x) - \mu \sum_{j=1}^{m} \ln z_j, \ C(v) := (h(x), e(x) + t, z - t),
\]
where we ignore the parameters $\mu$ and $\tau$ for simplicity of statement. Then, due to Lemma 2.2

$$\nabla F(v) = \left( \nabla f(x), -\sum_{j=1}^{m} \frac{\mu}{z_j + y_j} e_j, \sum_{j=1}^{m} \frac{\tau \mu}{z_j + y_j} e_j \right),$$

$$\nabla C(v) = \begin{pmatrix} \nabla h(x) & \nabla c(x) & 0 \\ 0 & I_m & -(Z + Y)^{-1}Y \\ 0 & 0 & -\tau(Z + Y)^{-1}Z \end{pmatrix},$$

where $e_j \in \mathbb{R}^m$ is the $j$-th coordinate vector in $\mathbb{R}^m$, $I_m$ is the order-$m$ identity matrix, $Z = \text{diag}(z)$ and $Y = \text{diag}(y)$.

Let

$$L(v, w) = f(x) - \mu \sum_{j=1}^{m} \ln z_j + \sum_{i=1}^{m_c} \lambda_i h_i(x) + \sum_{j=1}^{m} \beta_j (c_j(x) + t_j) + \sum_{j=1}^{m} \nu_j (z_j - t_j)$$

be the Lagrangian function of problem (2.7)–(2.10), where $w = (\lambda, \beta, \nu) \in \mathbb{R}^{m_c + 2m}$. Note $z_j y_j = \tau \mu$, $j = 1, \ldots, m$. By Lemma 2.2 its Hessian has the form

$$\nabla^2_{vv} L(v, w) = \begin{bmatrix} H(x, \lambda, \beta) & 0 \\ 0 & \mu \sum_{j=1}^{m} \frac{(\tau \nu_j + z_j) + (\tau \nu_j - y_j)}{(z_j + y_j)^2} e_j e_j^T \\ 0 & -\tau \mu \sum_{j=1}^{m} \frac{(\tau \nu_j + z_j) + (\tau \nu_j - y_j)}{(z_j + y_j)^2} e_j e_j^T \end{bmatrix},$$

where $H(x, \lambda, \beta) = \nabla^2 f(x) + \sum_{i=1}^{m_c} \lambda_i \nabla^2 h_i(x) + \sum_{j=1}^{m} \beta_j \nabla^2 c_j(x)$. If we take $\nu = y/\tau$ in $w$, then

$$\nabla^2_{vv} L(v, w) = \begin{bmatrix} H(x, \lambda, \beta) & 0 \\ 0 & \sum_{j=1}^{m} \frac{\mu}{(z_j + y_j)^2} e_j e_j^T \\ 0 & -\sum_{j=1}^{m} \frac{\tau \mu}{(z_j + y_j)^2} e_j e_j^T \end{bmatrix} - \begin{bmatrix} 0 \\ \sum_{j=1}^{m} \frac{\tau \mu}{(z_j + y_j)^2} e_j e_j^T \\ \sum_{j=1}^{m} \frac{\tau^2 \mu}{(z_j + y_j)^2} e_j e_j^T \end{bmatrix}.$$

3.1. The subproblems for search direction. Suppose $v_k := (x_k, t_k, s_k)$ be the current iterate and $w_k := (\lambda_k, \beta_k, \nu_k)$ be the corresponding estimate of the Lagrangian multiplier. The classic SQP approach for problem (2.7)–(2.10) solves the quadratic programming (QP) subproblem

$$\min \nabla F(v_k)^T d + \frac{1}{2} d^T Q(v_k, w_k) d \quad \text{s.t.} \ C(v_k) + \nabla C(v_k)^T d = 0,$$

where $Q(v_k, w_k)$ is some positive definite approximation to the Hessian $\nabla^2_{vv} L(v_k, w_k)$. Note that, even though $\nabla C(x_k)$ is of full column rank, the constraint system (3.3) may still have a unbounded solution since some diagonal element in $(Z_k + Y_k)^{-1}Z_k$ may be close to zero. In order to keep the solution $d_k$ bounded in our algorithm, we introduce the well-behaved null-space technology to the constraint system (for example, see [3] [5] for trust-region methods and [15] [16] for line-search methods). Moreover, $Q(v_k, w_k)$ is selected to have the same form as $\nabla^2_{vv} L(v_k, w_k)$.
in (3.1), where $H(x_k, \lambda_k, \beta_k)$ is replaced by a positive definite approximation $B_k \in \mathbb{R}^{n \times n}$ to $H(x_k, \lambda_k, \beta_k)$. Thus, $Q(v_k, w_k)$ is an approximation to $\nabla^2_{v_k}L(v_k, w_k)$ and

$$Q(v_k, w_k) = \begin{bmatrix} B_k & 0 \\ 0 & \sum_{j=1}^m \frac{\mu}{(z_{kj}+y_{kj})^2} e_j e_j^T \end{bmatrix} - \sum_{j=1}^m \frac{\tau \mu}{(z_{kj}+y_{kj})^2} e_j e_j^T \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} \sum_{j=1}^m \frac{\tau \mu}{(z_{kj}+y_{kj})^2} e_j e_j^T \\ \sum_{j=1}^m \frac{\tau \mu}{(z_{kj}+y_{kj})^2} e_j e_j^T \end{bmatrix}.$$

Motivated by above arguments, we firstly approximately solve the subproblem

$$\min q_k^N(d; \rho) := \frac{1}{2} \rho d^T Q(v_k, w_k) d + \|C(v_k) + \nabla C(v_k)^T d\| \quad \text{(3.4)}$$

$$\text{s.t. } \|Rd\| \leq \xi \|R^{-1} \nabla C(v_k)C(v_k)\| \quad \text{(3.5)}$$

where $\xi > 1$ is a constant, $\rho > 0$ is the current value of the penalty parameter used in the merit function (see the next subsection), $R = \text{diag}(1, \ldots , 1, \tau, \ldots , \tau) \in \mathbb{R}^{(n+2m) \times (n+2m)}$ with 1’s of number $(n + m)$ and $\tau$’s of number $m$, respectively. Let $p_k \in \mathbb{R}^{n+2m}$ be the solution. Then our search direction $d_k$ is generated by the null-space QP subproblem

$$\min q_k(d) := \nabla F(v_k)^T d + \frac{1}{2} d^T Q(v_k, w_k) d \quad \text{(3.6)}$$

$$\text{s.t. } \nabla C(v_k)^T (d - p_k) = 0. \quad \text{(3.7)}$$

Apparently, $q_k(d_k) \leq q_k(p_k)$. In particular, if $C(v_k) = 0, p_k = 0$ and $q_k(d_k) \leq 0$.

**Lemma 3.1** Assume $\nabla C(v_k)C(v_k) \neq 0$. Let $p_k$ be a solution of subproblem (3.4)–(3.5). If $Q(v_k, w_k)$ is positive semi-definite,

$$\|C(v_k)\| - q_k^N(p_k; \rho) \geq \frac{1}{2} \min\{1, \eta_k\} \left( \frac{\|R^{-1} \nabla C(v_k)C(v_k)\|^2}{\|C(v_k)\|^2} \|C(v_k)\| \right) \left( \|R^{-1} \nabla C(v_k)C(v_k)\|^2 \right)$$

$$- \rho \|C(v_k)\|^2 \frac{C(v_k)^T \nabla C(v_k)^T R^{-2} Q(v_k, w_k) R^{-2} \nabla C(v_k) C(v_k)}{\|R^{-1} \nabla C(v_k) C(v_k)\|^2}, \quad \text{(3.8)}$$

where $\eta_k = \|R^{-1} \nabla C(v_k) C(v_k)\|^2 / \|\nabla C(v_k)^T R^{-2} \nabla C(v_k) C(v_k)\|^2$. Thus,

$$\|C(v_k)\| - q_k^N(p_k; \rho) \geq \frac{1}{4} \min\{1, \eta_k\} \frac{\|R^{-1} \nabla C(v_k) C(p_k)\|^2}{\|C(v_k)\|^2} \quad \text{(3.9)}$$

provided $\rho \|C(v_k)\| \leq 1/(2\lambda_{\max}(R^{-1}Q(v_k, w_k)R^{-1}))$, where $\lambda_{\max}(R^{-1}Q(v_k, w_k)R^{-1})$ is the maximal eigenvalue of $R^{-1}Q(v_k, w_k)R^{-1}$.

**Proof.** Since $p_k^C = -\min\{1, \eta_k\} R^{-2} \nabla C(v_k) C(v_k)$ is feasible to the subproblem (3.4)–(3.5),

$$q_k^N(p_k; \rho) \leq q_k^N(p_k^C; \rho).$$
Thus, \( \|C(v_k)\| - q_k^N(p_k; \rho) \geq \|C(v_k)\| - q_k^N(p_k^C; \rho) \). Due to the positive semi-definiteness of \( Q(v_k, w_k) \),

\[
\|C(v_k)\| - q_k^N(p_k; \rho) \\
\geq \|C(v_k)\| - \|C(v_k) + \nabla C(v_k)^T p_k^C\| \\
- \frac{1}{2} \rho \min\{1, \eta_k\} C(v_k)^T \nabla C(v_k)^T R^{-2} Q(v_k, w_k) R^{-2} \nabla C(v_k) C(v_k).
\]

By Lemma 2.1 of Liu and Yuan [17],

\[
\|C(v_k)\| - \|C(v_k) + \nabla C(v_k)^T p_k^C\| \geq \frac{1}{2} \min\{1, \eta_k\} \|R^{-1} \nabla C(v_k) C(v_k)\|^2 / \|C(v_k)\|.
\]

The result (3.8) follows immediately.

If \( 2\rho \lambda_{\max}(R^{-1} Q(v_k, w_k) R^{-1}) \|C(v_k)\| \leq 1 \), then

\[
\rho \|C(v_k)\|^2 C(v_k)^T \nabla C(v_k)^T R^{-2} Q(v_k, w_k) R^{-2} \nabla C(v_k) C(v_k) \leq \frac{1}{2} \|C(v_k)\|^2.
\]

Thus, (3.9) is derived from (3.8).

\[\Box\]

### 3.2. The merit function.

The merit function plays an important role in prompting the global convergence of the algorithm. For the logarithmic barrier positive relaxation problem (2.7) – (2.10), we introduce the merit function

\[
\phi(v; \rho) = \rho (f(x) - \mu \sum_{j=1}^{m} \ln z_j) + \|(h(x), c(x) + t, z - t)\|,
\]

where \( \rho > 0 \) is a penalty parameter, parameters \( \mu \) and \( \tau \) are ignored for simplicity of statement. It is a function on \((x, t, s)\) (where \( x \in \mathbb{R}^n \) and \( t \in \mathbb{R}^m \) are primal variables and \( s \in \mathbb{R}^m \) is a dual vector), thus it is essentially different from that used in some existing primal-dual interior-point methods such as [8, 15, 17]. It is noted that some methods based on augmented Lagrangian have introduced merit functions with primal and dual variables, for example, see [10, 11, 12, 26]. However, they generated their search directions by different systems and subproblems and shared different motivations with our algorithm. In view of previous notations in this section, it can also be written as

\[
\phi(v; \rho) = \rho F(v) + \|C(v)\|.
\]

**Lemma 3.2** Given parameters \( \mu > 0 \) and \( \tau > 0 \). For any \( \rho \geq 0 \), \( v \in \mathbb{R}^{n+2m} \), \( d \in \mathbb{R}^{n+2m} \) and \( d \neq 0 \), the directional derivative \( \phi'(v, d; \rho) \) of function \( \phi(v; \rho) \) at \( v \) along \( d \) exists, and

\[
\phi'(v, d; \rho) \leq \pi(v, d; \rho), \tag{3.10}
\]

where \( \pi(v, d; \rho) = \rho \nabla F(v)^T d + \chi(v, d) \) and \( \chi(v, d) = \|C(v) + \nabla C(v)^T d\| - \|C(v)\| \).

**Proof.** If \( C(v) \neq 0 \), \( \phi(v; \rho) \) is differentiable at \( v \). Thus, \( \phi'(v, d; \rho) \) exists for any \( d \neq 0 \). For \( v \) such that \( C(v) = 0 \), using the definition of the directional derivative (for example, see (A.14)
Then the result follows immediately from (3.11). Therefore, \( \phi'(v, d; \rho) \) exists for every \( v \in \mathbb{R}^{n+2m}, \), \( d \in \mathbb{R}^{n+2m} \) and \( d \neq 0 \).

Since
\[
\phi'(v, d; \rho) = \lim_{\alpha \downarrow 0} \frac{\phi(v + \alpha d; \rho) - \phi(v; \rho)}{\alpha}
= \rho \nabla F(v)^T d + \lim_{\alpha \downarrow 0} \frac{||C(v) + \alpha C(v)^T d + o(\alpha)|| - ||C(v)||}{\alpha}
\leq \rho \nabla F(v)^T d + ||C(v) + \nabla C(v)^T d|| - ||C(v)|| + \lim_{\alpha \downarrow 0} \frac{||o(\alpha)||}{\alpha}
= \rho \nabla F(v)^T d + ||C(v) + \nabla C(v)^T d|| - ||C(v)||,
\]
(3.10) follows immediately.

The following result shows that \( d_k \) generated by subproblem (3.6)–(3.7) can be a descent direction of the merit function \( \phi(v; \rho) \), provided \( \rho \) is suitably selected.

**Lemma 3.3** For given \( \mu > 0 \) and \( \tau > 0 \), suppose that \( d_k \) is a solution of the QP subproblem (3.6)–(3.7) at \( v_k \). Let \( \phi'_k(d_k; \rho) = \phi'(v_k, d_k; \rho), \) \( \delta \in (0, 1) \) is a constant. If \( \rho \) is small enough, then
\[
\phi'_k(d_k; \rho) \leq (1 - \delta)(q^N_k(p_k; \rho) - ||C(v_k)||) - \frac{1}{2} \rho d_k^T Q(v_k, w_k)d_k.
\]
In particular, \( \phi'_k(d_k; \rho) \leq -\frac{1}{2} \rho d_k^T Q(v_k, w_k)d_k \) for any \( \rho > 0 \) provided \( ||C(v_k)|| = 0 \).

**Proof.** Due to Lemma 3.2, \( \phi'_k(d_k; \rho) \leq \rho \nabla F(v_k)^T d_k + \chi(v_k, d_k) \). Thus,
\[
\phi'_k(d_k; \rho) \leq \rho \nabla F(v_k)^T p_k - \frac{1}{2} \rho d_k^T Q(v_k, w_k)d_k + q^N_k(p_k; \rho) - ||C(v_k)||
\]
(3.11)
since \( \nabla F(v_k)^T d_k + \frac{1}{2} d_k^T Q(v_k, w_k)d_k \leq \nabla F(v_k)^T p_k + \frac{1}{2} p_k^T Q(v_k, w_k)p_k \). If \( ||C(v_k)|| = 0 \), \( p_k = 0 \) and \( q^N_k(p_k; \rho) = 0 \), thus, \( \phi'_k(d_k; \rho) \leq -\frac{1}{2} \rho d_k^T Q(v_k, w_k)d_k \). Otherwise, by Lemma 3.1 one can take \( \rho \) small enough such that \( q^N_k(p_k; \rho) - ||C(v_k)|| < 0 \) and
\[
\rho \nabla F(v_k)^T p_k + \delta(q^N_k(p_k; \rho) - ||C(v_k)||) \leq 0.
\]
Then the result follows immediately from (3.11).

**Lemma 3.4** Given \( \mu > 0 \) and \( \tau > 0 \). For any \( j = 1, \ldots, m, z_{k+1,j} \geq t_{k+1,j} \) if either \( t_{k+1,j} \leq 0 \) or \( t_{k+1,j} > 0 \) but \( s_{k+1,j} \leq \frac{\mu}{t_{k+1,j}} \), where \( z_{k+1,j} = \tau z_{j}(t_{k+1}, s_{k+1}; \mu, \tau) \) is given by (2.7).

**Proof.** If \( t_{k+1,j} \leq 0 \), then \( z_{k+1,j} - t_{k+1,j} > 0 \) since \( z_{k+1,j} > 0 \). Now we consider the case \( t_{k+1,j} > 0 \). If \( \tau s_{k+1,j} + t_{k+1,j} \leq 0 \), one has \( s_{k+1,j} < 0 < \frac{\mu}{t_{k+1,j}} \); else \( \tau s_{k+1,j} + t_{k+1,j} > 0 \), \( z_{k+1,j} \geq t_{k+1,j} \) if and only if
\[
\sqrt{(\tau s_{k+1,j} - t_{k+1,j})^2 + 4\tau \mu} \geq \tau s_{k+1,j} + t_{k+1,j},
\]
which is equivalent to \( t_{k+1,j}s_{k+1,j} \leq \mu \). The result follows immediately since \( t_{k+1,j} > 0 \).
3.3. Our algorithm. Similar to the existing primal-dual interior-point methods, our algorithm consists of the inner algorithm and the outer algorithm, where the inner algorithm tries to find an approximate KKT point of the logarithmic barrier problem, while the outer algorithm updates the parameters by the information derived from the inner algorithm.

For convenience of statement, we denote

\[ r(v_k, \lambda_k; \mu, \tau) = \begin{bmatrix} \nabla f(x_k) + \nabla h(x_k)\lambda_k + \nabla c(x_k)s_k \\ C(v_k) \end{bmatrix}, \]

\[ g(v_k; \mu, \tau) = \frac{1}{\|C(v_k)\|} \begin{bmatrix} \nabla h(x_k)h(x_k) + \nabla c(x_k)(z_k - t_k) \\ c(x_k) + t_k - (z_k - t_k) \\ Z_k(z_k - t_k) \end{bmatrix}, \]

where \( v_k = (x_k, t_k, s_k), \) \( z_k = z(t_k, s_k; \mu, \tau), \) \( C(v_k) = (h(x_k), c(x_k) + t_k, z_k - t_k), \) \( Z_k = \text{diag}(z_k). \) The following proposition provides the terminating conditions for our algorithm.

**Proposition 3.5** For any given \( \mu > 0 \) and \( \tau > 0, \) let \( \{v_k\} \) be any sequence convergent to the point \( v^*, \) where \( v_k = (x_k, t_k, s_k) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^m, \) \( v^* = (x^*, t^*, s^*). \) Let \( \epsilon > 0 \) be any given small scalar.

1. If \((v^*, w^*)\) is a KKT pair of problem \((2.7) - (2.10), \) where \( w^* = (\lambda^*, \beta^*, \nu^*) \in \mathbb{R}^{mc} \times \mathbb{R}^m \times \mathbb{R}^m \) is the corresponding Lagrangian multiplier, then there exists a \( \lambda_k \in \mathbb{R}^{mc} \) such that \( \|r(v_k, \lambda_k; \mu, \tau)\|_{\infty} \leq \epsilon \) for every sufficiently large \( k. \)

2. If \( z_k \geq t_k \) for all \( k \geq 0 \) and

\[ \lim_{k \to \infty} R^{-1}C(v_k)C(v_k)/\|C(v_k)\| = 0, \] (3.12)

then \( \|g(v_k; \mu, \tau)\|_{\infty} \leq \epsilon \) for every sufficiently large \( k. \)

**Proof.** (1) If \((v^*, w^*)\) is a KKT pair of problem \((2.7) - (2.10), \) \((v^*, w^*)\) satisfies the KKT conditions \((2.14) - (2.19). \) Thus, \( \beta^* = \nu^* = s^*, \) and \( t^* > 0, \) \( t_j^* s_j^* = \mu \) for \( j = 1, \ldots, m. \) It follows from \((2.14) \) and \((2.18), (2.19) \) that \( x^*, t^*, \lambda^*, \) and \( s^* \) satisfy \((1.7) - (1.9). \) Hence, \( \|r_1(v^*, \lambda^*; \mu, \tau)\|_{\infty} = 0. \) The result follows immediately from the continuity of \( \nabla f(x), \nabla h(x), \nabla c(x), \) and \( C(v; \mu, \tau). \)

2. Due to \((3.12) \), one has

\[ \lim_{k \to \infty} \nabla h(x_k)h(x_k)/\|C(v_k)\| + \nabla c(x_k)(c(x_k) + t_k)/\|C(v_k)\| = 0, \] (3.13)

\[ \lim_{k \to \infty} (c(x_k) + t_k)/\|C(v_k)\| - (Z_k + Y_k)^{-1}Y_k(z_k - t_k)/\|C(v_k)\| = 0, \] (3.14)

\[ \lim_{k \to \infty} - (Z_k + Y_k)^{-1}Z_k(z_k - t_k)/\|C(v_k)\| = 0. \] (3.15)

By \((3.14) \) and \((3.15), \) \( \lim_{k \to \infty} (c(x_k) + t_k)/\|C(v_k)\| - (z_k - t_k)/\|C(v_k)\| = 0. \) Thus, \((3.13) \) implies

\[ \lim_{k \to \infty} \frac{1}{\|C(v_k)\|} [\nabla h(x_k)h(x_k) + \nabla c(x_k)(z_k - t_k)] = 0. \]
Furthermore, if (3.15) holds, \( \lim_{k \to \infty} Z_k(z_k - t_k) / \|C(v_k)\| = 0 \). Hence, the result is proved. \( \square \)

Now we are ready to present our primal-dual interior-point relaxation algorithm for problem (1.1)–(1.3).

**Algorithm 3.6** *(A primal-dual interior-point relaxation algorithm for problem (1.1)–(1.3))*

**Step 1** Given \((x_0, t_0, s_0) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m, B_0 \in \mathbb{R}^{n \times n}, \mu_0 > 0, \tau_0 > 0, \rho_0 > 0, \delta \in (0, 1), \sigma \in (0, 1), \epsilon > 0 \). Set \( l := 0 \).

**Step 2** While \( \mu_l > \epsilon \) and \( \tau_l > \epsilon \), start the inner algorithm. Otherwise, stop the algorithm.

**Step 2.0** Let \((x_l, t_l, s_l)\), \(B_l = B_l\), \(\rho_l = \rho_l\), \(\mu = \mu_l\) and \(\tau = \tau_l\). Evaluate \(z_0\) and \(y_0\) by (2.7). Set \( k := 0 \).

**Step 2.1** Find first an approximate solution \(p_k\) of subproblem (3.4)–(3.7) such that (3.8) holds, then solve the QP subproblem (2.1). Set \( l := 0 \).

**Step 2.2** Choose \(\rho_{k+1}\) with either \(\rho_{k+1} = \rho_k\) or \(\rho_{k+1} \leq 0.5\rho_k\) such that

\[
2\rho_{k+1}\|C(v_k)\|_2 \frac{\|C(v_k)\|_2^T R^{-2} Q(v_k, w_k) R^{-2} C(v_k) C(v_k)\|_2}{\|R^{-1} C(v_k) C(v_k)\|_2} < 1, \quad \hbox{and} \quad (3.16)
\]

\[
\pi(v_k, d_k; \rho_{k+1}) = (1 - \delta)(g^N(p_k; \rho_{k+1}) - \|C(v_k)\|) - \frac{\delta}{2} \rho_{k+1} d_k^T Q(v_k, w_k) d_k, \quad (3.17)
\]

where \(\pi(v_k, d_k; \rho_{k+1})\) is defined in Lemma 3.2.

**Step 2.3** Choose the step-size \(\alpha_k \in (0, 1)\) to be the maximal in \(\{1, \delta, \delta^2, \ldots\}\) such that

\[
\phi(x_k + \alpha_k d_{xk}, t_k + \alpha_k d_{tk}, s_k + \alpha_k d_{sk}; \rho_{k+1}) - \phi(x_k, t_k, s_k; \rho_{k+1}) \leq \sigma \alpha_k \pi(v_k, d_k; \rho_{k+1}). \quad (3.18)
\]

**Step 2.4** Set \(x_{k+1} = x_k + \alpha_k d_{xk}, t_{k+1} = t_k + \alpha_k d_{tk}, \) and \(s_{k+1} = s_k + \alpha_k d_{sk}\).

**Step 2.5** For \(j = 1, \ldots, m\), set

\[
s_{k+1,j} = \begin{cases} 
\hat{s}_{k+1,j}, & \hbox{if } t_{k+1,j} \leq 0; \\
\min\{\hat{s}_{k+1,j}, \mu \frac{\tau_{k+1,j}}{\tau_{k+1,j}}\}, & \hbox{otherwise.} 
\end{cases} \quad (3.19)
\]

**Step 2.6** Compute an estimate \(\lambda_{k+1}\) of Lagrangian multiplier vector corresponding to the equality constraints in (1.3).

**Step 2.7** If \(\|r(v_{k+1}, \lambda_{k+1}; \mu_l, \tau_l)\|_\infty \leq 10\mu_l\), set \(\mu_{l+1} = \min\{0.5\mu_l, \|r_1(v_{k+1}, \lambda_{k+1}; \mu_l, \tau_l)\|_\infty\}^{1.8}\), \(\tau_{l+1} = \tau_l\).

Else if \(\|g(v_{k+1}; \mu_l, \tau_l)\|_\infty \leq \tau_l\), set \(\mu_{l+1} = \mu_l\), \(\tau_{l+1} \leq 0.6\tau_l\).

In above two cases, stop the inner algorithm, set \((x_{l+1}, t_{l+1}, s_{l+1}) = (x_{k+1}, t_{k+1}, s_{k+1})\), \(l := l + 1\).

Otherwise, update \(B_k\) to \(B_{k+1}\), set \(k := k + 1\) and go to Step 2.1.

**End (while)**

Due to Lemma 3.3, the update (3.19) guarantees \(z_{k+1} - t_{k+1} \geq 0\) for all \(k\). Moreover, it will further reduce the value of \(\phi(x_k + \alpha_k d_{xk}, t_k + \alpha_k d_{tk}, s_k + \alpha_k d_{sk}; \rho_{k+1})\). Thus, one has

\[
\phi(x_{k+1}, t_{k+1}, s_{k+1}; \rho_{k+1}) - \phi(x_k, t_k, s_k; \rho_{k+1}) \leq \sigma \alpha_k \pi(v_k, d_k; \rho_{k+1}) < 0 \quad (3.20)
\]
for all $k \geq 0$.

4. Global convergence

Without assuming any regularity of constraints such as feasibility or constraint qualification, the local solution of the problem (1.1)–(1.3) may be either a KKT point or a singular stationary point (i.e. a Fritz-John point). For an infeasible problem, we usually want to find an infeasible stationary point which is a stationary point for minimizing some measure of constraint violations (see [2, 4, 18]).

In order to solve the original problem (1.1)–(1.3), Algorithm 3.6 approximately solves a sequence of logarithmic barrier positive relaxation problem (2.7)–(2.10). In this section, we first prove that, for any given $\mu > 0$ and $\tau > 0$, the inner algorithm of Algorithm 3.6 will terminate in a finite number of iterations. Thus, either $\mu_l \to 0$ or $\tau_l \to 0$ as $l \to \infty$. After that, we consider the global convergence of the whole algorithm. It is proved that our algorithm will terminate at an approximate KKT point of the original problem provided the scaling parameter $\tau_l$ is far from zero. Otherwise, either an approximate infeasible stationary point or an approximate singular stationary point of the original problem will be found.

4.1. Global convergence of the inner algorithm. We consider the global convergence of the inner algorithm. Suppose that, for some given $\mu > 0$ and $\tau > 0$, the inner algorithm of Algorithm 3.6 does not terminate in a finite number of iterations and \{(x_k, t_k, s_k)\} is an infinite sequence generated by the algorithm. We need the following blanket assumptions for our global convergence analysis.

**Assumption 4.1**

1. The functions $f$ and $c_i (i \in I)$ are twice continuously differentiable on $\mathbb{R}^n$;
2. The iterative sequence \{x_k\} is in an open bounded set;
3. The sequence \{B_k\} is bounded, and for all $k \geq 0$ and $d_x \in \mathbb{R}^n$, $d_x^TB_kd_x \geq \gamma \|d_x\|^2$, where $\gamma > 0$ is a constant;
4. For all $k \geq 0$, $p_k$ is an approximate solution of subproblem (3.4)–(3.5) satisfying (3.8).

We have the following results which are proved similar to Lemma 5 of [5] and Lemma 4.2 of [15].

**Lemma 4.2** Suppose that Assumption 4.1 holds. Then \{z_k\} and \{t_k\} are bounded, \{y_k\} is componentwise bounded away from zero and \{s_k\} is lower bounded. Furthermore, if the penalty parameter $\rho_k$ is bounded away from zero as $k \to \infty$, then \{z_k\} is componentwise bounded away from zero, \{y_k\} and \{s_k\} are bounded.

**Proof.** Assumption 4.1 implies that there exists a scalar $\chi > 0$ such that $\|f(x_k)\| \leq \chi$ and $\|c(x_k)\| \leq \chi$ for all $k \geq 0$. Due to (3.20), $\phi(v_{k+1}; \rho_{k+1}) \leq \phi(v_k; \rho_{k+1})$ for all $k \geq 0$. Thus, for every $k \geq 0$,

$$\phi(v_{k+1}; \rho_{k+1}) - \phi(v_k; \rho_k) \leq (\rho_k - \rho_{k+1})(\chi + m\mu \ln \|z_k\|).$$
Therefore,
\[ \phi(v_{k+1}; \rho_{k+1}) \leq \phi(v_0; \rho_0) + (\rho_0 - \rho_{k+1})(\chi + m\mu \max_{0 \leq l \leq k+1} \ln \|z_l\|). \quad (4.1) \]

Note that the inequalities
\[ \phi(v_{k+1}; \rho_{k+1}) \geq -\rho_{k+1}(\chi + m\mu \max_{0 \leq l \leq k+1} \ln \|z_l\|) + \|t_{k+1}\| - \|c(x_{k+1})\| \quad (4.2) \]
and \( \phi(v_{k+1}; \rho_{k+1}) \geq -\rho_{k+1}(\chi + m\mu \max_{0 \leq l \leq k+1} \ln \|z_l\|) + \|z_{k+1}\| - \|t_{k+1}\| \), one has
\[ \phi(v_{k+1}; \rho_{k+1}) \geq -\rho_{k+1}(\chi + m\mu \max_{0 \leq l \leq k+1} \ln \|z_l\|) + \|z_{k+1}\| - \|c(x_{k+1})\|. \quad (4.3) \]

Hence, it follows from (4.1), (4.2), and (4.3) that, for all \( k \geq 0 \),
\[ \phi(v_0; \rho_0) + (1 + \rho_0)\chi + \rho_0m\mu \max_{0 \leq l \leq k+1} \ln \|z_l\| \geq \max(\|z_{k+1}\|, \|t_{k+1}\|), \]
which implies that \( \{z_k\} \) is bounded. Furthermore, \( \{t_k\} \) is bounded since \( \{z_k\} \) is bounded. Due to \( z_{kj} = \tau \mu \) for every \( j = 1, \ldots, m \), the results on \( \{y_k\} \) follow immediately.

For given \( \mu > 0 \) and \( \tau > 0 \), if \( \{z_k\} \) is bounded, by (2.1), \( s_{kj} > -\infty \) for all \( k \geq 0 \) and \( j = 1, \ldots, m \). Otherwise, if \( s_{kj} \rightarrow -\infty \) for some \( j \), then \( z_{kj} \rightarrow \infty \), which is a contradiction. If \( \{z_k\} \) is componentwise bounded away from zero, again by (2.1), \( s_{kj} < +\infty \) for all \( k \geq 0 \) and \( j = 1, \ldots, m \). Thus, the results on \( \{s_k\} \) are proved. \( \Box \)

**Corollary 4.3** Suppose that Assumption (4.1) holds. Then, for any given \( \mu > 0 \) and \( \tau > 0 \), all sequences \( \{C(v_k)\}, \{\nabla F(v_k)\}, \{\nabla C(v_k)\} \) and \( \{Q(v_k, w_k)\} \) are bounded. Thus, there is a constant \( \chi_0 > 0 \) such that \( \|C(v_k)\| \leq \chi_0 \), \( \|\nabla C(v_k)\| \leq \chi_0 \), and \( \lambda_{\max}(Q(v_k, w_k)) \leq \chi_0 \).

*Proof.** For \( j = 1, \ldots, m \), due to \( z_{kj}y_{kj} = \tau \mu \),
\[ \frac{1}{z_{kj} + y_{kj}} = \frac{z_{kj}}{z_{kj}^2 + z_{kj}y_{kj}} \leq \frac{1}{\tau \mu} z_{kj}. \]
Thus, by Lemma 4.2 \( 1/(z_{kj} + y_{kj}) \) is bounded. The boundednesses of \( \{C(v_k)\}, \{\nabla F(v_k)\}, \{\nabla C(v_k)\} \) and \( \{Q(v_k, w_k)\} \) follow immediately from their expressions in previous section. \( \Box \)

**Lemma 4.4** Suppose that Assumption (4.1) holds. If
\[ \|R^{-1}\nabla C(v_k)C(v_k)\| \geq \chi_1\|C(v_k)\| \quad (4.4) \]
for some constant \( \chi_1 > 0 \) and for all \( k \geq 0 \), then there is a constant \( \hat{\rho} > 0 \) such that \( \rho_{k+1} = \hat{\rho} \) for all sufficiently large \( k \).

*Proof.** Due to Corollary 4.3 \( \rho_{k+1}\|C(v_k)\| \leq 1/(2\lambda_{\max}(R^{-1}Q(v_k, w_k))R^{-1}) \) provided \( \rho_{k+1} \leq \tau^2/(2\chi_0^2) \). Thus, if \( \hat{\rho} \leq \tau^2/(2\chi_0^2) \), it follows from (3.9) and (4.1) that
\[ \|C(v_k)\| - \|C(v_k) + \nabla C(v_k)p_k\| \geq \frac{1}{4} \min\{1, \frac{1}{\chi_0}\} \chi_1\|C(v_k)\|. \quad (4.5) \]
We achieve the result by proving that (3.17) holds with \( \rho_{k+1} \leq \hat{\rho} \) for some scalar \( \hat{\rho} > 0 \). Since \( q_k(d_k) \leq q_k(p_k) \), one has

\[
\pi(v_k, d_k; \rho_{k+1}) - (1 - \delta)(q_k^N(p_k; \rho_{k+1}) - \|C(v_k)\|) + \frac{1}{2}\rho_{k+1} d_k^T Q(v_k, w_k) d_k \leq \rho_{k+1} \nabla F(v_k)^T p_k + \frac{1}{2} p_k^T Q(v_k, w_k) p_k + \delta(\|C(v_k)\| + \nabla C(v_k)^T p_k) - \|C(v_k)\| \leq (\rho_{k+1} \xi_1 - \delta \xi_2) \|C(v_k)\|,
\]

where \( \xi_1 \) and \( \xi_2 \) are positive constants. Hence, (3.17) holds with \( \rho_{k+1} \leq \hat{\rho} \) provided \( \hat{\rho} = \min \{\tau^2/(2\chi^2), \delta \xi_2/\xi_1\} \). \( \square \)

**Lemma 4.5** Suppose that Assumption 4.1 holds, \( d_k = (d_{xk}, d_{tk}, d_{sk}) \in \mathbb{R}^{n+2m} \) is a solution of QP (3.6)-(3.7), \( u_k = (\lambda_k, \beta_k, v_k, \nu_k) \) is the associated Lagrangian multiplier vector. If (4.4) holds for all sufficiently large \( k \), the sequence \( \{\|d_k\|\} \) is bounded.

**Proof.** If (4.4) holds for all sufficiently large \( k \), by Lemma 4.4, \( \rho_k \) is bounded away from zero. It follows from Lemma 4.2 that \( z_k \) and \( y_k \) are bounded. Thus, there exists a constant \( \chi_2 > 0 \) such that

\[
\frac{\mu}{(z_{kj} + y_{kj})^2} \geq \chi_2 \quad \text{for} \quad j = 1, \ldots, m.
\]

Due to

\[
d_k^T Q(v_k, w_k)d_k = d_k^T B_k d_k + \sum_{j=1}^{n} \frac{\mu}{(z_{kj} + y_{kj})^2} (d_{ij} - \tau d_{sj})^2 \geq \gamma \|d_x\|^2 + \gamma \|d_t - \tau d_s\|^2 \quad (4.6)
\]

for every \( d = (d_x, d_t, d_s) \), \( d_k^T Q(v_k, w_k)d_k \geq \xi_3 \|(d_{xk}, d_{tk} - \tau d_{sk})\|^2 \) for some scalar \( \xi_3 > 0 \).

Since \( q_k(d_k) \leq q_k(p_k) \), \( q_k(p_k) \leq \chi_3 \) and \( q_k(d_k) \geq -\chi_3 \|(d_{xk}, d_{tk} - \tau d_{sk})\| + \xi_3 \|(d_{xk}, d_{tk} - \tau d_{sk})\|^2 \) for some scalars \( \chi_3 > 0 \) and \( \xi_3 > 0 \), one can deduce that \( \|(d_{xk}, d_{tk} - \tau d_{sk})\| \) is bounded. Otherwise, if \( \|(d_{xk}, d_{tk} - \tau d_{sk})\| \) is unbounded, then \( \xi_3 \leq 0 \), which is a contradiction. Thus, \( \|d_{tk}\| \) is bounded due to \( d_{tk} = p_{tk} - \nabla c(x_k)^T (d_{xk} - p_{xk}) \). It implies that \( \|d_{sk}\| \) is bounded. \( \square \)

**Lemma 4.6** Suppose that Assumption 4.1 holds, \( \{\alpha_k\} \) is the sequence of step-sizes derived from (3.18) of Algorithm 3.6. If the inequality (4.4) holds for all sufficiently large \( k \), then \( \{\alpha_k\} \) is bounded away from zero.

**Proof.** Due to Lemmas 4.4 and 4.5 for every \( j = 1, \ldots, m \), one has

\[
- \ln z_j(v_k + \alpha d_k; \mu, \tau) + \ln z_{kj} - \alpha \frac{1}{z_{kj} + y_{kj}} e_j^T (d_{tk} - \tau d_{sk}) = o(\alpha), \quad (4.7)
\]

\[
\|C(v_k + \alpha d_k)\| = \|C(v_k) + \alpha \nabla C(v_k)^T d_k\| + o(\alpha) \quad (4.8)
\]

for all \( \alpha > 0 \) sufficiently small. Hence,

\[
\phi(v_k + \alpha d_k; \rho_{k+1}) - \phi(v_k; \rho_{k+1}) = \alpha \pi(v_k, d_k; \rho_{k+1}) + o(\alpha) \quad (4.9)
\]
for all $\alpha \in [0, \bar{\alpha}]$, where $\bar{\alpha} > 0$ is a sufficiently small constant. Due to

$$(1 - \sigma)\alpha \pi(v_k, d_k; \rho_{k+1}) \leq \alpha (1 - \delta)(q_k^b(p_k; \rho_{k+1}) - \|C(v_k)\|) \leq -\alpha \xi_4 \|C(v_k)\|$$

(4.10)

(where $\xi_4 = \xi_2(1 - \sigma)(1 - \delta)$), it follows from (4.9) and (4.10) that there exists a scalar $\hat{\alpha} \in (0, \bar{\alpha}]$ such that

$$\phi(v_k + \alpha d_k; \rho_{k+1}) - \phi(v_k; \rho_{k+1}) \leq \sigma \alpha \pi(v_k, d_k; \rho_{k+1})$$

for all $\alpha \in (0, \hat{\alpha}]$ and all $k \geq 0$. Thus, by Step 2.3 of Algorithm 3.6, $\alpha_k \geq \delta \hat{\alpha}$ for all $k \geq 0$.

Lemma 4.7 Suppose that Assumption 4.1 holds. If the inequality (4.4) holds for all sufficiently large $k$, then

$$\lim_{k \to \infty} \|C(v_k)\| = 0 \quad \text{and} \quad \lim_{k \to \infty} \|d_k\| = 0.$$ (4.11)

Proof. According to Lemma 4.4, without loss of generality, we suppose that $\rho_k = \hat{\rho}$ for all $k \geq 0$. Then, by (3.20), $\{\phi(v_k; \rho_{k+1})\}$ is a monotonically non-increasing sequence. Note that it is also a bounded sequence. Thus,

$$\lim_{k \to \infty} \pi(v_k, d_k; \rho_{k+1}) = 0$$ (4.12)

since $\alpha_k$ is bounded away from zero. Using the last inequality of (4.10), one has

$$\lim_{k \to \infty} \|C(v_k)\| = 0,$$

which implies $\lim_{k \to \infty} \|p_k\| = 0$. Thus, by (3.17) and (4.6), $\lim_{k \to \infty} \|d_k\| = 0$.}

Now we are ready to present our global convergence results on the inner algorithm.

Theorem 4.8 Given $\mu > 0$ and $\tau > 0$. Suppose that Assumption 4.1 holds. Let $\{v_k\}$ and $\{\lambda_k\}$ be two sequences generated by the inner algorithm of Algorithm 3.6. Let $\epsilon > 0$ be any given small scalar. Then there is an integer $k > 0$ such that either $\|r(v_{k+1}, \lambda_{k+1}; \mu, \tau)\|_{\infty} \leq \epsilon$ or $\|g(v_{k+1}; \mu, \tau)\|_{\infty} \leq \epsilon$.

Proof. If (4.4) holds for all $k \geq 0$, by Lemma 4.4, $p_k$ remains a positive constant after a finite number of iterations. Thus, $\{z_k\}$ and $\{y_k\}$ are bounded above and componentwise bounded away from zero, $\{s_k\}$ is bounded.

We prove the result by contradiction. If the result does not hold, then the inner algorithm will not terminate in a finite number of iterations. Thus, $\{v_k\}$ is an infinite sequence. Let $v^* = (x^*, t^*, s^*)$ be any limit point of $\{v_k\}$. Without loss of generality, suppose that $\lim_{k \to \infty} v_k = v^*$. Due to Lemma 4.7 $\lim_{k \to \infty} \|C(v_k)\| = 0$ and $\lim_{k \to \infty} \|d_k\| = 0$. Thus,

$$\lim_{k \to \infty} z_k = t^* > 0, \quad \lim_{k \to \infty} y_k = \tau s^* > 0.$$
since $z_k - t_k = y_k - \tau s_k$. Moreover, by taking the limit on $k \to \infty$ in both sides of the following KKT condition
\[ \nabla F(v_k) + Q(v_k, w_k) d_k + \nabla C(v_k) u_k = 0 \]
of subproblem (3.6)–(3.7), due to Corollary 4.3 one has
\[ \nabla F(v_k) + \nabla C(v_k) u_k = 0. \tag{4.13} \]
Note that $\lim_{k \to \infty} \|C(v_k)\| = 0$. Therefore, every limit point of $\{v_k\}$ is a KKT point of (2.7)–(2.10). In view of Proposition 3.5 there exists a $\lambda_k \in \mathbb{R}^{me}$ such that the inequality $\|r(v_{k+1}, \lambda_{k+1}; \mu, \tau)\|_{\infty} \leq \epsilon$ holds for every sufficiently large $k$.

In the following, suppose that (4.4) does not hold for all sufficiently large $k$. Then there exists some infinite index subset $\mathcal{K}$ such that the condition (4.4) does not hold for all $k \in \mathcal{K}$. That is,
\[ \lim_{k \in \mathcal{K}, k \to \infty} \|R^{-1} \nabla C(v_k) C(v_k)\| / \|C(v_k)\| = 0. \tag{4.14} \]
Due to Proposition 3.5 one has $\|g(v_{k+1}; \mu, \tau)\|_{\infty} \leq \epsilon$ for all sufficiently large $k \in \mathcal{K}$.

The above arguments show that the sequence $\{v_k\}$ cannot be an infinite sequence. This contradiction implies the result of the theorem.

**4.2. Convergence results of the whole algorithm.** Now we consider the global convergence of the whole algorithm. It is well known that, without assuming any constraint qualification, a local solution of general nonlinear program can be either a KKT point or a Fritz-John point of the problem. For those nonlinear programs arising from practical situation, whether they are feasible are not known before solving them. Thus, some robust methods for nonlinear programs not only focus on convergence to KKT points of problems under some assumptions on constraint regularities, but also concern about convergence to Fritz-John points and infeasible stationary points of problems without assuming any regularity on constraints (for example, see [2, 3, 4, 7, 8, 10, 15, 17, 28]).

**Definition 4.9** $x^* \in \mathbb{R}^n$ is called a Fritz-John point or a singular stationary point of problem (1.1)–(1.3) if there exist $\lambda^* \in \mathbb{R}^{me}$ and $\beta^* \in \mathbb{R}^m$ such that
\[ \nabla h(x^*) \lambda^* + \nabla c_i(x^*) \beta^*_i = 0, \]
\[ h_i(x^*) = 0, \quad i = 1, \ldots, me, \]
\[ \beta^*_j \geq 0, \quad c_j(x^*) \leq 0, \quad \beta^*_j c_j(x^*) = 0, \quad j = 1, \ldots, m. \]

**Definition 4.10** $x^* \in \mathbb{R}^n$ is called an infeasible stationary point of problem (1.1)–(1.3) if $x^*$ is an infeasible point and
\[ \nabla h(x^*) h(x^*) + \nabla c(x^*) \max(0, c(x^*)) = 0. \]

Now we are ready to present our convergence results on the whole algorithm.
Theorem 4.11 Suppose that Assumption 4.1 holds for every given parameter $\mu_l > 0$ and $\tau_l > 0$, sequences $\{x_l\}$ and $\{B_l\}$ are bounded. Let $\epsilon > 0$ be small enough. Then either $\mu_l \leq \epsilon$ or $\tau_l \leq \epsilon$ for some $l$, and one of the following statement is true.

1. The parameter $\tau_l > \epsilon$, the inner algorithm terminates at a point $v_{l+1}$ where the terminating condition $\|r(v_{l+1}; \lambda_{l+1}; \mu_l, \tau_l)\|_\infty \leq 10\epsilon$ holds. Algorithm 3.6 terminates at an approximate KKT point of the original problem (1.1)–(1.3);

2. The parameter $\tau_l \leq \epsilon$, the inner algorithm terminates at a point $v_{l+1}$ where the condition $\|g(v_{l+1}; \mu_l, \tau_l)\|_\infty \leq \epsilon$ is satisfied and $\|C(v_k)\|$ is small enough. Algorithm 3.6 terminates at a point which is an approximate singular stationary point of the problem (1.1)–(1.3);

3. The parameter $\tau_l \leq \epsilon$, the inner algorithm terminates due to $\|g(v_{l+1}; \mu, \tau)\|_\infty \leq \epsilon$ and $\|C(v_k)\|$ is bounded away from zero. Algorithm 3.6 terminates at a point $v_{l+1}$ which is an approximate infeasible stationary point of the problem (1.1)–(1.3).

Proof. For every $l \geq 0$, Theorem 4.8 shows that, in Algorithm 3.6 either $\mu_l$ or $\tau_l$ will be reduced. Finally, there is either $\mu_l \leq \epsilon$ or $\tau_l \leq \epsilon$. Furthermore, The argument of Lemma 4.2 shows that $\{z_l\}$ and $\{t_l\}$ are bounded.

The condition $\|r(v_{l+1}; \lambda_{l+1}; \mu_l, \tau_l)\|_\infty \leq \epsilon$ implies that $(x_{l+1}, t_{l+1})$ is an approximate KKT point of the logarithmic barrier positive relaxation problem (1.1)–(1.3). It is typically known that, as $\mu_l$ is small enough, $x_{l+1}$ is also an approximate KKT point of the original problem (1.1)–(1.3).

If it is other than the above case, then one has $\tau_l \leq \epsilon$ and $\|g(v_{l+1}; \mu_l, \tau_l)\|_\infty \leq \epsilon$. That is,

$$\|\nabla h(x_{l+1})h(x_{l+1}) + \nabla c(x_{l+1})(z_{l+1} - t_{l+1})\|_\infty/\|C(v_{l+1})\| \leq \epsilon,$$  \hspace{1cm} (4.15)

$$\|Z_{l+1}(z_{l+1} - t_{l+1})\|_\infty/\|C(v_{l+1})\| \leq \epsilon.$$  \hspace{1cm} (4.16)

In order to demonstrate that the point $x_{l+1}$ satisfying (4.15)–(4.16) is an approximate singular stationary point of the problem (1.1)–(1.3) when $\|C(v_{l+1})\|$ is small enough, we take the limit $\epsilon \to 0$ on both sides of (4.15)–(4.16). The limit $\epsilon \to 0$ implies $\tau_l \to 0$. Without loss of generality, suppose $z_{l+1} \to z^*$ and $x_{l+1} \to x^*, t_{l+1} \to t^*$ as $\tau_l \to 0$, where $x^*, t^*, z^*$ satisfy $h(x^*) = 0$, $c(x^*) + t^* = 0$, $z^* - t^* = 0$. Furthermore, $h(x_{l+1})/\|C(v_{l+1})\| \to \lambda^*$, $(z_{l+1} - t_{l+1})/\|C(v_{l+1})\| \to \beta^*$. Thus, $c(x^*) \leq 0$, $\beta^* \geq 0$ (since $z_{l+1} - t_{l+1} \geq 0$ for all $l$), and due to (4.15), (4.16), and $c(x^*) = -t^* = -z^*$,

$$\nabla h(x^*)\lambda^* + \nabla c(x^*)\beta^* = 0,$$

$$\beta^*_j c_j(x^*) = 0.$$

That is, $x^*$ is a Fritz-John point problem (1.1)–(1.3). Therefore, $x_{l+1}$ is an approximate singular stationary point of the problem.

If $\|g(v_{l+1}; \mu_l, \tau_l)\|_\infty \leq \epsilon$ but $C(v_{l+1})$ is bounded away from zero, then

$$\|\nabla h(x_{l+1})h(x_{l+1}) + \nabla c(x_{l+1})(z_{l+1} - t_{l+1})\|_\infty \leq \epsilon,$$  \hspace{1cm} (4.17)

$$\|c(x_{l+1}) + t_{l+1} - (z_{l+1} - t_{l+1})\|_\infty \leq \epsilon,$$  \hspace{1cm} (4.18)

$$\|Z_{l+1}(z_{l+1} - t_{l+1})\|_\infty \leq \epsilon.$$  \hspace{1cm} (4.19)
Similar to the preceding arguments, suppose \( z_{l+1} \rightarrow z^* \) and \( x_{l+1} \rightarrow x^* \), \( t_{l+1} \rightarrow t^* \) as \( \tau_l \rightarrow 0 \), then \( t^* = \frac{1}{\tau}(z^* - c(x^*)) \) due to (4.18). Thus, by (4.19), \( z_j^*(z_j^* + c_j(x^*)) = 0 \) for \( j = 1, \ldots, m \). This fact implies

\[
c_j(x^*) + t_j^* = z_j^* - t_j^* = \max(0, c_j(x^*)), \quad j = 1, \ldots, m.
\]

Therefore, it follows from (3.13) that

\[
\nabla h(x^*)h(x^*) + \nabla c(x^*) \max(0, c(x^*)) = 0.
\]

That is, \( x^* \) is a stationary point to minimize \( \frac{1}{2}||(h(x), \max\{0, c(x)\})||^2 \), thus is an infeasible stationary point of problem (1.1)–(1.3). It shows that \( x_{l+1} \) is an approximate infeasible stationary point of problem (1.1)–(1.3).

5. Numerical experiments

The numerical experiments were conducted on a Lenovo laptop with the LINUX operating system (Fedora 11). The algorithm were implemented in MATLAB (version R2008a). Two kinds of test problems originated from the literature were solved, including some simple but hard problems, which may be an infeasible problem, a problem feasible but LICQ and MFCQ failing to hold at the solution, or is a well-posed one but some class of interior-point methods was proved not to be globally convergent, and some standard test problems of the CUTE collection [1]. These problems have been solved in [10].

We use the standard initial point \( x_0 \) for all test problems, and set \( t_0 = -c(x_0) \), \( s_{0j} = \min\{1, 0.95\mu/\max(0, t_{0j})\} \). The initial parameters are selected as follows: \( \mu_0 = 0.1, \tau_0 = 1, \delta = 0.5, \sigma = 0.0001, \) and \( \epsilon = 10^{-8} \). The initial penalty is selected to be dependent on the initial point \( x_0 \) as \( \rho_0 = \min\{100, \max\{1, ||(\max\{0, c(x_0)\}, h(x_0))||/||f(x_0)||\}\} \), \( B_0 \) is simply taken as the identity matrix and \( B_k \) is updated by the well-known Powell’s damped BFGS update formula (for example, see [13]). The subproblems are solved by similar techniques as those used in [10], while subproblem (3.4)–(3.5) in this paper should be treated more carefully since \( Q(v_k, w_k) \) is always positive semi-definite. The whole algorithm is terminated as either \( \mu_l \leq \epsilon \) or \( \tau_l \leq \epsilon \), or the total number of iterations (that is, the number of solving QP (3.6)–(3.7)) is larger than 1000.

5.1. Numerical results on three simple but hard problems. In this subsection, we report our numerical results on three simple but hard examples taken from the literature. In Tables 1, 2 and 3, the number in column \( l \) means that the data in the row are taken from the corresponding outer iterate at which either \( \mu_l \) or \( \tau_l \) is reduced, \( f_l = f(x_l), v_l = ||(h(x_l), \max\{0, c(x_l)\})||, ||r_l||_\infty = ||r(x_{l+1}, s_{l+1}; \lambda_{l+1}; \mu_l, \tau_l)||_\infty, ||g_l||_\infty = ||g(x_{l+1}, s_{l+1}; \mu_l, \tau_l)||_\infty, k \) is the number of inner iterations needed from \( (\mu_{l-1}, \tau_{l-1}) \) to \( (\mu_l, \tau_l) \).

The first example was presented by Wächter and Biegler [24] and further discussed by Byrd, Marazzi and Nocedal [7]:

\[
\min x_1
\]
Table 1: Output for test problem (TP1)

| $t$ | $f_t$ | $v_t$ | $||r_t||_{\infty}$ | $||g_t||_{\infty}$ | $\mu_t$ | $\tau_t$ | $k$ |
|-----|-------|-------|-------------------|-------------------|---------|---------|-----|
| 0   | -4    | 14    | 14                | 7.6026            | 0.1000  | 1       | -   |
| 1   | -1.5235 | 3.5562 | 3.5477            | 0.9974            | 0.1000  | 0.6000  | 2   |
| 2   | -1.2344 | 2.2938 | 0.9737            | 0.5926            | 0.1000  | 0.3600  | 2   |
| 3   | -1.0298 | 2.2408 | 0.4768            | 0.3133            | 0.1000  | 0.2160  | 1   |
| 4   | -0.4257 | 1.9679 | 0.1957            | 0.6123            | 0.0500  | 0.2160  | 5   |
| 5   | 2.0245  | 1.6851e-04 | 0.0049           | 4.4642            | 0.0011  | 0.2160  | 6   |
| 6   | 2.0011  | 5.4789e-04 | 1.1834e-04       | 3.9978            | 1.3479e-06 | 0.2160  | 1   |
| 7   | 2.0000  | 1.1711e-06 | 2.5296e-07       | 4.0000            | 1.0000e-09 | 0.2160  | 1   |
| 8   | 2.0000  | 1.8137e-12 | 5.6019e-11       | 4.0000            | -       | -       | 1   |

The standard initial point is $x_0 = (-4, 1, 1)$. This problem is a well-posed problem. It has a unique global minimizer $(2, 3, 0)$, at which gradients of the active constraints are linearly independent, and MFCQ holds. However, [24] showed that many line search interior-point methods might fail to find the solution.

Our algorithm terminates at the approximate solution $x^* = (2.0000, 3.0000, 0.0000)$ together with $t^* = (3.0000, 0.0000)$ and $s^* = (0.0000, 1.0000)$ in totally 19 iterations. The numbers of function and gradient evaluations are 20 and 20, respectively. See Table 1 for more details on iterations, from there one can observe the superlinear convergence of $v_t$ and $||r_t||_{\infty}$.

The second example is a standard test problem taken from [14, Problem 13]:

\[
\begin{align*}
\min & \quad (x_1 - 2)^2 + x_2^2 \\
(TP2) & \quad \text{s.t.} \quad (1 - x_1)^3 - x_2 \geq 0, \quad x_1 \geq 0, \quad x_2 \geq 0.
\end{align*}
\]

The standard initial point $x_0 = (-2, -2)$ for problem (TP2) is an infeasible point. Note that its optimal solution $x^* = (1, 0)$ is not a KKT point but a singular stationary point, at which LICQ and MFCQ fail to hold.

This problem has not been solved in [20] [27], but has been solved in [6] [21]. Algorithm 3.6 terminated at an approximate point to the solution $x_t = (0.9905, -0.0000)$. The associated vectors are $t_t = (0.0000, 0.9905, 0.0000)$, $s_t = 1c3(7.3986, 0.0000, 7.3989)$. The numbers of iterations, function evaluations and gradient evaluations are 28, 76 and 29, respectively (see Table 2).

The third example is an infeasible problem named isolated presented by Byrd, Curtis and Nocedal [4].

\[
\min x_1 + x_2
\]
Table 2: Output for test problem (TP2)

| $l$ | $f_l$ | $v_l$ | $\|r_l\|_{\infty}$ | $\|g_l\|_{\infty}$ | $\mu_l$ | $\tau_l$ | $k$   |
|-----|-------|-------|---------------------|---------------------|---------|---------|------|
| 0   | 20    | 2     | 8.9116              | 0.7071              | 0.1000  | 1       | -    |
| 1   | 19.6742 | 2.7923 | 1.6715              | 0.7592              | 0.1000  | 1.5502e-05 | 1   |
| 2   | 3.9492 | 0     | 6.2765e-06         | 1.0000              | 1.7127e-07 | 1.5502e-05 | 9   |
| 3   | 1.0192 | 9.1401e-11 | 1.3450e-11 | 1.0000              | 1.0000e-09 | 1.5502e-05 | 17  |
| 4   | 1.0192 | 9.1390e-11 | 1.3450e-11 | 0.0203              | -       | -       | 1   |

Table 3: Output for test problem (TP3)

| $l$ | $f_l$ | $v_l$ | $\|r_l\|_{\infty}$ | $\|g_l\|_{\infty}$ | $\mu_l$ | $\tau_l$ | $k$   |
|-----|-------|-------|---------------------|---------------------|---------|---------|------|
| 0   | 5     | 12    | 43.1944             | 7.5805              | 0.1000  | 1       | -    |
| 1   | 2.5509 | 5.8735 | 4.3125              | 3.1566              | 0.1000  | 0.6000  | 1    |
| 2   | 0.7102 | 2.3657 | 0.5663              | 0.5946              | 0.0500  | 0.6000  | 1    |
| 3   | -0.1702 | 2.0217 | 0.4846              | 0.3160              | 0.0500  | 0.3600  | 1    |
| 4   | 0.0606 | 2.0029 | 0.2179              | 0.1233              | 0.0500  | 0.2160  | 1    |
| 5   | -0.0633 | 2.0042 | 0.1209              | 0.0903              | 0.0500  | 2.0359e-04 | 1  |
| 6   | 1.0010e-04 | 2.0000 | 1.0183e-04 | 1.8054e-04 | 0.0500  | 1.0000e-09 | 11  |
| 7   | -7.8060e-05 | 2.0000 | 5.0005e-10 | 2.3597e-07 | -       | -       | 1    |

(TP3) \[ s.t. \quad \begin{align*}
    x_1^2 - x_2 + 1 & \leq 0, \\
    x_1^2 + x_2 + 1 & \leq 0, \\
    -x_1 + x_2^2 + 1 & \leq 0, \\
    x_1 + x_2^2 + 1 & \leq 0.
\end{align*} \]

The standard initial point is $x_0 = (3, 2)$, its solution $x^* = (0, 0)$ is a strict minimizer of the Euclidean norm of constraint infeasibility measures. The algorithm presented in [4] found this point. Our algorithm terminates at an approximate point to it with $x_l = 1.0e-04(-0.1547, -0.6259)$, $t_l = (-0.5000, -0.5000, -0.5000, -0.5000)$, $s_l = 1.0e04(5.3752, -5.4001, 3.6816, -3.6977)$ in 17 iterations. Note that $t_l$ is not positive as it should always be in existing interior-point methods based on solving logarithmic barrier problem (1.4)–(1.6). The numbers of function and gradient evaluations are 20 and 18, respectively. Some more details please refer to Table 3.

5.2. Numerical results on test problems of the CUTE collection. A set of 59 small- and medium-size test problems with general inequality constraints from the CUTE collection [1] were solved. These problems were selected since they had actual numbers of problem variables and general inequality constraints (i.e., not only bound constraints). Besides general inequality constraints, some test problems may also have equality constraints. In particular, these problems have been solved and the numerical results have been reported in [10].

We report our numerical results in Tables 4 and 5 where the columns “$n$” and “$m$” are the numbers of variables and constraints of test problems, respectively. The columns of $f$ and $v$ show,
respectively, the values of objective functions and the infinite norms of constraint violations at
the terminating points, “iter” represents the total number of iterations needed for obtaining
those values. For comparison, we denote “∥r∥∞” to be the same as “∥ϕ∥∞” in [10]. The last
two columns “N_f” and “N_g” of Tables 4 and 5 are respectively the numbers of evaluations of
functions and gradients needed by the algorithm.

Note that Tables 4 and 5 only list 48 test problems for which the terminating conditions
of our algorithm have been satisfied before reaching the restriction on number of iterations.
The algorithm in [10] has terminated far from reaching the restriction on the total number of
iterations for 55 problems. The observed reason is that our algorithm in this paper seems to be
more sensitive to the update of scaling parameter corresponding to the penalty parameter in [10],
which possibly results in either more iterations or smaller step-sizes. This may be reasonable
when comparing with the interior-point method using an augmented Lagrangian function.

The preliminary results show that our algorithm can still be efficient for most of test prob-
lems. In particular, our algorithm need obviously fewer iterations, fewer evaluations of functions
and gradients for some test problems such as CHACONN2, HAIFA S, HS29, HS43, POLAK5,
and so on. However, our algorithm is still not available for using the exact Hessian, thus it is
not comparable with some well performed algorithms (for example, [9, 25]) and software such as
IPOPT. Since our MATLAB implementation uses MATLAB routines simply, it is believed that
further improvements can be achieved by using advanced techniques for, e.g., the computations
of subproblems.

6. Conclusion

We present a primal-dual interior-point relaxation method for nonlinear programs in this
article. The method is based on an equivalence between the classic logarithmic barrier problem
and a particular logarithmic barrier positive relaxation problem. Remarkably different from
the current primal-dual interior-point methods in the literature, our method does not require
any primal or dual iterates to be interior-point points. Thus, the interior-point restriction
in existing line-search methods can be removed. A new logarithmic barrier penalty function
dependent on both primal and dual variables was used to prompt the global convergence of the
method, where the penalty parameter is updated adaptively. Without assuming any regularity
of constraints such as feasibility or constraint qualification, the method is proved to be of strong
global convergence. Preliminary numerical results demonstrates that the method can not only
be efficient for well-posed feasible problems, but also is applicable for some ill-posed feasible
problems and some even infeasible problems.

References

[1] I. Bongartz, A.R. Conn, N.I.M. Gould and P.L. Toint, CUTE: Constrained and Uncon-
strained Testing Environment, ACM Tran. Math. Software, 21(1995), 123–160.
| Problem       | n | m | f   | v    | iter | $\|r\|_\infty$ | $N_f$ | $N_g$ |
|--------------|---|---|-----|------|------|----------------|-------|-------|
| CB2          | 3 | 3 | 1.9522 | 9.4943e-08 | 8 | 8.2734e-06 | 9 | 9 |
| CB3          | 3 | 3 | 2.0000 | 0 | 9 | 4.3026e-09 | 10 | 10 |
| CHACONN1     | 3 | 3 | 1.9522 | 0 | 8 | 5.3363e-10 | 9 | 9 |
| CHACONN2     | 3 | 3 | 2.0000 | 0 | 9 | 2.6625e-11 | 10 | 10 |
| CONIGMZ      | 3 | 5 | 28.0000 | 2.5079e-09 | 36 | 3.4740e-07 | 69 | 37 |
| DIPIGRI      | 7 | 4 | 680.6301 | 0 | 24 | 3.0065e-09 | 88 | 25 |
| EXPFITA      | 5 | 22 | 0.0011 | 0 | 191 | 8.8791e-04 | 1478 | 192 |
| EXPFITC      | 5 | 502 | 0.0233 | 0 | 707 | 0.0019 | 5813 | 708 |
| GIGOMEZ1     | 3 | 3 | -3.0000 | 0 | 16 | 6.5600e-11 | 43 | 17 |
| GIGOMEZ2     | 3 | 3 | 1.9522 | 9.7055e-10 | 8 | 9.6530e-08 | 10 | 9 |
| GIGOMEZ3     | 3 | 3 | 2.0000 | 0 | 13 | 7.5763e-08 | 28 | 14 |
| GIGOMEZ4     | 13 | 9 | -0.4500 | 0 | 12 | 1.7810e-09 | 21 | 13 |
| HS10         | 2 | 1 | -1.0000 | 3.6877e-09 | 11 | 8.0982e-09 | 12 | 12 |
| HS11         | 2 | 1 | -8.4985 | 0 | 7 | 3.8354e-09 | 8 | 8 |
| HS12         | 2 | 1 | -30.0000 | 0 | 12 | 8.0016e-09 | 36 | 13 |
| HS14         | 2 | 2 | 1.3935 | 0 | 6 | 4.5752e-13 | 7 | 7 |
| HS22         | 2 | 2 | 1.0000 | 0 | 8 | 6.4723e-11 | 10 | 9 |
| HS29         | 3 | 1 | -22.6274 | 0 | 35 | 2.0747e-14 | 142 | 36 |
| HS34         | 4 | 3 | -44.0000 | 0 | 14 | 1.1305e-09 | 18 | 15 |
| HS88         | 2 | 1 | 1.2980e-05 | 0.1332 | 6 | 2.0113e-10 | 8 | 7 |
| HS89         | 3 | 1 | 0.8163 | 0.0050 | 14 | 3.2978e-11 | 19 | 15 |
| HS90         | 4 | 1 | 7.8447e-06 | 0.1332 | 9 | 6.6620e-11 | 40 | 10 |
| HS91         | 5 | 1 | 8.2135e-05 | 0.1332 | 6 | 6.6644e-11 | 8 | 7 |
| HS92         | 6 | 1 | 2.6369e-06 | 0.1332 | 6 | 4.3302e-10 | 8 | 7 |
| HS100        | 7 | 4 | 680.6301 | 0 | 24 | 3.0065e-09 | 88 | 25 |
| HS100MOD     | 7 | 4 | 678.6798 | 0 | 54 | 0.0036 | 263 | 55 |
| HS113        | 10 | 8 | 24.3062 | 0 | 139 | 2.3607e-10 | 1222 | 140 |
| KISSING      | 127 | 903 | 32.7317 | 2.3054 | 85 | 1.4267e-08 | 1043 | 86 |
| KIWCRES      | 3 | 2 | 2.0000e-09 | 0 | 13 | 2.7207e-09 | 27 | 14 |
| MAKElia1     | 3 | 2 | -1.4142 | 0 | 10 | 1.4611e-09 | 22 | 11 |
| MAKElia2     | 3 | 3 | 7.2000 | 0 | 16 | 1.3945e-09 | 41 | 17 |
| MAKElia3     | 21 | 20 | 1.9895e-08 | 0 | 331 | 4.9710e-10 | 365 | 332 |
| MAKElia4     | 21 | 40 | 5.4155e-08 | 0 | 58 | 2.5122e-08 | 354 | 59 |
| MIFFLIN1     | 3 | 2 | -1.0000 | 0 | 7 | 4.0317e-07 | 8 | 8 |
| MINMAXRB     | 3 | 4 | 2.1057e-06 | 6.0793e-06 | 508 | 0.0052 | 5492 | 509 |
| PENTAGON     | 6 | 15 | 1.3685e-04 | 0 | 177 | 1.0524e-04 | 1066 | 178 |
| POLAK1       | 3 | 2 | 2.7183 | 0 | 165 | 1.6286e-09 | 1569 | 166 |
| POLAK3       | 12 | 10 | 5.9330 | 0 | 22 | 1.9982e-08 | 26 | 23 |
| POLAK5       | 3 | 2 | 50.0000 | 0 | 7 | 9.4837e-12 | 9 | 8 |
| POLAK6       | 5 | 4 | 15.6949 | 0.0013 | 52 | 6.6387e-10 | 553 | 53 |
| ROSENMMX     | 5 | 4 | -44.0000 | 0 | 85 | 6.6579e-08 | 854 | 86 |
| S268         | 5 | 5 | 0.0289 | 0 | 21 | 3.2134e-05 | 106 | 22 |
| SPIRAL       | 3 | 2 | -2.6781e-08 | 2.6781e-08 | 162 | 7.6149e-09 | 422 | 163 |
Table 5: Results for CUTE problems using approximate Hessians, part 2.

| Problem      | n  | m   | f   | v    | iter | $\|r\|_\infty$ | $N_f$ | $N_q$ |
|--------------|----|-----|-----|------|------|----------------|------|------|
| TFI2         | 3  | 101 | 0.6468 | 0.0088 | 30   | 8.8354e-12 | 237 | 31   |
| VANDERM1     | 100 | 199 | 0   | 0.0168 | 6    | 1.0561e-10 | 8   | 7    |
| VANDERM2     | 100 | 199 | 0   | 0.0168 | 6    | 1.0561e-10 | 8   | 7    |
| VANDERM3     | 100 | 199 | 0   | 0.0203 | 7    | 1.6996e-10 | 8   | 8    |
| WOMFLET      | 3  | 3   | 4.3957 | 0     | 23   | 1.3885e-07 | 116 | 24   |

[2] J.V. Burke, F.E. Curtis and H. Wang, A sequential quadratic optimization algorithm with rapid infeasibility detection, SIAM J. Optim., 24(2014), 839–872.

[3] R.H. Byrd, Robust trust-region method for constrained optimization, Paper presented at the SIAM Conference on Optimization, Houston, TX, 1987.

[4] R.H. Byrd, F.E. Curtis and J. Nocedal, Infeasibility detection and SQP methods for nonlinear optimization, SIAM J. Optim., 20(2010), 2281–2299.

[5] R.H. Byrd, J.C. Gilbert and J. Nocedal, A trust region method based on interior point techniques for nonlinear programming, Math. Program., 89(2000), 149–185.

[6] R.H. Byrd, M.E. Hribar and J. Nocedal, An interior point algorithm for large-scale nonlinear programming, SIAM J. Optim., 9 (1999), 877–900.

[7] R.H. Byrd, M. Marazzi and J. Nocedal, On the convergence of Newton iterations to nonstationary points, Math. Program., 99(2004), 127–148.

[8] L.F. Chen and D. Goldfarb, Interior-point $\ell_2$-penalty methods for nonlinear programming with strong global convergence properties, Math. Program., 108(2006), 1–36.

[9] F.E. Curtis, A penalty-interior-point algorithm for nonlinear constrained optimization, Math. Program. Comput., 4(2012), 181–209.

[10] Y.-H. Dai, X.-W. Liu and J. Sun, An interior-point primal-dual method capable to rapidly detect infeasibility for nonlinear programs, Research Report, 2014.

[11] A. Forsgren and P.E. Gill, Primal-dual interior methods for nonconvex nonlinear programming, SIAM J. Optim., 8 (1998), 1132–1152.

[12] P.E. Gill and D.P. Robinson, A primal-dual augmented Lagrangian, Comput. Optim. Appl., 51 (2012), 1–25.

[13] N.I.M. Gould, D. Orban and Ph.L. Toint, An interior-point $\ell_1$-penalty method for nonlinear optimization, Recent Developments in Numerical Analysis and Optimization, Proceedings of NAOIII 2014, Springer, Verlag, 134(2015), 117–150.

[14] W. Hock and K. Schittkowski, Test Examples for Nonlinear Programming Codes, Lecture Notes in Eco. and Math. Systems 187, Springer-Verlag, Berlin, New York, 1981.

[15] X.-W. Liu and J. Sun, A robust primal-dual interior point algorithm for nonlinear programs, SIAM J. Optim., 14(2004), 1163–1186.
[16] X.-W. Liu and Y.-X. Yuan, A robust algorithm for optimization with general equality and inequality constraints, SIAM J. Sci. Comput., 22(2000), 517–534.

[17] X.-W. Liu and Y.-X. Yuan, A null-space primal-dual interior-point algorithm for nonlinear optimization with nice convergence properties, Math. Program., 125(2010), 163–193.

[18] J. Nocedal, F. Öztürk and R.A. Waltz, An interior point method for nonlinear programming with infeasibility detection capabilities, 2012.

[19] J. Nocedal and S. Wright, Numerical Optimization, Springer-Verlag New York, Inc., 1999.

[20] D. F. Shanno and R. J. Vanderbei, Interior-point methods for nonconvex nonlinear programming: Orderings and higher-order methods, Math. Program., 87 (2000), 303–316.

[21] P. Tseng, Convergent infeasible interior-point trust-region methods for constrained minimization, SIAM J. Optim., 13 (2002), 432–469.

[22] M. Ulbrich, S. Ulbrich and L.N. Vicente, A globally convergent primal-dual interior-point filter method for nonlinear programming, Math. Program., 100(2004), 379–410.

[23] R.J. Vanderbei and D.F. Shanno, An interior-point algorithm for nonconvex nonlinear programming, Comput. Optim. Appl., 13 (1999), 231–252.

[24] A. Wächter and L. T. Biegler, Failure of global convergence for a class of interior point methods for nonlinear programming, Math. Program., 88 (2000), 565–574.

[25] A. Wächter and L.T. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Math. Program., 106 (2006), 25–57.

[26] M.H. Wright, Why a pure primal Newton barrier step may be infeasible? SIAM J. Optim., 5 (1995), 1–12.

[27] H. Yamashita, A globally convergent primal-dual interior point method for constrained optimization, Optim. Methods Softw., 10 (1998), 448–469.

[28] Y.-X. Yuan, On the convergence of a new trust region algorithm, Numer. Math., 70(1995), 515–539.