A feasible descent SQP algorithm for general constrained optimization without strict complementarity∗

Jin-Bao Jian∗, Chun-Ming Tang, Qing-Jie Hu, Hai-Yan Zheng

College of Mathematics and Information Science, Guangxi University, 530004, Nanning, PR China

Received 13 May 2004; received in revised form 4 November 2004

Abstract

In this paper, a class of optimization problems with equality and inequality constraints is discussed. Firstly, the original problem is transformed to an associated simpler problem with only inequality constraints and a parameter. The later problem is shown to be equivalent to the original problem if the parameter is large enough (but finite), then a feasible descent SQP algorithm for the simplified problem is presented. At each iteration of the proposed algorithm, a master direction is obtained by solving a quadratic program (which always has a feasible solution). With two corrections on the master direction by two simple explicit formulas, the algorithm generates a feasible descent direction for the simplified problem and a height-order correction direction which can avoid the Maratos effect without the strict complementarity, then performs a curve search to obtain the next iteration point. Thanks to the new height-order correction technique, under mild conditions without the strict complementarity, the globally and superlinearly convergent properties are obtained. Finally, an efficient implementation of the numerical experiments is reported.

© 2004 Elsevier B.V. All rights reserved.

MSC: 90C30; 49M30

Keywords: General constrained optimization; SQP; Feasible descent algorithm; Superlinear convergence; Strict complementarity

∗ Project supported by the National Natural Science Foundation (No. 10261001) and Guangxi Science Foundation (Nos. 0236001, 0249003) of China.

∗ Corresponding author. Tel.: +86 771 323 9716; fax: +86 771 323 2084.

E-mail address: jianjb@gxu.edu.cn (J.-B. Jian).

0377-0427/ - see front matter © 2004 Elsevier B.V. All rights reserved.
1. Introduction

We consider the general constrained optimization problem with nonlinear equality and inequality constraints as follows

$$\begin{align*}
\text{min} & \quad f_0(x) \\
\text{s.t.} & \quad f_j(x) \leq 0, \quad j \in L_1 = \{1, ..., m'\}, \\
& \quad f_j(x) = 0, \quad j \in L_2 = \{m' + 1, \ldots, m\},
\end{align*}$$

(1.1)

where \( x \in \mathbb{R}^n \), \( f_j : \mathbb{R}^n \rightarrow \mathbb{R} \) \((j = 0, 1, \ldots, m)\) are smooth functions. We denote the feasible set \( X \) of problem (P) as follows

\[ X = \{ x \in \mathbb{R}^n : f_j(x) \leq 0, \quad j \in L_1; \quad f_j(x) = 0, \quad j \in L_2 \}. \]

Sequential quadratic programming (SQP) methods are derived from Wilson’ SOVLER algorithm (1963), and have proved highly effective for solving constrained optimization problems with smooth nonlinear objective function and constraints. An excellent survey on SQP methods was made by Boggs and Tolle in Ref.[2].

Generally, let \( x^k \) be a current iteration point for problem (P), SQP methods obtain a search direction \( d^k \) by solving the following quadratic program (QP)

$$\begin{align*}
\text{min} & \quad \nabla f_0(x^k)^T d + \frac{1}{2} d^T H_k d \\
\text{s.t.} & \quad f_j(x^k) + \nabla f_j(x^k)^T d \leq 0, \quad j \in L_1, \\
& \quad f_j(x^k) + \nabla f_j(x^k)^T d = 0, \quad j \in L_2,
\end{align*}$$

(1.2)

where \( H_k \in \mathbb{R}^{n \times n} \) is a positive definite matrix. Then perform a line search to get a steplength \( \tau_k \), and let the next iterate be \( x^{k+1} = x^k + \tau_k d^k \).

Most early SQP methods for dealing with constrained optimization problems having equality constraints focus on using penalty function (see, e.g., Refs. [9] and [30]). In particular, Han [9] used \( \ell_1 \) exact penalty function as the line search function (usually be called Merit function), and established its global convergence under slightly strong conditions (e.g., require that QP (1.2) always has a KKT point and the Lagrangian multipliers satisfy certain inequality). However, in fact, subproblem (1.2) may be inconsistent. In order to overcome this shortcoming, Powell [30] improved Han’s work by introducing an auxiliary variable, and solved an associated linear program in advance. Although Powell’s modification is showed efficient numerically, it is incomplete theoretically, Chamberlain [3] gave some examples to illuminate that such modification may cause cycling, and later, Chamberlain et al. [4] proposed the so-called watchdog technique to prevent cycling.

Besides the possibility that the QP subproblem may be inconsistent, there still exist other difficulties in these penalty function type SQP methods. One is the choice of the penalty function parameter, see Refs. [9,30,22,32], the techniques for choosing parameter is either theoretically incomplete in establishing the global convergence, or relatively complex and computationally expensive. Recently, De et al. [5] presented a simple scheme for updating the penalty parameter, and established an associated SQP algorithm which is proved to be globally and superlinearly convergent. However, its convergence rate strictly depend on the strict complementarity which is relatively strong and difficult for testing. This
condition also appears in the most recent works, such as Liu and Yuan [20], the authors proposed a robust algorithm for solving problem (P), in which the superlinear convergence rate achieves under certain conditions including the strict complementarity. Another difficulty exists in the traditional SQP methods is the Maratos effect, i.e., the unit step-size cannot be accepted even the iterate points are close to the optimum of problem (P), which is proposed firstly by Maratos in his Ph.D. Thesis [21].

On the other hand, many practical problems arise from engineering design and real-time applications strictly require certain feasibility of the iteration points, i.e., the iteration points must satisfy all or part of the constraints, which lead to the feasible direction methods. Early feasible direction methods were so-called first-order methods, only used the information of first derivatives, so, such algorithms converge linearly at best (see, e.g., [34,29]). In recent decades, many efforts have been made on the researches of the feasible direction methods (see Refs. [12,14,16,17,19,24,28,31]). In Ref. [24] Panier and Tits proposed a feasible SQP (FSQP) algorithm for optimization problems with inequality constraints, advantages and further studies of this algorithm can be found in e.g., [31,19]. At each iteration, FSQP algorithm generates a feasible decent direction by solving two quadratic subprograms, and obtains a correction direction used to avoid the Maratos effect by solving a linear least squares problem. Under certain conditions, FSQP algorithm is proved to be globally convergent and locally two-step superlinearly convergent. However, it seems that FSQP method cannot be used directly to deal with the optimization problems having equality constraints, and the computational cost is relatively expensive at each iteration, furthermore, the assumption of strict complementarity is also necessary. Most recently (2001), Lawrence and Tits [19] proposed another type feasible SQP algorithm, thereinto, the master direction is obtained by solving a perturbation form of QP subproblem, and revised by solving two equality constrained QPs, as a result, the computational effort of this algorithm is reduced comparing with traditional FSQP algorithms. However, such algorithm still cannot be extended directly to solve problem (P), though the authors suggested a way like [18], and the strict complementarity condition is necessary.

As mentioned above, the drawback which exists in many current FSQP methods is the requirement of the strict complementarity condition in establishing the convergent rate. Although some recently excellent SQP algorithms, e.g., see [1,6,26,27], obtained the superlinearly convergent rate under weaker conditions without the strict complementarity, these algorithms do not belong to feasible direction method, i.e., the iterative points do not strictly lie in the feasible set, and the objective function value is nonmonotone.

In fact, combining the idea of penalty function methods and feasible direction methods, Mayne and Polak [23] considered another way for solving problem (P) in 1976. Their scheme considers the following related family of simpler problems (SP) with only inequality constraints

$$\min \ F_c(x) = f_0(x) - c \sum_{j \in L_2} f_j(x)$$

subject to

$$f_j(x) \leq 0, \ j \in L = L_1 \cup L_2,$$

where parameter $c > 0$, which is updated by a simple procedure, the feasible set of problem (SPc) is denoted by

$$X^+ = \{x \in \mathbb{R}^n : f_j(x) \leq 0, \ j \in L\}.$$ 

Mayne and Polak showed that the simplified problem (SPc) is equivalent to the original problem if $c$ is sufficiently large (but finite), and then presented a feasible direction algorithm for solving problem (SPc). More details and advantages are discussed in [18], some further applications of this technique, such as Herskovits [10] and Jian [12], also show that Mayne and Polak’s scheme brings many advantages.
In this paper, combining the technique in [23] with the feasible direction methods, we present here a new SQP algorithm for general nonlinearly constrained optimization problems, and hope to overcome the above-mentioned shortcomings of penalty function type SQP methods, reduce the computational cost of traditional feasible direction SQP methods, establish the superlinear convergence without the strict complementarity, and avoid the Maratos effect.

Firstly, similar to the method in [23], we transform the original problem (P) to an associated simpler problem with only inequality constraints as the form of (1.3). As we will see (Lemma 2.2) that the KKT points of the original problem (P) and the inequality constrained problem (SPc) are equivalent if the parameter $c$ is sufficiently large.

Secondly, to construct a feasible QP subproblem and yield an improved direction, at each iteration of our algorithm, an $\varepsilon$-active constraint subset of $L$ is generated by a pivoting operation (POP) procedure. Based on the $\varepsilon$-active constraint subset, we construct a quadratic program corresponding to the simplified problem, which always has a feasible solution. By solving this quadratic program, a master direction is obtained, with a correction on the master direction by an explicit formula, the algorithm generates a feasible descent direction for the simplified problem. Furthermore, in order to avoid the Maratos effect, we construct a new height-order correction direction by introducing another explicit formula. Then a curve search is performed to obtain the next iteration point. Thanks to the new height-order correction technique, under mild conditions without the strict complementarity, the global and superlinear properties are obtained.

The main features of the proposed algorithm are summarized as follows:

- the objective function of the simplified problem is used directly as the search function;
- the algorithm is a feasible decent method for the simplified problems;
- the parameter is adjusted automatically only for a finite number of times;
- at each iteration, only one quadratic program needs to be solved;
- only the $\varepsilon$-active constrained functions and their gradients are considered in the quadratic program;
- the feasible decent direction and the height-order correction direction are generated by two simple explicit formulas.
- the superlinearly convergent rate is obtained without the strict complementarity.

The paper is organized as follows. In the next section, we present the details of our algorithm and discuss its properties. In Sections 3 and 4, we analyze the global, strong and superlinear convergent properties of the proposed method, respectively. In Section 5, some preliminary numerical tests are reported. Finally, a brief comment on the proposed algorithm is given in Section 6.

2. Algorithm

We assume that the following assumptions for problem (P) hold in this paper.

(H1) Functions $f_j$ ($j \in L^0 \triangleq \{0, 1, \ldots, m\}$) are all continuously differentiable.
(H2) The gradient vectors $\nabla f_j(x)$, $j \in I(x)$ are linearly independent for each feasible point $x$ of (SPc), i.e., $x \in X^+$, where the active set $I(x)$ is defined by

$$I(x) = I_1(x) \cup L_2, \quad I_1(x) = \{j \in L_1 : f_j(x) = 0\}.$$
For convenience of presentation, for a given subset $J \subseteq L$, we use the following notation throughout the remainder of the paper

$$f(x) = (f_1(x), \ldots, f_m(x))^T, \quad f_J(x) = (f_j(x), \ j \in J),$$

$$g_j(x) = \nabla f_j(x), \ j \in L^0, \quad g_J(x) = (g_j(x) = \nabla f_j(x), \ j \in J).$$

(2.1)

Let $x^k \in X^+$ be a given iterative point, we use the following pivoting operation to generate an $\epsilon$-active constraint subset $I_k \supseteq I(x^k)$, such that the matrix $g_{I_k}(x^k)$ is full of column rank.

**Pivoting operation (POP)**

*Step (i):* Select an initial parameter $\epsilon > 0$.

*Step (ii):* Generate the $\epsilon$-active constraint subset $I(x^k, \epsilon)$ by

$$I(x^k, \epsilon) = I_1(x^k, \epsilon) \cup L_2, \quad I_1(x^k, \epsilon) = \{ j \in L : -\epsilon \leq f_j(x^k) \leq 0 \}.\hspace{1cm} (2.2)$$

*Step (iii):* If $I_1(x^k, \epsilon) = \emptyset$ or $\det \left( (g_{I_1(x^k, \epsilon)}(x^k))^T g_{I_1(x^k, \epsilon)}(x^k) \right) \geq \epsilon$, set $I_k = I(x^k, \epsilon)$ and $\epsilon_k = \epsilon$, stop; otherwise set $\epsilon := \frac{1}{2} \epsilon$ and repeat Step (ii), where the matrix $g_{I(x^k, \epsilon)}(x^k)$ defined by (2.1).

In order to show the nice properties of the pivoting operation (POP) above, we present the following lemma, and its proof is similar to Lemmas 1.1 and 2.8 in Ref. [8], so is omitted here.

**Lemma 2.1.** Suppose that (H1), (H2) hold, and let $x^k \in X^+$. Then

(i) The pivoting operation (POP) can be finished in a finite number of computations, i.e., there is no infinite times of loop between Step (ii) and Step (iii).

(ii) If a sequence $\{x^k\}$ of points has an accumulation point, then there is an $\bar{\epsilon} > 0$ such that the associated sequence $\{\epsilon_k\}$ of parameters generated by (POP) satisfies $\epsilon_k \geq \bar{\epsilon}$ for all $k$.

Denote matrix and multiplier vector as

$$N_k = g_{I_k}(x^k), \quad u(x^k) = -(N_k^T N_k)^{-1} N_k^T g_0(x^k) = (u_j(x^k), \ j \in I_k),$$

(2.3)

then we have the following lemma.

**Lemma 2.2.** If parameter $c_k > |u_j(x^k)|, \ j \in L_2$, then $(x^k, \mu^k)$ is a KKT pair of problem (P) if and only if $(x^k, \lambda^k)$ is a KKT pair of problem (SPc_k), where $\mu^k, \lambda^k$ satisfy

$$\mu_j^k = \lambda_j^k, \ j \in I_1(x^k, \epsilon_k), \quad \mu_j^k = \lambda_j^k - c_k, \ j \in L_2.\hspace{1cm} (2.4)$$

**Proof.** (i) Suppose that $(x^k, \mu^k)$ is a KKT pair of problem (P), then $x^k \in X \subseteq X^+$, and from KKT conditions we have

$$g_0(x^k) + \sum_{j \in I_k} \mu_j^k g_j(x^k) = 0, \hspace{1cm} (2.5)$$

$$\mu_j^k f_j(x^k) = 0, \quad \mu_j^k \geq 0, \ j \in I_1(x^k, \epsilon_k), \hspace{1cm} (2.6)$$

$$f_j(x^k) \leq 0, \ j \in L_1, \quad f_j(x^k) = 0, \ j \in L_2.\hspace{1cm} (2.7)$$
Combining the definition of $F_{c_k}(x^k) = f_0(x^k) - c_k \sum_{j \in L_2} f_j(x^k)$, we can deduce
\begin{equation}
\nabla F_{c_k}(x^k) + \sum_{j \in I_1(x^k, \epsilon_k)} \mu_j^k g_j(x^k) + \sum_{j \in L_2} (\mu_j^k + c_k) g_j(x^k) = 0,
\end{equation}
\begin{equation}
\mu_j^k f_j(x^k) = 0, \quad \mu_j^k \geq 0, \quad j \in I_1(x^k, \epsilon_k), \quad (\mu_j^k + c_k) f_j(x^k) = 0, \quad j \in L_2,
\end{equation}
\begin{equation}
f_j(x^k) \leq 0, \quad j \in L_1, \quad f_j(x^k) = 0, \quad j \in L_2.
\end{equation}
Furthermore, we have from (2.5) and (2.6)
\begin{equation}
g_0(x^k) + N_k \mu^k = 0, \quad \mu^k = -(N_k^T N_k)^{-1} N_k^T g_0(x^k) = u(x^k).
\end{equation}
In view of $c_k > |u_j(x^k)|$, $j \in L_2$, so
\begin{equation}
\mu_j^k + c_k = u_j(x^k) + c_k > 0, \quad j \in L_2,
\end{equation}
this together with (2.8), (2.9), (2.10) and (2.4) shows that $(x^k, \lambda^k)$ is a KKT pair of problem (SP$c_k$).
\begin{itemize}
  \item[(ii)] Suppose that $(x^k, \lambda^k)$ is a KKT pair of problem (SP$c_k$), then
\end{itemize}
\begin{equation}
\nabla F_{c_k}(x^k) + \sum_{j \in I_1(x^k, \epsilon_k)} \lambda_j^k g_j(x^k) + \sum_{j \in L_2} \lambda_j^k g_j(x^k) = 0,
\end{equation}
\begin{equation}
\lambda_j^k f_j(x^k) = 0, \quad \lambda_j^k \geq 0, \quad j \in I_1(x^k, \epsilon_k) \cup L_2, \quad f_j(x^k) \leq 0, \quad j \in L.
\end{equation}
So
\begin{equation}
g_0(x^k) + \sum_{j \in I_1(x^k, \epsilon_k)} \lambda_j^k g_j(x^k) + \sum_{j \in L_2} (\lambda_j^k - c_k) g_j(x^k) = 0.
\end{equation}
\begin{equation}
\lambda_j^k f_j(x^k) = 0, \quad \lambda_j^k \geq 0, \quad j \in I_1(x^k, \epsilon_k) \cup L_2, \quad f_j(x^k) \leq 0, \quad j \in L.
\end{equation}
Let us denote
\begin{equation}
\mu_j^k = \lambda_j^k, \quad j \in I_1(x^k, \epsilon_k), \quad \mu_j^k = \lambda_j^k - c_k, \quad j \in L_2, \quad \mu^k = (\mu_j^k, j \in I_k).
\end{equation}
So we have from (2.15) $g_0(x_k) + N_k \mu^k = 0$, and
\begin{equation}
\mu^k = -(N_k^T N_k)^{-1} N_k^T g_0(x^k) = u(x^k), \quad \lambda_j^k = \mu_j^k + c_k = u_j(x^k) + c_k > 0, \quad j \in L_2.
\end{equation}
Therefore, from (2.16) and (2.17), we have $f_j(x^k) = 0, \quad j \in L_2$, this shows $x^k \in X$, so, combining (2.15) and (2.16), we can conclude that $(x^k, \mu^k)$ is a KKT pair of problem (P). The whole proof is completed. □

The nice properties given in Lemma 2.2 motivate us to propose an effective algorithm for solving the general constrained optimization problem (P) based on a sequential optimization problems (SP$c_k$) with only inequality constraints. In this paper, we apply the idea of FSQP methods to (SP$c_k$), let point
\( x^k \in X^+ \) and \( I_k \) be the corresponding subset generated by (POP), for the \( k \)th simplified problem \((SP_{ck})\), the corresponding quadratic program \((QPC_k)\) is given below

\[
(QPC_k) \quad \begin{align*}
\min & \quad \nabla F_{ck}(x^k)^T d + \frac{1}{2} d^T H_k d \\
\text{s.t.} & \quad f_j(x^k) + g_j(x^k)^T d \leq 0, \quad j \in I_k,
\end{align*}
\]

(2.18)

where \( H_k \in \mathbb{R}^{n \times n} \) is a positive definite matrix.

It is obvious that the subproblem \((QPC_k)\) always has a feasible solution \( d = 0 \), while \((QPC_k)\) is a strict convex program, so it always has a (unique) solution. Furthermore, \( d^k_0 \) is a solution of \((QPC_k)\) if and only if it is a KKT point of \((QPC_k)\), i.e., there exists a corresponding KKT multiplier \( \lambda^k_{I_k} = (\lambda^k_j, j \in I_k) \) such that

\[
\nabla F_{ck}(x^k) + H_k d^k_0 + \sum_{j \in I_k} \lambda^k_j g_j(x^k) = 0,
\]

\[
f_j(x^k) + g_j(x^k)^T d^k_0 \leq 0, \quad \lambda^k_j \geq 0, \quad \lambda^k_j \left( f_j(x^k) + g_j(x^k)^T d^k_0 \right) = 0, \quad \forall j \in I_k.
\]

(2.19)

Suppose that vector \( \mu^k \) is defined by (2.4), then the KKT conditions (2.19) can be rewritten as

\[
g_0(x^k) + H_k d^k_0 + \sum_{j \in I_k} \mu^k_j g_j(x^k) = 0,
\]

\[
\mu^k_j \geq 0, \quad \mu^k_j (f_j(x^k) + g_j(x^k)^T d^k_0) = 0, \quad \forall j \in I_1(x^k, \varepsilon_k),
\]

\[
(\mu^k_j + c_k) \geq 0, \quad (\mu^k_j + c_k) (f_j(x^k) + g_j(x^k)^T d^k_0) = 0, \quad \forall j \in L_2,
\]

\[
f_j(x^k) + g_j(x^k)^T d^k_0 \leq 0, \quad j \in I_1(x^k, \varepsilon_k) \cup L_2.
\]

(2.20)

So one has immediately the following lemma from (2.19) and Lemma 2.2.

**Lemma 2.3.** If \( d^k_0 = 0 \), then \( x^k \) is a KKT point of problem \((SP_{ck})\). Furthermore, if \( c_k > |u_j(x^k)|, j \in L_2 \), then \( x^k \) is a KKT point of problem \((P)\).

By solving \((QPC_k)\), we get a solution \( d^k_0 \). However, \( d^k_0 \) may not be a feasible direction of the \( k \)th simplified problem \((SP_{ck})\) at the feasible point \( x^k \in X^+ \). So, a suitable strategy must be carried out to generate a feasible direction, for example, solving another QP (e.g., see [24]) or a system of linear equations (see Refs. [25] and [7]). In order to reduce the amount of computations, we use an explicit formula to update \( d^k_0 \) as follows.

\[
d^k = d^k_0 - \delta(x^k) N_k (N_k^T N_k)^{-1} e_k,
\]

(2.21)

with \( e_k = (1, \ldots, 1)^T \in \mathbb{R}^{|I_k|} \) and

\[
\delta(x^k) = \frac{\|d^k_0\| (d^k_0)^T H_k d^k_0}{2 \|e_k^T \pi(x^k)\| \cdot \|d^k_0\| + 1}, \quad \pi_j(x^k) = \begin{cases} u_j(x^k), & j \in I_1(x^k), \\ u_j(x^k) - c_k, & j \in L_2, \end{cases}
\]

\[
\pi(x^k) = (\pi_j(x^k), j \in I_k).
\]

(2.22)
Using the definition of $\nabla F_{c_k}(x^k)$ (1.3) and (2.3), it is not difficult to show\n\begin{equation}
\pi(x^k) = -(N_k^T N_k)^{-1} N_k^T \nabla F_{c_k}(x^k) .
\end{equation} (2.23)

Combining the KKT conditions (2.19) and the definition of $d^k$ (2.21) as well as (2.23), we can deduce the following relations.
\begin{equation}
g_I(x^k)^T d^k = N_k^T d^k = N_k^T d_0^k - \delta(x^k) e_k \leq - f_I(x^k) - \delta(x^k) e_k .
\end{equation} (2.24)
\begin{equation}
\nabla F_{c_k}(x^k)^T d_0^k = -(d_0^k)^T H_k d_0^k - (\lambda_{I_k}^T)^T N_k^T d_0^k
\end{equation}
\begin{equation}
= -(d_0^k)^T H_k d_0^k + (\lambda_{I_k}^T)^T f_I(x^k) ,
\end{equation} (2.25)
\begin{equation}
\nabla F_{c_k}(x^k)^T d^k = \nabla F_{c_k}(x^k)^T d_0^k - \delta(x^k) \nabla F_{c_k}(x^k)^T N_k (N_k^T N_k)^{-1} e_k
\end{equation}
\begin{equation}
\leq -(d_0^k)^T H_k d_0^k + (\lambda_{I_k}^T)^T f_I(x^k) + \delta(x^k) |\pi(x^k)^T e_k|
\end{equation}
\begin{equation}
\leq -(d_0^k)^T H_k d_0^k + (\lambda_{I_k}^T)^T f_I(x^k) + \frac{1}{2} (d_0^k)^T H_k d_0^k
\end{equation}
\begin{equation}
= -\frac{1}{2} (d_0^k)^T H_k d_0^k + (\lambda_{I_k}^T)^T f_I(x^k) \leq -\frac{1}{2} (d_0^k)^T H_k d_0^k ,
\end{equation}
which implies
\begin{equation}
\nabla F_{c_k}(x^k)^T d^k \leq -\frac{1}{2} (d_0^k)^T H_k d_0^k .
\end{equation} (2.26)

If the iterative $x^k$ is not a KKT point of problem (SP$c_k$), then $d_0^k \neq 0$, furthermore, from formulas (2.24) and (2.26), one can conclude that $d^k$ is a feasible descent direction of problem (SP$c_k$) at feasible point $x^k$.

The last issue to be addressed here is to overcome the Maratos effect. So a “height-order” correction direction must be introduced by a suitable technique, e.g., in [24], a linear least squares problem needs to be solved. In this paper, we introduce a new explicit correction direction $d_1^k$ as follows, which new technique can avoid the assumption of the strict complementarity used in many existing references (such as [20,24] and [19]).
\begin{equation}
d_1^k = -N_k(N_k^T N_k)^{-1} (\|d_0^k\| \kappa e_k + \tilde{f}_I(x^k + d^k)) ,
\end{equation} (2.27)
where the constant $\kappa \in (2, 3)$ and vector
\begin{equation}
\tilde{f}_I(x^k + d^k) = f_I(x^k + d^k) - f_I(x^k) - g_I(x^k)^T d^k .
\end{equation} (2.28)

At the end of this section, we give the details of our algorithm as follows.

**Algorithm A.**

**Parameters:** $\varepsilon^{-1} > 0$, $\kappa \in (2, 3)$, $\varepsilon \in (0, 0.5)$, $\beta \in (0, 1)$, $\ell, r, c^{-1} > 0$.

**Data:** $x^0 \in X^+$, a symmetric positive definite matrix $H_0 \in R^{n \times n}$.

**Step 0: Initialization:** Let $k := 0$.

**Step 1: Pivoting operation:** Set parameter $\varepsilon = \varepsilon_{k-1}$, generate an active constraint set $I_k$ by the POP and let $\varepsilon_k$ be the corresponding termination parameter.
Step 2: Update parameter $c_k$: Compute $c_k$, $(k = 0, 1, 2, \ldots)$ by
\[
c_k = \begin{cases} 
\max \{s_k, c_{k-1} + r\}, & \text{if } s_k > c_{k-1}; \\
\quad c_{k-1}, & \text{if } s_k \leq c_{k-1}.
\end{cases}
\]

$\quad s_k = \max \{|u_j(x^k)|, j \in L_2\} + \ell.$ \hfill (2.29)

Step 3: Solve $(\text{QP}_{c_k})$: Solve $(\text{QP}_{c_k})$ (2.18) to get a (unique) solution $d^0_k$ and the corresponding KKT multiplier vector $\lambda^k_i = (\lambda^k_j, j \in I_k)$. If $d^0_k = 0$, then $x^k$ is a KKT point of problem (P) and stop; otherwise, enter Step 4.

Step 4: Generate search directions: Compute the improved direction $d^k$ by formula (2.21) and the height-order updating direction $d^1_k$ by (2.27).

Step 5: Perform curve search: Compute the step size $\tau_k$, the first number $\tau$ of the sequence $\{1, \beta, \beta^2, \ldots\}$ satisfying
\[
F_{c_k}(x^k + \tau d^k + \tau^2 d^1_k) \\
\leq F_{c_k}(x^k) + \tau \nabla F_{c_k}(x^k)^T d^k, \quad f_j(x^k + \tau d^k + \tau^2 d^1_k) \leq 0, \; \forall j \in L.
\]

(2.30)

Step 6: Updates: Compute a new symmetric positive definite matrix $H_{k+1}$, set $x^{k+1} = x^k + \tau_k d^k + \tau^2_k d^1_k$ and $k := k + 1$, go back to Step 1.

3. Global convergence analysis

In this section, we will establish the global convergence of the proposed Algorithm A. If the solution $d^0_k$ generated at Step 3 equals zero, then Algorithm A stops at $x^k$, we know from Lemma 2.3 that $x^k$ is a KKT point of the original problem (P). And if $d^0_k \neq 0$, in view of (2.24) and (2.26), one knows that $d^k$ is a feasible descent direction of the simplified problem $(\text{SP}_{c_k})$ at point $x^k$, which implies that the curve search (2.30) can stop in a finite number of computations, so the proposed Algorithm A is well defined.

We further assume that an infinite sequence $\{x^k\}$ of points is generated by Algorithm A, and the consequent task is to show that every accumulation point $x^*$ of $\{x^k\}$ is a KKT point of problem (P). Firstly, we make the following assumption which holds in the rest of this paper.

(H3) The sequence $\{H_k\}$ of matrices is uniformly positive definite, i.e., there exist two positive constants $a$ and $b$ such that
\[
a\|d\|^2 \leq d^T H_k d \leq b\|d\|^2, \quad \forall d \in \mathbb{R}^n, \; \forall k.
\]
\hfill (3.1)

**Lemma 3.1.** If the sequence $\{x^k\}$ is bounded, then there exists a $k_0 > 0$, such that $c_k \equiv c_{k_0}$, for all $k \geq k_0$.

**Proof.** We suppose by contradiction that $c_k$ increases indefinitely, then Lemma 2 in [23] gives that $\{x^k\}$ has no accumulation points, this is a contradiction, so the lemma holds. \hfill □

Due to Lemma 3.1, we always assume that $c_k \equiv c$ for all $k$ in the rest of this paper.
Lemma 3.2. Suppose that assumptions (H1)–(H3) hold, and \( \{x^k\} \) is bounded. Then

(i) There exists a constant \( \zeta > 0 \) such that \( \| (N_k^T N_k)^{-1} \| \leq \zeta, \forall k \).
(ii) The sequences \( \{d_k^0\}_{k=1}^{\infty}, \{d_k\}_{k=1}^{\infty} \) and \( \{d_k^1\}_{k=1}^{\infty} \) are all bounded.

Proof. (i) By contradiction, suppose that sequence \( \{\| (N_k^T N_k)^{-1} \|\} \) is unbounded, then there exists an infinite subset \( K \), such that

\[
\| (N_k^T N_k)^{-1} \| \to \infty, \quad k \in K.
\] (3.2)

In view of the boundedness of \( \{x^k\} \) and \( I_k \) being a subset of the finite set \( L = \{1, \ldots, m\} \) as well as Lemma 2.1, we know that there exists an infinite index set \( K' \subseteq K \) such that

\[
x^k \to \tilde{x}, \quad I_k \equiv I', \quad \forall k \in K', \quad \det(N_k^T N_k) \geq \bar{\varepsilon}, \quad \varepsilon_k \geq \bar{\varepsilon}, \quad \forall k.
\] (3.3)

Therefore,

\[
\lim_{k \in K'} N_k^T N_k = g_{I'}(\tilde{x})^T g_{I'}(\tilde{x}), \quad \det(g_{I'}(\tilde{x})^T g_{I'}(\tilde{x})) \geq \bar{\varepsilon} > 0.
\]

Thus, we have \( \| (N_k^T N_k)^{-1} \| \to \| (g_{I'}(\tilde{x})^T g_{I'}(\tilde{x}))^{-1} \| \), this contradict (3.2), so the first conclusion (i) follows.

(ii) We begin by showing that the sequence \( \{d_k^0\}_{k=1}^{\infty} \) is bounded.

Due to the fact that \( \tilde{d}_k \) is a feasible solution of (QPC_k) (2.18) and \( d_k^0 \) is an optimal solution, we have

\[
\nabla F_c(x^k)^T d_k^0 + \frac{1}{2} (d_k^0)^T H_k d_k^0 \leq \nabla F_c(x^k)^T \tilde{d}_k + \frac{1}{2} (\tilde{d}_k)^T H_k \tilde{d}_k.
\]

By Lemma 3.1, part (i) and the boundedness of \( \{x^k\} \), we know that \( \{\nabla F_c(x^k)\} \) and \( \{\tilde{d}_k\} \) are all bounded, i.e., there exists a constant \( \bar{c} > 0 \) such that \( \| \nabla F_c(x^k) \| \leq \bar{c} \) and \( \| \tilde{d}_k \| \leq \bar{c} \) for all \( k \), thus from the inequality above and (3.1), we get

\[
-\bar{c} \| d_0^k \| + \frac{1}{2} a \| d_0^k \|^2 \leq \bar{c}^2 (1 + \frac{1}{2} b),
\]

which implies that \( \{d_0^k\} \) is bounded. Taking into account formulas (2.21) and (2.27), we can easily obtain the boundedness of \( \{d_k\} \) and \( \{d_k^1\} \) by employing the result of part (i), thus part (ii) holds. \( \square \)

Theorem 3.1. If assumptions (H1)–(H3) are satisfied, and \( \{x^k\} \) is bounded, then

(i) \( \lim_{k \to \infty} d_k^0 = \lim_{k \to \infty} d_k = \lim_{k \to \infty} d_k^1 = 0. \)
(ii) \( \lim_{k \to \infty} \| x^{k+1} - x^k \| = 0. \)

Proof. To prove part (i), we firstly show that \( \lim_{k \to \infty} d_0^k = 0. \)
We suppose by contradiction that $\lim_{k \to \infty} d^k_0 \neq 0$, then there exist an infinite index set $K$ and a constant $\sigma > 0$ such that $\|d^k_0\| \geq \sigma$ holds for all $k \in K$. Taking notice of the boundedness of $\{x^k\}$, by taking a subsequence if necessary, we may suppose that

$$x^k \to \tilde{x}, \quad I_k \equiv I', \quad k \in K.$$  

In the following discussion, we assume that $k \in K$ is sufficient large and $\tau > 0$ is sufficient small. Firstly, we will show that there exists a constant $\bar{\tau} > 0$ such that the step size $\tau_k \geq \bar{\tau}$ for $k \in K$.

**Analyze the first search inequality of (2.30):** Using Taylor expansion, combining (2.26), (3.1) and Lemma 3.2(ii), one has

$$F_c(x^k + \tau d^k + \tau^2 d^k_1) = F_c(x^k) + \tau \nabla F_c(x^k)^T d^k + (1 - \alpha) \tau \nabla F_c(x^k)^T d^k + o(\tau)$$

$$\leq F_c(x^k) + \tau \nabla F_c(x^k)^T d^k - \frac{1}{2} (1 - \alpha) \tau (d^k_0)^T H_k d^k_0 + o(\tau)$$

$$\leq F_c(x^k) + \tau \nabla F_c(x^k)^T d^k - \frac{1}{2} (1 - \alpha) \tau \|d^k_0\|^2 + o(\tau)$$

$$\leq F_c(x^k) + \tau \nabla F_c(x^k)^T d^k - \frac{1}{2} (1 - \alpha) \tau \sigma^2 + o(\tau).$$

Which shows that the first inequality of (2.30) holds for $k \in K$ and $\tau > 0$ small enough.

**Analyze the later inequalities of (2.30):** If $j \notin I(\tilde{x})$, i.e., $f_j(\tilde{x}) < 0$, from the continuity of function $f_j(x)$ and the boundedness of $\{d^k, d^k_1\}$, we know $f_j(x^k + \tau d^k + \tau^2 d^k_1) \leq 0$ holds for $k \in K$ large enough and $\tau > 0$ small enough.

Let $j \in I(\tilde{x})$, i.e., $f_j(\tilde{x}) = 0$, then $j \in I_k$ by Lemma 2.1(ii), similarly, using Taylor expansion and (2.24), we have

$$f_j(x^k + \tau d^k + \tau^2 d^k_1) = f_j(x^k) + \tau g_j(x^k)^T d^k + o(\tau)$$

$$\leq f_j(x^k) - \tau f_j(x^k) - \tau \delta(x^k) + o(\tau).$$

On the other hand, formula (2.22) gives

$$\delta(x^k) = \frac{(d^k_0)^T H_k d^k_0}{2|e_k^T \pi(x^k)| + \frac{1}{\|d^k_0\|}} \geq \frac{a \|d^k_0\|^2}{2|e_k^T \pi(x^k)| + \frac{1}{\sigma}} \geq \bar{a} \|d^k_0\|^2 \geq \bar{a} \sigma^2.$$

Thus

$$f_j(x^k + \tau d^k + \tau^2 d^k_1) \leq (1 - \tau) f_j(x^k) - \tau \bar{a} \sigma^2 + o(\tau) \leq 0$$

holds for $k \in K$ large enough and $\tau > 0$ small enough.

Summarizing the analysis above, we conclude that there exists a $\bar{\tau} > 0$ such that $\tau_k \geq \bar{\tau}$ for all $k \in K$.

Secondly, we use $\tau_k \geq \bar{\tau} > 0$ to bring a contradiction. Combining Lemma 3.1, the first inequality of (2.30), formulas (2.26) and (3.1), we have

$$F_c(x^{k+1}) \leq F_c(x^k) + \alpha \tau_k \nabla F_c(x^k)^T d^k \leq F_c(x^k) - \frac{1}{2} \alpha \tau_k (d^k_0)^T H_k d^k_0$$

$$\leq F_c(x^k) - \frac{1}{2} \alpha \tau_k \|d^k_0\|^2, \quad \forall k.$$
This shows that \( \{ F_c(x^k) \} \) is decreasing, combining \( \lim_{k \in K} F_c(x^k) = F_c(\bar{x}) \), one knows \( \lim_{k \to \infty} F_c(x^k) = F_c(\bar{x}) \). On the other hand, one also has
\[
F_c(x^{k+1}) \leq F_c(x^k) - \frac{1}{2} a\bar{x}\bar{\sigma}^2, \quad \forall k \in K.
\]
Passing to the limit \( k \in K \) and \( k \to \infty \) in this inequality, we have \( -\frac{1}{2} a\bar{x}\bar{\sigma}^2 \geq 0 \), which is a contradiction, thus \( \lim_{k \to \infty} d^k_0 = 0 \), furthermore, \( \lim_{k \to \infty} d^k = 0 \) and \( \lim_{k \to \infty} d^k_I = 0 \) follow from formulas (2.21) and (2.27) as well as Lemma 3.2(i).

(ii) From part (i), we have
\[
\lim_{k \to \infty} \| x^{k+1} - x^k \| = \lim_{k \to \infty} \| \tau_k d^k + \tau_k^2 d_1^k \| \leq \lim_{k \to \infty} (\| d^k \| + \| d_1^k \|) = 0.
\]
So part (ii) follows, and the whole proof is completed. \( \square \)

Now, it is sufficient for us to establish the following globally convergent theorem of the proposed Algorithm A.

**Theorem 3.2.** Suppose that assumptions (H1)–(H3) hold and \( \{ x^k \} \) is bounded, then Algorithm A either stops at a KKT point \( x^k \) of problem (P) in a finite number of steps or generates an infinite sequence \( \{ x^k \} \) of points such that each accumulation point \( x^* \) is a KKT point of problem (P). Furthermore, there exists an index set \( K \) such that \( \{(x^k, \lambda^k) : k \in K\} \) and \( \{(x^k, \mu^k) : k \in K\} \) converge to the KKT pair \( (x^*, \lambda^*) \) of the simplified problem \( (SPc) \) and the KKT pair \( (x^*, \mu^*) \) of the original problem (P), respectively, where \( \lambda^* = (\lambda^*_I, 0_{L\setminus I}) \) and \( \mu^* = (\mu^*_I, 0_{L\setminus I}) \) defined by (2.4).

**Proof.** Note that the index sets \( I_k \) are all subset of \( \{1, \ldots, m\} \), we can suppose without loss of generality that there exists an infinite subset \( K \), such that
\[
x^k \to x^*, \quad I_k \equiv I', \quad k \in K.
\]
Using the KKT condition (2.19), we can deduce
\[
\nabla F_c(x^k) + H_k d^k_0 + N_k \lambda^*_I = 0.
\]
In view of the results given in Lemma 3.2(i) and Theorem 3.1(i), the above equality shows that
\[
\lambda^*_I = \lambda^*_I' = -(N_k^T N_k)^{-1} N_k^T (\nabla F_c(x^k) + H_k d^k_0) \to -(N_*^T N_*)^{-1} N_*^T \nabla F_c(x^*) \overset{\text{def}}{=} \lambda^*_I', \quad k \in K,
\]
where \( N_* \overset{\text{def}}{=} g_I'(x^*) = (g_j(x^*), \quad j \in I') \).

By passing to the limit \( k \in K \) and \( k \to \infty \) in (2.19), we have
\[
\nabla F_c(x^*) + N_* \lambda^*_I = 0, \quad f_I'(x^*) = 0, \quad \lambda^*_I \geq 0, \quad f_I'(x^*)^T \lambda^*_I = 0,
\]
which shows that \( (x^*, \lambda^*) \) with \( \lambda^* = (\lambda^*_I, 0_{L\setminus I}) \) is a KKT pair of problem \( (SPc) \). From the definition of \( c_k \) at Step 2 (\( c_k \equiv c \) for all \( k \) is large enough by Lemma 3.1), we know \( c > \max\{|u_j(x^*)|, \quad j \in L_2\} \). So from Lemma 2.2, we can conclude \( (x^*, \mu^*) \) is a KKT pair of problem (P) with \( \mu^*_j = \lambda^*_j, \quad j \in I' \setminus L_2; \quad \mu^*_j = \lambda^*_j - c, \quad j \in L_2; \quad \mu^*_j = 0, \quad j \in L \setminus I' \). Obviously, \( \lim_{k \in K} (x^k, \lambda^k) = (x^*, \lambda^*) \) and \( \lim_{k \in K} (x^k, \mu^k) = (x^*, \mu^*) \). The proof is completed. \( \square \)
4. Strong and superlinear convergence

We begin this section by stating the following assumption (H4). Next, under mild conditions without the strict complementarity, we will discuss the strong and superlinear convergence of the proposed Algorithm A.

(H4) (i) The functions \( f_j(x) \) (\( j \in L^0 \)) are all twice continuously differentiable in the feasible set \( X^+ \);

(ii) The sequence \( \{x^k\} \) generated by Algorithm A is bounded, and possesses an accumulation point \( x^* \) (by Theorem 3.2, \( x^* \) is a KKT point of problem (P), suppose its corresponding multipliers are \( \mu^* \)), such that the KKT pair \( (x^*, \mu^*) \) of problem (P) satisfies the strong second-order sufficiency conditions, i.e.,

\[
d^T \nabla^2_{xx} L(x^*, \mu^*) \sigma > 0, \quad \forall d \in \mathbb{R}^n : d \neq 0, \quad g_j(x^*)^T d = 0, \quad j \in I^+,
\]

where

\[
\nabla^2_{xx} L(x^*, \mu^*) = \nabla^2 f_0(x^*) + \sum_{j \in L} \mu_j^* \nabla^2 f_j(x^*), \quad I^+ = \{ j \in L_1 : \mu_j^* > 0 \} \cup L_2.
\]

Remark 4.1. Similar to the proof of Lemma 2.2, we can conclude that \( (x^*, \lambda^*) \) with

\[
\lambda_j^* = \mu_j^*, \quad j \in I_1(x^*); \quad \lambda_j^* = \mu_j^* + c, \quad j \in L_2; \quad \lambda_j^* = \mu_j^* = 0, \quad j \in L_1 \setminus I_1(x^*)
\]

is a KKT pair of (SPc), and \( \lambda_j^* = \mu_j^* + c > 0, \quad j \in L_2 \). Therefore

\[
\{ j \in L : \lambda_j^* > 0 \} = \{ j \in L_1 : \mu_j^* > 0 \} \cup L_2,
\]

which implies that the KKT pair \( (x^*, \lambda^*) \) of problem (SPc) also satisfies the strong second-order sufficiency conditions, i.e.,

\[
d^T \nabla^2_{xx} L_c(x^*, \lambda^*) d > 0, \quad \forall d \in \mathbb{R}^n : d \neq 0, \quad g_j(x^*)^T d = 0, \quad j \in \tilde{I}^+,
\]

where

\[
\nabla^2_{xx} L_c(x^*, \lambda^*) = \nabla^2 F_c(x^*) + \sum_{j \in L} \lambda_j^* \nabla^2 f_j(x^*), \quad \tilde{I}^+ = \{ j \in L : \lambda_j^* > 0 \}.
\]

Theorem 4.1. If assumptions (H1)–(H4) are all satisfied, then

(i) \( \lim_{k \to \infty} x^k = x^* \), and Algorithm A is said to be strongly convergent in this sense.

(ii) \( \lim_{k \to \infty} \lambda^k = \lambda^* \), \( \lim_{k \to \infty} \mu^k = \mu^* \).

Proof. (i) From Theorem 1.2.5 of [13] or Proposition 4.1 of [28], we can obtain that \( x^* \) is an isolated KKT point of the simplified problem (SPc)(1.3) by employing the strong second-order sufficiency conditions (H4)(ii). Thus by Theorem 3.2, \( x^* \) is an isolated accumulation point of \( \{x^k\} \), and this together with \( \lim_{k \to \infty} \|x^{k+1} - x^k\| = 0 \) implies \( \lim_{k \to \infty} x^k = x^* \) (see Theorem 1.1.5 in [13] or [28]).

(ii) We assume by contradiction that \( \lim_{k \to \infty} \lambda^k \neq \lambda^* \), then there exist an infinite set \( K \) and a constant \( \bar{a} > 0 \), such that

\[
\|\lambda^k - \lambda^*\| \geq \bar{a}, \quad k \in K.
\]
Hence there exists an infinite set $K' \subseteq K$, such that

$$x^k \to x^*, \ I_k \equiv I', \ \|x^k_{J'} - x^*_J\| \geq \bar{a}, \ k \in K',$$

but from the proof of Theorem 3.2, we can see $x^k_{J'} \to x^*_J$ (since the KKT multiplier is unique), $k \in K'$, which is a contradiction, so the whole proof is finished. \(\square\)

**Lemma 4.1.** Under above-mentioned assumptions (H1)–(H4), the following relations hold

$$\|d^k\| \sim \|d_0^k\|, \ \|d^k - d_0^k\| = O(\|d_0^k\|^2), \ \|d_1^k\| = O(\|d_0^k\|^2), \ \|d_1^k\| = O(\|d^k\|^2). \ (4.2)$$

$$\tilde{I}^+ \subseteq J_k \overset{def}{=} \{i \in I_k : f_i(x^k) + g_i(x^k)^T d_0^k = 0\} \subseteq I(x^*) \subseteq I_k. \ (4.3)$$

**Proof.** From (2.21), (2.27), Lemma 3.2(i), Theorems 3.1 and 4.1, it is not difficult to prove (4.2).

For (4.3), one first gets $J_k \subseteq I(x^*) \subseteq I_k$ from $\lim_{k \to \infty} (x_k, d_{0k}) = (x^*, 0)$ and Lemma 2.1(ii); furthermore, one has $\lim_{k \to \infty} \tilde{x}_{I^+}^k = \tilde{x}_{I^+}^* > 0$ from Theorem 4.1(ii), so $\tilde{x}_{I^+}^k > 0$ and $\tilde{I}^+ \subseteq J_k$ holds for $k$ large enough. \(\square\)

In order to obtain the superlinearly convergent rate of the proposed algorithm, first of all, we should guarantee that the size $\tau_k \equiv 1$ is accepted by the curve search for $k$ large enough. For this purpose, the following additional assumption (H5) is necessary.

(H5) Suppose that $\|(\nabla_{xx}^2 L_c(x^k, \tilde{x}_{I^+_k}) - H_k)d^k\| = o(\|d^k\|)$, where $\nabla_{xx}^2 L_c(x^k, \tilde{x}_{I^+_k}) = \nabla^2 F_c(x^k) + \sum_{j \in I_k} \tilde{x}_{I^+_k}^j \nabla^2 f_j(x^k) = \nabla^2 L(x^k, \tilde{\mu}_{I^+_k})$.

**Remark 4.2.** In fact, we can get an equivalent statement of the above assumption as follows:

$$\|(\nabla_{xx}^2 L_c(x^k, \tilde{x}_{I^+_k}) - H_k)d^k\| = o(\|d^k\|) \iff \|(\nabla_{xx}^2 L_c(x^*, \tilde{x}_{I^+_k}^*) - H_k)d^k\| = o(\|d^k\|).$$

**Theorem 4.2.** Under all above assumptions (H1)–(H5), the step size in Algorithm A always equals one, i.e., $\tau_k \equiv 1$, for $k$ large enough.

**Proof.** Firstly, we verify the inequalities of the second part of (2.30) hold for $\tau = 1$ and $k$ large enough.

For $j \notin I(x^*)$, i.e., $f_j(x^*) < 0$, by using $(x^k, d^k, d_1^k) \to (x^*, 0, 0) \ (k \to \infty)$, we can conclude $f_j(x^k + d^k + d_1^k) \leq 0$ holds for $k$ large enough.

For $j \in I(x^*) \subseteq I_k$, (2.24) and (2.27) give

$$g_j(x^k)^T d^k = g_j(x^k)^T d_0^k - \delta(x^k),$$

$$g_j(x^k)^T d_1^k = -\|d_0^k\|^k - f_j(x^k + d^k) + f_j(x^k) + g_j(x^k)^T d^k, \ j \in I_k. \ (4.4)$$
Expanding $f_j$ around $x^k + d^k$, combining (4.2) and (4.4), we obtain

$$f_j(x^k + d^k + d^k_1) = f_j(x^k + d^k) + g_j(x^k + d^k)^T d^k_1 + O(\|d^k_1\|^2)$$

$$= f_j(x^k + d^k) + g_j(x^k)^T d^k_1 + O(\|d^k\| \cdot \|d^k_1\|) + O(\|d^k\|^2)$$

$$= -\|d^k_0\|^2 + f_j(x^k) + g_j(x^k)^T d^k + O(\|d^k_0\|^3)$$

$$= -\|d^k_0\|^2 + f_j(x^k) + g_j(x^k)^T d^k - \delta(x^k) + O(\|d^k_0\|^3)$$

$$\leq -\|d^k_0\|^2 + O(\|d^k_0\|^3) \leq 0, \ j \in I_k.$$

(4.5)

Hence the second inequalities of (2.30) hold for $\tau = 1$ and $k$ large enough.

Secondly, we show that the first inequality of (2.30) holds for $\tau = 1$ and $k$ large enough.

Expanding $F_c$ around $x^k$ and taking into account relationships (4.2), we get

$$b_k \overset{\text{def}}{=} F_c(x^k + d^k + d^k_1) - F_c(x^k) - \alpha \nabla F_c(x^k)^T d^k$$

$$= \nabla F_c(x^k)^T (d^k + d^k_1) + \frac{1}{2} (d^k)^T \nabla^2 F_c(x^k) d^k - \alpha \nabla F_c(x^k)^T d^k + o(\|d^k\|^2).$$

(4.6)

From the KKT condition of (2.18) and the active set $J_k$ defined by (4.3), we obtain

$$\nabla F_c(x^k) = -H_k d^k_0 - \sum_{j \in J_k} \lambda_j^k g_j(x^k) = -H_k d^k - \sum_{j \in J_k} \lambda_j^k g_j(x^k) + o(\|d^k\|^2).$$

(4.7)

This, together with (4.2), gives

$$\nabla F_c(x^k)^T d^k = -(d^k)^T H_k d^k - \sum_{j \in J_k} \lambda_j^k g_j(x^k)^T d^k + o(\|d^k\|^2)$$

$$= -(d^k)^T H_k d^k - \sum_{j \in J_k} \lambda_j^k g_j(x^k)^T d^k_0 + o(\|d^k\|^2),$$

(4.8)

$$\nabla F_c(x^k)^T (d^k + d^k_1) = -(d^k)^T H_k d^k - \sum_{j \in J_k} \lambda_j^k g_j(x^k)^T (d^k + d^k_1) + o(\|d^k\|^2).$$

(4.9)

On the other hand, by (4.5) and Taylor expansion, we have

$$o(\|d^k\|^2) = f_j(x^k + d^k + d^k_1) = f_j(x^k) + g_j(x^k)^T (d^k + d^k_1) + \frac{1}{2} (d^k)^T \nabla^2 f_j(x^k) d^k$$

$$+ o(\|d^k\|^2), \ j \in J_k.$$

Thus

$$- \sum_{j \in J_k} \lambda_j^k g_j(x^k)^T (d^k + d^k_1) = \sum_{j \in J_k} \lambda_j^k f_j(x^k) + \frac{1}{2} (d^k)^T \left( \sum_{j \in J_k} \lambda_j^k \nabla^2 f_j(x^k) \right) d^k$$

$$+ o(\|d^k\|^2).$$

(4.10)
So (4.9) and (4.10) give
\[
\nabla F_c(x^k)^T (d^k + d_1^k) = - (d^k)^T H_k d^k + \sum_{j \in J_k} \lambda^k_j f_j(x^k) + \frac{1}{2} (d^k)^T \left( \sum_{j \in J_k} \lambda^k_j \nabla^2 f_j(x^k) \right) + o(\|d^k\|^2),
\]
(4.11)
Substituting (4.11) and (4.8) into (4.6), we obtain
\[
b_k = \left( \alpha - \frac{1}{2} \right) (d^k)^T H_k d^k + (1 - \alpha) \sum_{j \in J_k} \lambda^k_j f_j(x^k)
\]
\[
+ \frac{1}{2} (d^k)^T \left( \nabla^2 F_c(x^k) + \sum_{j \in J_k} \lambda^k_j \nabla^2 f_j(x^k) - H_k \right) d^k + o(\|d^k\|^2)
\]
\[
= \left( \alpha - \frac{1}{2} \right) (d^k)^T H_k d^k + (1 - \alpha) \sum_{j \in J_k} \lambda^k_j f_j(x^k)
\]
\[
+ \frac{1}{2} (d^k)^T (\nabla^2 \lambda^k \lambda^k_{x^k} - H_k) d^k + o(\|d^k\|^2).
\]
This, together with (3.1), assumption (H5) and \( \lambda^k_j f_j(x^k) \leq 0 \), shows that
\[
b_k \leq (\alpha - \frac{1}{2}) a \|d^k\|^2 + o(\|d^k\|^2) \leq 0.
\]
So the first inequality of (2.30) holds for \( \tau = 1 \) and \( k \) large enough. Hence the whole proof is complete. \( \square \)

Based on Theorem 4.2, we now state and prove the main result of this section.

**Theorem 4.3.** Under the assumptions (H1)–(H5), Algorithm A is superlinearly convergent, i.e., the sequence \( \{x^k\} \) generated by the algorithm satisfies \( \|x^{k+1} - x^*\| = o(\|x^k - x^*\|) \).

**Proof.** Relations (4.2) and Theorem 4.2 give that the sequence \( \{x^k\} \) generated by Algorithm A has the form of
\[
x^{k+1} = x^k + d^k + d_1^k = x^k + d_0^k + (d^k - d_0^k + d_1^k) = x^k + d_0^k + \bar{d}^k,
\]
for \( k \) large enough, and \( \bar{d}^k \) such that \( \bar{d}^k = O(\|d_0^k\|^2) \). If we let sets \( T_k \equiv L_2, L^-_k \equiv I_k, L^0_k \equiv O, \phi^k_j \equiv 0, \forall j \in L \), then the proposed Algorithm A will reduce to a special case of the Algorithm Model 1.1 of [15], so the conclusion follows immediately from Theorem 2.3 in [15] which established a general result of the convergent rate of Algorithm Model 1.1 of [15]. \( \square \)
5. Numerical experiments

In this section, we test some practical problems based on the proposed algorithm. The numerical experiments are implemented on MATLAB 6.5, and we utilize its optimization toolbox to solve the quadratic program (2.18). The preliminary numerical results show that the proposed algorithm is efficient.

Due to the fact that the updating procedure for matrix \( H_k \) is very important in SQP methods, it determines the superlinearly convergent property of the proposed algorithm. The most widely choice is to use the BFGS formula with Powell’s modification given in Ref. [30], however, such modification is only proved to be two-step superlinear convergence. Pantoja and Mayne in [5] proposed another version of the BFGS formula, which is superior to that described in [30], so we use it in the tests with slight modification to suit our algorithm as follows:

\[
H_{k+1} = H_k - \frac{H_k s_k (s_k)^T H_k}{(s_k)^T H_k s_k} + \frac{\gamma_k (y_k)^T (s_k)^T y_k}{(s_k)^T y_k}, \quad (k \geq 0)
\]

where

\[
s^k = x^{k+1} - x^k, \quad y^k = y^k + a_k (\gamma_k s^k + A_k A_k^T s^k), \quad \gamma_k = \min \{ \| d_0^k \|^2, \xi \}, \quad \xi \in (0, 1)
\]

\[
y^k = \nabla_x L_{c_k} (x^{k+1}, \lambda^k) - \nabla_x L_{c_k} (x^k, \lambda^k), \quad A_k = (\nabla f_j (x^k), j \in J_k),
\]

\[
\nabla_x L_{c_k} (x, \lambda) = \nabla F_{c_k} (x) + \sum_{j \in I_k} \lambda_j \nabla f_j (x), \quad J_k = \{ j \in I_k : f_j (x^k) + g_j (x^k)^T d_0^k = 0 \}.
\]

\[
a_k = \begin{cases} 
0, & \text{if } (s^k)^T y^k \geq \delta \| s^k \|^2, \delta \in (0, 1), \\
1, & \text{if } 0 \leq (s^k)^T y^k < \delta \| s^k \|^2, \\
1 + \frac{\gamma_k \| s^k \|^2 - (s^k)^T y^k}{\gamma_k \| s^k \|^2 + (s^k)^T A_k (A_k)^T s_k}, & \text{otherwise}.
\end{cases}
\]

During the numerical experiments, we set

\[
e_{-1} = 2, \quad \kappa = 2.5, \quad \alpha = 0.1, \quad \beta = 0.5, \quad \ell = 0.1, \quad r = 0.1, \quad c_{-1} = 1.
\]

In formula (5.1), we select \( \gamma_k = 0.5 \), \( \delta = 0.5 \) and \( H_0 = E \), where \( E \in R^{n \times n} \) is an identity matrix. Execution is terminated if the norm of \( d_0^k \) is less than a given constant \( e > 0 \).

The test problems in Table 1 are selected from Refs. [11] and [33], particularly, HS032, HS063, HS081 etc. is problem 32, 63, 81 etc. in [11], respectively, and S217, S225, S263 etc. is problem 217, 225, 263 etc. in [33], respectively. The columns of Table 1 have the following meanings:

- \( |L_1| \), \( |L_2| \) give the number of inequality and equality constraints, respectively;
- \( Ni \), \( Nf_0 \), \( Nf \) give the number of iterations, objective function evaluations and all constraint functions evaluations, respectively;
- \( Eps \) denotes the stopping criterion threshold \( \varepsilon \).

From Table 1, we can see that our Algorithm A executes well for these problems taken from [11] and [33], and the approximately optimal solutions are listed in Table 2. We must point out here that we record all the KKT multipliers of \( (QP_{c_k}) \) for each problem, and find out the strict complementarity condition does not satisfy at the approximately optimal solution for problem HS032, since the approximately
multiplier associated with constraint \( -x_1 \leq 0 \) equals zero, while \( -x_1 \leq 0 \) is an active constraint at \( x^* \) (see Table 2), and our algorithm also performs well for this problem. Tables 3 and 4 give the detailed iterations for two typical problems in \( R^2 \), so as to be illustrated intuitively in figures, Figs. 1 and 2, respectively. Moreover, \( r_k = 1 \) is accepted at most steps of Tables 3 and 4, \( c_k \) in Table 4 increases only once.

The following problem is used to show the Maratos effect in [21].

\[
\begin{align*}
\min \quad & f_0(x) = x_1^2 + x_2^2, \\
\text{s.t.} \quad & f_1(x) = -(x_1 + 1)^2 - x_2^2 + 4 \leq 0.
\end{align*}
\]  

(5.2)

Table 3 shows the detailed iterations of problem (5.2).
The following problem is taken from [33], test example 325.

\[
\begin{align*}
\min & \quad f_0(x) = x_1^2 + x_2 \\
\text{s.t.} & \quad f_1(x) = x_1 + x_2 - 1 \leq 0, \\
& \quad f_2(x) = x_1 + x_2^2 - 1 \leq 0, \\
& \quad f_3(x) = x_1^2 + x_2^2 - 9 = 0.
\end{align*}
\]  

(5.3)

Table 4 shows the detailed iterations of problem (5.3).

The intuitional iterations for Tables 3 and 4 are given in Figs. 1 and 2, respectively. The symbol “∗” in these figures denotes the site of the iteration point, and symbol “x0”, “x1” and “x7” etc. denotes the initial point \(x^0\), the first iteration point \(x^1\) and the 7th iteration point \(x^7\) etc., respectively.
Fig. 1. An intuitive figure for Table 3.

Fig. 2. An intuitive figure for Table 4.
6. Concluding remarks

In this paper, we proposed a feasible descent SQP algorithm for general optimization problems with equality and inequality constraints. At first, we transform the original problem to an associated simpler problem with only inequality constraints, then use feasible descent SQP method to solve the simplified problem. At each iteration, only one QP subproblem needs to be solved, consequently, a master direction is obtained. In order to get a feasible descent direction and a height-order correction direction which can use to avoid the Maratos effect and guarantee the superlinear convergence rate under weaker condition without the strict complementarity, we correct the master direction by two simple explicit formulas instead of solving QPs, as a result, the computational cost is reduced comparing with many other existing feasible SQP methods. Numerical results in Section 5 also show that the proposed algorithm is promising.

In fact, if the Hessian matrices $\nabla^2 f_j(x)$ ($j \in L^0$) of the objective function and constraint functions are used to generate the updating matrix $H_k$, it is possible to establish the quadratically convergent property of our Algorithm A. For example, if we use formula (2.32) described in [15], then the quadratical convergence rate achieves immediately from Theorem 2.10 of [15].

References


\[34\] G. Zoutendijk, Methods of Feasible Directions, Elsevier Science, Amsterdam, Netherlands, 1960.