TECHNICAL RESEARCH REPORT

A Computationally Efficient Feasible Sequential Quadratic Programming Algorithm

by C. Lawrence, A. Tits

T.R. 98-46

INSTITUTE FOR SYSTEMS RESEARCH

ISR develops, applies and teaches advanced methodologies of design and analysis to solve complex, hierarchical, heterogeneous and dynamic problems of engineering technology and systems for industry and government.

ISR is a permanent institute of the University of Maryland, within the Glenn L. Martin Institute of Technology/A. James Clark School of Engineering. It is a National Science Foundation Engineering Research Center.

Web site http://www.isr.umd.edu

A COMPUTATIONALLY EFFICIENT FEASIBLE SEQUENTIAL QUADRATIC PROGRAMMING ALGORITHM

Craig T. Lawrence and André L. Tits
Department of Electrical Engineering
and Institute for Systems Research
University of Maryland, College Park
College Park, Maryland 20742

September 9, 1998

Abstract

A Sequential Quadratic Programming (SQP) algorithm generating feasible iterates is described and analyzed. What distinguishes this algorithm from previous feasible SQP algorithms proposed by various authors is a drastic reduction in the amount of computation required to generate a new iterate while the proposed scheme still enjoys the same global and fast local convergence properties. A preliminary implementation has been tested and some promising numerical results are reported.

1 Introduction

Consider the inequality constrained nonlinear programming problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g_j(x) \leq 0, \quad j = 1, \ldots, m, \end{array} \tag{P}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \ldots, m$, are continuously differentiable. When the number of variables n is not too large, Sequential Quadratic Programming (SQP) algorithms are widely acknowledged to be the most successful algorithms available for solving (P). For an excellent recent survey of SQP algorithms, and the theory behind them, see [2].

Denote the feasible set for (P) by

$$X \triangleq \{\ x \in \mathbb{R}^n \mid g_j(x) \leq 0,\ j = 1, \ldots, m\ \}.$$

In [14, 6, 11, 12, 1], variations on the standard SQP iteration for solving (P) are proposed which generate iterates lying within X. Such methods are sometimes referred to as "Feasible SQP" (or FSQP) algorithms. It was observed that requiring feasible iterates has both algorithmic and application-oriented advantages. Algorithmically, feasible iterates are desirable because

- The QP subproblems are always consistent, i.e. a feasible solution always exists, and
- The objective function may be used directly as a merit function in the line search.

In an engineering context, feasible iterates are important because

- Often $f(x)$ is undefined outside of the feasible region X,
- The optimization process may be stopped after a few iterations, yielding a feasible point, and
- Trade-offs may be meaningfully explored.

Each of these features is relevant in both engineering analysis and design. Further, the second is critical for real-time applications, where a feasible point may be required before the algorithm has had time to "converge" to a solution.

An important function associated with the problem (P) is the *Lagrangian* $L : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, which is defined by

$$L(x, \lambda) \triangleq f(x) + \sum_{i=1}^{m} \lambda_i g_i(x).$$

Given estimates $x \in X$ of the solution of (P), $0 \leq \lambda \in \mathbb{R}^m$ of the Lagrange multipliers at the solution, and $0 < H = H^T \in \mathbb{R}^{n \times n}$ of the Hessian of the Lagrangian $L(x, \lambda)$, the standard SQP search direction $d^0 = d^0(x, H) \in \mathbb{R}^n$ is computed as a solution of the Quadratic Program (QP)

$$\begin{array}{ll} \min & \frac{1}{2}\langle d^0, Hd^0\rangle + \langle \nabla f(x), d^0\rangle \\ \text{s.t.} & g_j(x) + \langle \nabla g_j(x), d^0\rangle \leq 0, \quad j = 1, \ldots, m. \end{array} \qquad QP^0(x, H)$$

With an appropriate merit function, line search procedure, Hessian approximation procedure, and (if necessary) Maratos avoidance scheme, the SQP iteration is well-known to be globally and locally superlinearly convergent (see, e.g., [2]).

A *feasible direction* at a point $x \in X$ is defined as any vector $d \in \mathbb{R}^n$ which satisfies $x + td \in X$ for all $t \in [0, \bar{t}\,]$, for some $\bar{t} > 0$. Note that the SQP direction $d^0 = d^0(x, H)$, a direction of descent for f, may not be a feasible direction at x, though it is at worst tangent to the active constraint surface. Thus, in order to generate feasible iterates in the SQP framework, it is necessary to "tilt" d^0 into the feasible set. A number of different approaches have been considered in the literature for generating feasible directions and, specifically, tilting the SQP direction.

Early feasible direction algorithms (see, e.g., [24, 14]) were first-order methods, i.e. only first derivatives were used and no attempt was made to accumulate and use second-order information. Furthermore, search directions were often computed via linear programs instead of QPs. As a consequence, such algorithms converged linearly at best. Polak proposed several extensions to these algorithms (see [14], Section 4.4) which took second-order information into account when computing the search direction. A few of the search directions proposed by Polak could be viewed as tilted SQP directions (with proper choice of the matrices encapsulating the second-order information in the defining equations). Even with the second-order information, though, it was not possible to guarantee superlinear convergence because no mechanism was included for controlling the *amount* of tilting.

A straightforward way to tilt the SQP direction is, of course, to perturb the right-hand side of the constraints in $QP^0(x, H)$ directly. Building on this observation, Herskovits and Carvalho [6] and Panier and Tits [11] independently developed similar feasible SQP algorithms in which the size of the perturbation was a function of the norm of $d^0(x, H)$ at the current point $x \in X$. Thus, their algorithms required the solution of $QP^0(x, H)$ in order to *define* the perturbed QP. Both algorithms were shown to be superlinearly convergent. On the other hand, as a by-product of the tilting scheme, global convergence proved to be more elusive. In fact, the algorithm in [6] is not globally convergent, while the algorithm in [11] had to resort to a first-order search direction far from a solution in order to guarantee global convergence. Such a hybrid scheme could give slow convergence if a poor initial point is chosen.

The algorithm developed by Panier and Tits in [12], and analyzed under weaker assumptions by Qi and Wei in [17], has enjoyed a great deal of success in practice as implemented in the FFSQP/CFSQP [23, 10] software

packages. We will refer to their algorithm throughout this paper as **FSQP**. In [12], instead of directly perturbing $QP^0(x,H)$, tilting is accomplished by replacing d^0 with the convex combination $d = (1-\rho)d^0 + \rho d^1$, where $d^1 \in \mathbb{R}^n$ is an (essentially) arbitrary feasible descent direction. To preserve the local convergence properties of the SQP iteration, $\rho = \rho(d^0) \in [0,1]$ is computed so that d approaches d^0 fast enough (in particular, $\rho(d^0) = O(\|d^0\|^2)$) as the solution is approached. Finally, in order to avoid the Maratos effect and guarantee a superlinear rate of convergence, a second order correction $\tilde{d} = \tilde{d}(x,d,H) \in \mathbb{R}^n$ is used to "bend" the search direction. That is, an Armijo-type search is performed along the arc $x + td + t^2\tilde{d}$. In [12], the directions d^1 and $\tilde{d}$ are both computed via QPs, which we will refer to as, respectively, $QP^1(x)$ and $\widetilde{QP}(x,d,H)$. It is observed in [12] that $\tilde{d}$ could instead be taken as the solution of a linear least squares problem without affecting the asymptotic convergence properties.

From the point of view of computational cost, the main drawback of algorithm **FSQP** is the need to solve three QPs (or two QPs and a linear least squares problem) at each iteration. Clearly, for many problems it would be desirable to reduce the number of QPs at each iteration while preserving the generation of feasible iterates as well as the global and local convergence properties. This is especially critical in the context of those large-scale nonlinear programs for which the time spent solving the QPs dominates that used to evaluate the functions.

In this paper, we consider a perturbation of $QP^0(x,H)$ which allows more control over the tilting. Specifically, given $x \in X$, $0 < H = H^T \in \mathbb{R}^{n\times n}$, and $0 \le \eta \in \mathbb{R}$, let $(\hat{d},\hat{\gamma}) = (\hat{d}(x,H,\eta),\hat{\gamma}(x,H,\eta)) \in \mathbb{R}^n \times \mathbb{R}$ solve the QP

$$\begin{aligned} \min \quad & \tfrac{1}{2}\langle \hat{d}, H\hat{d}\rangle + \hat{\gamma} \\ \text{s.t.} \quad & \langle \nabla f(x), \hat{d}\rangle \le \hat{\gamma}, \\ & g_j(x) + \langle \nabla g_j(x), \hat{d}\rangle \le \hat{\gamma}\cdot\eta, \quad j = 1,\ldots,m. \end{aligned} \qquad \widehat{QP}(x,H,\eta)$$

In Section 3, we show that $\hat{d}$ is a descent direction and, for $\eta > 0$, $\hat{d}$ is a feasible direction. Note that for $\eta \equiv 1$, the search direction is a special case of those computed in Polak's second-order feasible direction algorithms (again, see Section 4.4 in the book [14]). Further, it is not difficult to show that when $\eta \equiv 0$, we recover the SQP direction, i.e. $\hat{d}(x,H,0) = d^0(x,H)$. Large values of the parameter η, which we will call the *tilting parameter*, emphasize feasibility, while small values of η emphasize descent.

In [1], Birge, Qi, and Wei propose a feasible SQP algorithm based on $\widehat{QP}(x,H,\eta)$. Their motivation for introducing the right-hand-side constraint

perturbation and the tilting parameters (they use a vector of parameters, one for each constraint) is, like us, to obtain a feasible search direction. Specifically, motivated by the nature of the application problems they are interested in tackling, their goal is to ensure a full step of one is accepted in the line search as early as is possible (so that costly line searches are avoided for most iterations). To this end, their tilting parameters start out positive and, if anything, increase when a step of one is not accepted. A side-effect of such an updating scheme is that the algorithm cannot achieve a superlinear rate of convergence, as the authors point out in Remark 5.1 of [1].

In the present paper, our goal is to compute a feasible descent direction which approaches the true SQP direction fast enough so as to ensure superlinear convergence. Furthermore, we would like to do this with as little computation per iteration as possible. While computationally the most expensive, algorithm **FSQP** of [12] has the convergence properties and practical performance we seek. Motivated by this observation, we now examine the relevant properties of the search directions generated by algorithm **FSQP** on the sequence of iterates $\{x_k\}$. For $x \in X$, define

$$I(x) \triangleq \{\ j \mid g_j(x) = 0\ \},$$

the index set of active constraints at the point x. In [12], in order for the line-search (with the objective function $f(x)$ used directly as the merit function) to be well-defined, and in order to preserve global and fast local convergence, the sequence of search directions $\{d_k\}$ generated by algorithm **FSQP** is constructed so that the following properties hold:

1. $d_k = 0$ if x_k is a KKT point for (P),
2. $\langle \nabla f(x_k), d_k \rangle < 0$ if x_k is not a KKT point,
3. $\langle \nabla g_j(x_k), d_k \rangle < 0$, for all $j \in I(x_k)$ if x_k is not a KKT point, and
4. $d_k = d_k^0 + O(\|d_k^0\|^2)$.

We will show in Section 3 that for $H_k = H_k^T > 0$ and $\eta_k \geq 0$, $\hat{d}_k = \hat{d}(x_k, H_k, \eta_k)$ automatically satisfies the first two properties. Furthermore, $\hat{d}_k$ satisfies the third property if $\eta_k > 0$. Ensuring the fourth property is satisfied requires a bit more care.

In the algorithm presented in Section 2, at iteration k, we compute the search direction via $\widehat{QP}(x_k, H_k, \eta_k)$ and the tilting parameter η_k is iteratively adjusted to ensure the four properties are satisfied. The resultant algorithm will be shown to be locally superlinearly convergent and globally convergent

without resorting to a first-order direction far from the solution. Further, the generation of a new iterate will only require the solution of one QP and two closely related linear least squares problems. Note that, in contrast with the algorithm presented in [1], our tilting parameter starts out positive and asymptotically approaches zero.

Recently there has been a great deal of interest in interior point algorithms for nonconvex nonlinear programming (see, e.g., [4, 5, 21, 3, 13, 20]). Such algorithms generate feasible iterates and typically only require the solution of linear systems of equations in order to generate new iterates. Performance of interior point algorithms tends to be closely related to the careful iterative reduction of a barrier parameter. Essentially, search directions are computed based upon quadratic models of logarithmic barrier functions. On the other hand, SQP-type methods, such as the algorithm proposed here, base search directions upon a quadratic model of the original problem. Thus SQP-type methods should, in general, generate better search directions than interior point methods at the expense of possibly more work per iteration. Of course, work is still very much in its infancy for interior point nonconvex nonlinear programming algorithms. Eventually, such algorithms may be an attractive alternative, especially for very large problems.

In Section 2, we present the details of our new FSQP algorithm. In Section 3, we show that under mild assumptions our iteration is globally convergent, as well as locally superlinearly convergent. The algorithm has been implemented and tested and we show in Section 4 that the numerical results are quite promising. Finally, in Section 5, we offer some concluding remarks and discuss some extensions to the algorithm which are currently being explored.

2 Algorithm

We begin by making a few assumptions that will be in force throughout.

Assumption 1: The set X is non-empty.

Assumption 2: The functions $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \ldots, m$, are continuously differentiable.

Assumption 3: For all $x \in X$ with $I(x) \neq \emptyset$, the set $\{\nabla g_j(x) \mid j \in I(x)\}$ is linearly independent.

A point $x \in \mathbb{R}^n$ is said to be a *Karush-Kuhn-Tucker (KKT)* point for

the problem (P) if there exist scalars (*KKT multipliers*) λ^j, $j = 1, \dots, m$, satisfying

$$\begin{cases} \nabla f(x) + \sum_{j=1}^{m} \lambda^j \nabla g_j(x) = 0, \\ g_j(x) \leq 0, \quad j = 1, \dots, m, \\ \lambda^j g_j(x) = 0 \text{ and } \lambda^j \geq 0, \quad j = 1, \dots, m. \end{cases} \tag{1}$$

It is well known that, under our assumptions, a necessary condition for optimality for a point $x \in X$ is that it be a KKT point, i.e. satisfy the KKT conditions.

Note that, with $x \in X$, $\widehat{QP}(x, H, \eta)$ is always consistent: $(0, 0)$ satisfies the constraints. Indeed, $\widehat{QP}(x, H, \eta)$ always has a unique solution $(\hat{d}, \hat{\gamma})$ (see Lemma 1 below) which, by convexity, is its unique KKT point; i.e. there exists multipliers $\hat{\mu} \in \mathbb{R}$ and $\hat{\lambda}^j$, $j = 1, \dots, m$, which, together with $(\hat{d}, \hat{\gamma})$, satisfy

$$\begin{cases} \begin{bmatrix} H\hat{d} \\ 1 \end{bmatrix} + \hat{\mu} \begin{bmatrix} \nabla f(x) \\ -1 \end{bmatrix} + \sum_{j=1}^{m} \hat{\lambda}^j \begin{bmatrix} \nabla g_j(x) \\ -\eta \end{bmatrix} = 0, \\ \langle \nabla f(x), \hat{d} \rangle \leq \hat{\gamma}, \\ g_j(x) + \langle \nabla g_j(x), \hat{d} \rangle \leq \hat{\gamma} \cdot \eta, \quad \forall j = 1, \dots, m, \\ \hat{\mu} \left(\langle \nabla f(x), \hat{d} \rangle - \hat{\gamma} \right) = 0 \text{ and } \hat{\mu} \geq 0, \\ \hat{\lambda}^j \left(g_j(x) + \langle \nabla g_j(x), \hat{d} \rangle - \hat{\gamma} \cdot \eta \right) = 0 \text{ and } \hat{\lambda}^j \geq 0, \quad \forall j = 1, \dots, m. \end{cases} \tag{2}$$

A simple consequence of the first equation in (2), which will be used throughout our analysis, is an affine relationship amongst the multipliers, namely

$$\hat{\mu} + \eta \cdot \sum_{j=1}^{m} \hat{\lambda}^j = 1. \tag{3}$$

The parameter η will be iteratively adjusted, i.e. $\eta = \eta_k$, to ensure that $\hat{d}_k = \hat{d}(x_k, H_k, \eta_k)$ has the necessary properties. At iteration k, choosing $\eta_k > 0$ is sufficient to guarantee the first three properties are satisfied. As it turns out, though, we will need something a little stronger than this. In order to ensure that, away from a solution, there is adequate tilting into the feasible set (hence the step size does not collapse) we strengthen the positivity requirement to force η_k to be bounded away from zero away from

KKT points of (P). Finally, the fourth property requires that $\eta_k \to 0$, as $k \to \infty$, sufficiently fast as $d^0(x_k, H_k) \to 0$. Of course, we do not want to compute $d_k^0 = d^0(x_k, H_k)$, as is done in [11], so we must rely on some other information to update η_k.

Given an estimate I_k of the active set $I(x_k)$, we can compute an estimate $\widehat{d_k^0} = \widehat{d^0}(x_k, H_k, I_k)$ of $d^0(x_k, H_k)$ by solving the *equality* constrained QP

$$\begin{array}{ll} \min & \frac{1}{2}\langle \widehat{d^0}, H_k \widehat{d^0} \rangle + \langle \nabla f(x_k), \widehat{d^0} \rangle \\ \text{s.t.} & g_j(x_k) + \langle \nabla g_j(x_k), \widehat{d^0} \rangle = 0, \quad j \in I_k, \end{array} \qquad LS^0(x_k, H_k, I_k)$$

which is equivalent (after a change of variables) to solving a linear least squares problem.[1] Let $\hat{I}_k$ be the set of active constraints, not including the "objective descent" constraint $\langle \nabla f(x_k), \hat{d}_k \rangle \leq \hat{\gamma}_k$, for $\widehat{QP}(x_k, H_k, \eta_k)$, i.e.

$$\hat{I}_k \stackrel{\Delta}{=} \{ \ j \mid g_j(x_k) + \langle \nabla g_j(x_k), \hat{d}_k \rangle = \hat{\gamma}_k \cdot \eta_k \ \}.$$

We will show in Section 3 that $\widehat{d^0}(x_k, H_k, \hat{I}_{k-1}) = d^0(x_k, H_k)$ for all k sufficiently large. Furthermore, it will be shown that, when $\hat{d}_k$ is small, choosing

$$\eta_k \propto \|\widehat{d^0}(x_k, H_k, \hat{I}_{k-1})\|^2$$

will be sufficient to establish global and 2-step superlinear convergence. Proper choice of the proportionality constant (C_k in the algorithm statement below), while not important in the convergence analysis, is critical for satisfactory numerical performance. This will be discussed in Section 4.

In [12], the Maratos correction $\tilde{d}_k$ is taken as the solution of the QP

$$\begin{array}{ll} \min & \frac{1}{2}\langle \hat{d}_k + \tilde{d}, H_k(\hat{d}_k + \tilde{d}) \rangle + \langle \nabla f(x_k), \hat{d}_k + \tilde{d} \rangle \\ \text{s.t.} & g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k), \hat{d}_k + \tilde{d} \rangle \leq -\|\hat{d}_k\|^\tau, \quad j = 1, \ldots, m, \end{array} \qquad \widetilde{QP}(x_k, \hat{d}_k, H_k)$$

if it exists and has norm less than $\min\{\|\hat{d}_k\|, C\}$, where $\tau \in (2, 3)$ and C large are given. Otherwise, $\tilde{d}_k = 0$. In Section 1, it was mentioned that a linear least squares problem could be used instead of a QP to compute a version of the Maratos correction $\tilde{d}$ with the same asymptotic convergence properties. Given that our goal is to reduce the computational cost per iteration, it makes sense to use such an approach here. Thus, we take $\tilde{d}_k = \tilde{d}(x_k, \hat{d}_k, H_k, \hat{I}_k)$ as the solution, if it exists and is not too large, of

[1] Which is, in turn, equivalent to solving a square system of linear equations in $n + |\hat{I}_k^0|$ variables.

the equality constrained QP (equivalent to a least squares problem after a change of variables)

$$\begin{array}{ll} \min & \langle \hat{d}_k + \tilde{d}, H_k(\hat{d}_k + \tilde{d})\rangle + \langle \nabla f(x_k), \hat{d}_k + \tilde{d}\rangle \\ \text{s.t.} & g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k), \tilde{d}\rangle = -\|\hat{d}_k\|^\tau, \quad \forall j \in \hat{I}_k, \\ & \qquad\qquad\qquad\qquad\qquad\qquad \widetilde{LS}(x_k, \hat{d}_k, H_k, \hat{I}_k) \end{array}$$

where $\tau \in (2,3)$, a direct extension of an alternative considered in [11]. Such an objective, as compared to the pure least squares objective $\|\tilde{d}\|^2$, should improve numerical performance without significantly increasing computational requirements (or affecting the convergence analysis). In the case that $\widetilde{LS}(x_k, \hat{d}_k, H_k, \hat{I}_k)$ is inconsistent, or the computed solution $\tilde{d}_k$ is too large, we could simply set $\tilde{d}_k = 0$. Note that one should use $\widetilde{QP}(x_k, \hat{d}_k, H_k)$ for problems in which function evaluations are expensive compared to the solution of a QP since it provides a better model of (P).

The proposed algorithm is as follows.

Algorithm FSQP$'$

Parameters: $\alpha \in (0, \frac{1}{2})$, $\beta \in (0,1)$, $\tau \in (2,3)$, $\epsilon_\ell > 0$, $0 < \underline{C} \leq \overline{C}$, $\bar{D} > 0$.

Data: $x_0 \in X$, $0 < H_0 = H_0^T \in \mathbb{R}^{n\times n}$, $0 < \eta_0 \in \mathbb{R}$.

Step 0 - Initialization. **set** $k \leftarrow 0$.

Step 1 - Computation of search arc.

(i). **compute** $(\hat{d}_k, \hat{\gamma}_k) = (\hat{d}(x_k, H_k, \eta_k), \hat{\gamma}(x_k, H_k, \eta_k))$, the active set $\hat{I}_k$, and the associated multipliers $\hat{\mu}_k \in \mathbb{R}$, $\hat{\lambda}_k \in \mathbb{R}^m$.
if $(\hat{d}_k = 0)$ **then stop**.

(ii). **compute** $\tilde{d}_k = \tilde{d}(x_k, \hat{d}_k, H_k, \hat{I}_k)$ if it exists and satisfies $\|\tilde{d}_k\| \leq \|\hat{d}_k\|$. Otherwise, **set** $\tilde{d}_k = 0$.

Step 2 - Arc search. **compute** t_k, the first number t in the sequence $\{1, \beta, \beta^2, \dots\}$ satisfying

$$f(x_k + t\hat{d}_k + t^2\tilde{d}_k) \leq f(x_k) + \alpha t \langle \nabla f(x_k), \hat{d}_k\rangle,$$

$$g_j(x_k + t\hat{d}_k + t^2\tilde{d}_k) \leq 0, \quad j = 1, \dots, m.$$

Step 3 - Updates.

(i). **set** $x_{k+1} \leftarrow x_k + t_k\hat{d}_k + t_k^2\tilde{d}_k$.

(ii). **compute** a new symmetric positive definite estimate H_{k+1} to the Hessian of the Lagrangian.

(iii). **select** $C_{k+1} \in [\underline{C}, \overline{C}]$.

- $*$ **if** $(\|\hat{d}_k\| < \epsilon_\ell)$ **then**
  - $\cdot$ **compute**, if possible,[2] $\widehat{d^0_{k+1}} = \widehat{d^0}(x_{k+1}, H_{k+1}, \hat{I}_k)$, and the associated multipliers $\widehat{\lambda^0_{k+1}} \in \mathbb{R}^{|\hat{I}_k|}$.
  - $\cdot$ **if** $\left(\widehat{d^0_{k+1}}\right.$ exists and $\|\widehat{d^0_{k+1}}\| \leq \bar{D}$ and $\left.\widehat{\lambda^0_{k+1}} \geq 0\right)$ **then set**

$$\eta_{k+1} \leftarrow C_{k+1} \cdot \|\widehat{d^0_{k+1}}\|^2.$$

  - $\cdot$ **else set** $\eta_{k+1} \leftarrow C_{k+1} \cdot \|\hat{d}_k\|^2$.
- $*$ **else set** $\eta_{k+1} \leftarrow C_{k+1} \cdot \epsilon_\ell^2$.

(iv). **set** $k \leftarrow k+1$ and **goto** *Step 1.*

3 Convergence Analysis

Much of our analysis, especially the local analysis, will be devoted to establishing the relationship between $\hat{d}(x, H, \eta)$ and the SQP direction $d^0(x, H)$. As a consequence, we will be referring to the KKT conditions for $QP^0(x, H)$ in several places. The direction $d^0 = d^0(x, H)$ is a KKT point for $QP^0(x, H)$ if there exists a multiplier $\lambda^0 \in \mathbb{R}^m$ satisfying

$$\begin{cases} Hd^0 + \nabla f(x) + \displaystyle\sum_{j=1}^{m} \lambda^{0,j} \nabla g_j(x) = 0, \\ g_j(x) + \langle \nabla g_j(x), d^0 \rangle \leq 0, \quad j = 1, \ldots, m, \\ \lambda^{0,j} \cdot \big(g_j(x) + \langle \nabla g_j(x), d^0 \rangle\big) = 0 \text{ and } \lambda^{0,j} \geq 0, \quad j = 1, \ldots, m. \end{cases} \tag{4}$$

Further, an estimate $\widehat{d^0} = \widehat{d^0}(x, H, I)$ is a KKT point for $LS^0(x, H, I)$ if there exists a multiplier $\widehat{\lambda^0} \in \mathbb{R}^m$ satisfying

$$\begin{cases} H\widehat{d^0} + \nabla f(x) + \displaystyle\sum_{j \in I} \widehat{\lambda^0}^j \nabla g_j(x) = 0, \\ g_j(x) + \langle \nabla g_j(x), \widehat{d^0} \rangle = 0, \quad j \in I. \end{cases} \tag{5}$$

Note that the components of $\widehat{\lambda^0}$ for $j \notin I$ play no role in the optimality conditions. We chose to always use $\widehat{\lambda^0} \in \mathbb{R}^m$, independent of the size of I, for notational convenience and consistency in indexing.

[2] That is, if $LS^0(x_{k+1}, H_{k+1}, \hat{I}_k)$ is non-degenerate.

3.1 Global Convergence

In this section we establish that, under mild assumptions, our proposed algorithm **FSQP′** generates a sequence of iterates $\{x_k\}$ with the property that all accumulation points are KKT points for the problem (P). We begin by establishing some properties of the tilted SQP search direction $\hat{d}(x, H, \eta)$.

Lemma 1. *Given $H = H^T > 0$, $x \in X$, and $\eta \geq 0$, $\hat{d}(x, H, \eta)$ is well-defined and $(\hat{d}, \hat{\gamma}) = (\hat{d}(x, H, \eta), \hat{\gamma}(x, H, \eta))$ is the unique KKT point of $\widehat{QP}(x, H, \eta)$. Furthermore, for $H = H^T > 0$ and $\eta \geq 0$ fixed, $\hat{d}(x, H, \eta)$ is bounded over bounded subsets of X.*

Proof. First note that the feasible set for $\widehat{QP}(x, H, \eta)$ is non-empty, since $(\hat{d}, \hat{\gamma}) = (0, 0)$ is always feasible. Now consider the cases $\eta = 0$ and $\eta > 0$ separately. From (2) and (4), it is clear that, if $\eta = 0$, then $(\hat{d}, \hat{\gamma})$ is a solution to $\widehat{QP}(x, H, 0)$ if, and only if, $\hat{d}$ is a solution of $QP^0(x, H)$ and $\hat{\gamma} = \langle \nabla f(x), \hat{d} \rangle$. It is well known that, under our assumptions, $d^0(x, H)$ is well-defined, unique, and continuous as a function of x. Thus, the Lemma follows immediately for this case. Suppose now that $\eta > 0$. In this case, $(\hat{d}, \hat{\gamma})$ is a solution of $\widehat{QP}(x, H, \eta)$ if, and only if, $\hat{d}$ solves the unconstrained problem

$$\min \frac{1}{2}\langle \hat{d}, H\hat{d} \rangle + \max \left\{ \langle \nabla f(x), \hat{d} \rangle,\ \frac{1}{\eta} \cdot \max_{j=1,\dots,m} \{ g_j(x) + \langle \nabla g_j(x), \hat{d} \rangle \} \right\}. \quad (6)$$

and

$$\hat{\gamma} = \max \left\{ \langle \nabla f(x), \hat{d} \rangle,\ \frac{1}{\eta} \cdot \max_{j=1,\dots,m} \{ g_j(x) + \langle \nabla g_j(x), \hat{d} \rangle \} \right\}.$$

Since the function being minimized in (6) is strictly convex and radially unbounded, it follows that $(\hat{d}(x, H, \eta), \hat{\gamma}(x, H, \eta))$ is well-defined and unique as a global minimizer for the convex problem $\widehat{QP}(x, H, \eta)$, and thus unique as a KKT point for that problem. Boundedness of $\hat{d}(x, H, \eta)$ over bounded subsets of X follows from the first equation in (2), $H > 0$, our regularity assumptions, and (3), which shows (since $\eta > 0$) that the multipliers are bounded. □

Lemma 2. *Given $H = H^T > 0$ and $\eta \geq 0$*

(i). $\hat{\gamma}(x, H, \eta) \leq 0$ for all $x \in X$. Moreover, $\hat{\gamma}(x, H, \eta) = 0$ if, and only if, $\hat{d}(x, H, \eta) = 0$.

(ii). $\hat{d}(x, H, \eta) = 0$ if, and only if, x is a KKT point for (P).

Proof. To prove (i), note that since $(\hat{d}, \hat{\gamma}) = (0,0)$ is always feasible for $\widehat{QP}(x, H, \eta)$, the optimal value of the QP is non-positive. Further, since $H > 0$, the quadratic term in the objective is non-negative, which implies $\hat{\gamma}(x, H, \eta) \leq 0$. Now suppose $\hat{d}(x, H, \eta) = 0$, then feasibility of the first QP constraint implies $\hat{\gamma}(x, H, \eta) = 0$. Finally, suppose $\hat{\gamma}(x, H, \eta) = 0$. Since $x \in X$, $H > 0$, and $\eta \geq 0$, it is clear that $\hat{d} = 0$ is both feasible and achieves the minimum value of the objective. Thus, uniqueness gives $\hat{d}(x, H, \eta) = 0$ and part (i) is proved.

Suppose now that $\hat{d}(x, H, \eta) = 0$. Then $\hat{\gamma}(x, H, \eta) = 0$ and by (2) there exists multipliers $\hat{\lambda} \in \mathbb{R}^m$ and $0 \leq \hat{\mu} \in \mathbb{R}$ satisfying

$$\begin{cases} \hat{\mu} \nabla f(x) + \sum_{j=1}^{m} \hat{\lambda}^j \nabla g_j(x) = 0, \\ g_j(x) \leq 0, \quad \forall j = 1, \ldots, m, \\ \hat{\lambda}^j g_j(x) = 0 \text{ and } \hat{\lambda}^j \geq 0, \quad \forall j = 1, \ldots, m. \end{cases}$$

We begin by showing that $\hat{\mu} > 0$. Proceeding by contradiction, suppose $\hat{\mu} = 0$, then by (3) we have

$$\sum_{j=1}^{m} \hat{\lambda}^j > 0.$$

Note that,

$$\begin{aligned} \hat{I} &\triangleq \{ \, j \mid g_j(x) + \langle \nabla g_j(x), \hat{d}(x, H, \eta) \rangle = \hat{\gamma}(x, H, \eta) \cdot \eta \, \} \\ &= \{ \, j \mid g_j(x) = 0 \, \} = I(x). \end{aligned}$$

Thus, by the complementary slackness condition of (2) and the optimality conditions above,

$$0 = \sum_{j=1}^{m} \hat{\lambda}^j \nabla g_j(x) = \sum_{j \in I(x)} \hat{\lambda}^j \nabla g_j(x).$$

By Assumption 3, if $I(x) \neq \emptyset$, then this sum vanishes only if $\hat{\lambda}^j = 0$, for all $j \in I(x)$, but we saw above that this is not the case. Hence we have a contradiction and it follows that $\hat{\mu} > 0$. It is now immediate that x is a KKT point for (P) with multipliers $\lambda^j = \hat{\lambda}^j / \hat{\mu}$, $j = 1, \ldots, m$.

Finally, to prove the necessity portion of part (ii) note that if x is a KKT point for (P), then (1) shows that $(\hat{d}, \hat{\gamma}) = (0,0)$ is a KKT point for $\widehat{QP}(x, H, \eta)$, with $\hat{\mu} = (1 + \eta \sum_j \lambda_j)^{-1}$ and $\hat{\lambda}_j = \lambda_j (1 + \eta \sum_j \lambda_j)^{-1}$, $j = 1, \ldots, m$. Uniqueness of such points (Lemma 1) gives the result. $\square$

The next two lemmas establish that the line search in *Step 2* of Algorithm **FSQP′** is well defined.

Lemma 3. *Suppose $x \in X$ is not a KKT point for (P), $H = H^T > 0$, and $\eta > 0$. Then*

(i). $\langle \nabla f(x), \hat{d}(x, H, \eta) \rangle < 0$, *and*

(ii). $\langle \nabla g_j(x), \hat{d}(x, H, \eta) \rangle < 0$, *for all* $j \in I(x)$.

Proof. Both follow immediately from Lemma 2 and the fact that $\hat{d}(x, H, \eta)$ and $\hat{\gamma}(x, H, \eta)$ must satisfy the constraints in $\widehat{QP}(x, H, \eta)$. □

Lemma 4. *If $\eta_k = 0$, then x_k is a KKT point for (P) and the algorithm will stop in Step 1(i) at iteration k. On the other hand, whenever the algorithm does not stop in Step 1(i), the line search is well defined, i.e. Step 2 yields a step $t_k = \beta^j$ for some finite $j = j(k)$.*

Proof. Suppose that $\eta_k = 0$. Then $k > 0$ and, by *Step 3(iii)*, either $\widehat{d_k^0} = 0$ with $\widehat{\lambda_k^0} \geq 0$, or $\hat{d}_{k-1} = 0$. The latter case cannot hold, as the stopping criterion in *Step 1(i)* would have stopped the algorithm at iteration $k-1$. On the other hand, if $\widehat{d_k^0} = 0$ with $\widehat{\lambda_k^0} \geq 0$, then in view of the optimality conditions (5), and the fact that x_k is always feasible for (P), we see that x_k is a KKT point for (P) with multipliers

$$\lambda^j = \begin{cases} \widehat{\lambda_k^0}^j, & j \in \hat{I}_{k-1}, \\ 0, & \text{otherwise.} \end{cases}$$

Thus, by Lemma 2, $\hat{d}_k = 0$ and the algorithm will stop in *Step 1(i)*. The first claim is thus proved. Also, we have established that $\eta_k > 0$ whenever *Step 2* is reached. The second claim now follows immediately from Lemma 3 and Assumption 2. □

The previous lemmas imply that the algorithm is well-defined. In addition, Lemma 2 shows that if Algorithm **FSQP′** generates a finite sequence terminating at the point x_N, then x_N is a KKT point for the problem (P). We now concentrate on the case in which an infinite sequence $\{x_k\}$ is generated, i.e. the algorithm never satisfies the termination condition in *Step 1(i)*. Note that, in view of Lemma 4, we may assume throughout that

$$\eta_k > 0, \quad \forall k \in \mathbb{N}. \tag{7}$$

Given an infinite index set $\mathcal{K}$, we will use the notation

$$x_k \xrightarrow{k\in\mathcal{K}} x^*$$

to mean

$$x_k \to x^* \quad \text{as} \quad k \to \infty, \; k \in \mathcal{K}.$$

Lemma 5. *Suppose $\mathcal{K} \subseteq \mathbb{N}$ is an infinite index set such that $x_k \xrightarrow{k\in\mathcal{K}} x^* \in X$, $H_k \xrightarrow{k\in\mathcal{K}} H^* > 0$, $\{\eta_k\}$ is bounded on $\mathcal{K}$, and $\hat{d}_k \xrightarrow{k\in\mathcal{K}} 0$. Then $\hat{I}_k \subseteq I(x^*)$, for all $k \in \mathcal{K}$, k sufficiently large and the QP multiplier sequences $\{\hat{\mu}_k\}$ and $\{\hat{\lambda}_k\}$ are bounded on $\mathcal{K}$. Further, given any accumulation point $\eta^* \geq 0$ of $\{\eta_k\}_{k\in\mathcal{K}}$, $(0,0)$ is the unique solution of $\widehat{QP}(x^*, H^*, \eta^*)$.*

Proof. It follows immediately from non-negativity and (3) that $\{\hat{\mu}_k\}_{k\in\mathcal{K}}$ is bounded. Assumption 2 allows us to conclude that $\{\nabla f(x_k)\}_{k\in\mathcal{K}}$ is bounded. Lemma 2 and the first constraint in $\widehat{QP}(x_k, H_k, \eta_k)$ give

$$\langle \nabla f(x_k), \hat{d}_k \rangle \leq \hat{\gamma}_k \leq 0, \quad \forall k \in \mathcal{K}.$$

Thus, $\hat{\gamma}_k \xrightarrow{k\in\mathcal{K}} 0$. Next, we will show that $\hat{I}_k \subseteq I(x^*)$, for all $k \in \mathcal{K}$, k sufficiently large. Consider $j' \notin I(x^*)$. There exists $\delta_{j'} > 0$ such that $g_{j'}(x_k) \leq -\delta_{j'} < 0$, for all $k \in \mathcal{K}$, k sufficiently large. In view of Assumption 2, and since $\hat{d}_k \xrightarrow{k\in\mathcal{K}} 0$, $\hat{\gamma}_k \xrightarrow{k\in\mathcal{K}} 0$, and $\{\eta_k\}$ is bounded on $\mathcal{K}$, it is clear that

$$g_{j'}(x_k) + \langle \nabla g_{j'}(x_k), \hat{d}_k \rangle - \hat{\gamma}_k \cdot \eta_k \leq -\frac{\delta_{j'}}{2} < 0,$$

i.e. $j' \notin \hat{I}_k$, for all $k \in \mathcal{K}$, k sufficiently large. Hence, $\hat{I}_k \subseteq I(x^*)$, for all $k \in \mathcal{K}$, k sufficiently large, which proves the first claim of the Lemma.

Boundedness of $\{\hat{\mu}_k\}_{k\in\mathcal{K}}$ has been proved. To prove that of $\{\hat{\lambda}_k\}_{k\in\mathcal{K}}$, using complementary slackness, and the first equation in (2), write

$$H_k \hat{d}_k + \hat{\mu}_k \nabla f(x_k) + \sum_{j\in I(x^*)} \hat{\lambda}_k^j \nabla g_j(x_k) = 0. \tag{8}$$

Proceeding by contradiction, suppose that $\{\hat{\lambda}_k\}_{k\in\mathcal{K}}$ is unbounded. Without loss of generality, assume that $\|\hat{\lambda}_k\|_\infty > 0$, for all $k \in \mathcal{K}$ and define for all $k \in \mathcal{K}$

$$\nu_k^j \triangleq \frac{\hat{\lambda}_k^j}{\|\hat{\lambda}_k\|_\infty} \in [0,1].$$

Note that, for all $k \in \mathcal{K}$, $\|\nu_k\|_\infty = 1$. Dividing (8) by $\|\hat{\lambda}_k\|_\infty$ and taking limits on an appropriate subsequence of $\mathcal{K}$, it follows that

$$\sum_{j \in I(x^*)} \nu^{*,j} \nabla g_j(x^*) = 0,$$

for some $\nu^{*,j}$, $j \in I(x^*)$, where $\|\nu^*\|_\infty = 1$. As this contradicts Assumption 3, it is established that $\{\hat{\lambda}_k\}_{k \in \mathcal{K}}$ is bounded.

To complete the proof, let $\mathcal{K}' \subseteq \mathcal{K}$ be an infinite index set such that $\eta_k \xrightarrow{k \in \mathcal{K}'} \eta^*$ and assume without loss of generality that $\hat{\mu}_k \xrightarrow{k \in \mathcal{K}'} \hat{\mu}^*$ and $\hat{\lambda}_k \xrightarrow{k \in \mathcal{K}'} \hat{\lambda}^*$. Taking limits in the optimality conditions (2) shows that, indeed, $(\hat{d}, \hat{\gamma}) = (0, 0)$ is a KKT point for $\widehat{QP}(x^*, H^*, \eta^*)$ with multipliers $\hat{\mu}^*$ and $\hat{\lambda}^*$. Finally, uniqueness of such points (Lemma 1) proves the result. □

Before proceeding, we make an assumption concerning the estimates H_k of the Hessian of the Lagrangian.

Assumption 4: There exists constants $0 < \sigma_1 \leq \sigma_2$ such that, for all k,

$$\sigma_1 \|d\|^2 \leq \langle d, H_k d \rangle \leq \sigma_2 \|d\|^2, \quad \forall d \in \mathbb{R}^n.$$

Lemma 6. *The sequences $\{H_k\}$ and $\{\eta_k\}$ generated by Algorithm* **FSQP′** *are bounded. Further, the sequence $\{\hat{d}_k\}$ is bounded on subsequences on which $\{x_k\}$ is bounded.*

Proof. That $\{H_k\}$ is bounded follows immediately from Assumption 4. *Step 3(iii)* of Algorithm **FSQP′** ensures that the sequence $\{\eta_k\}$ is bounded. Finally, it then follows from Lemma 1 that $\{\hat{d}_k\}$ is bounded on subsequences on which $\{x_k\}$ is bounded. □

Lemma 7. *If $\mathcal{K} \subseteq \mathbb{N}$ is an infinite index set such that $\hat{d}_k \xrightarrow{k \in \mathcal{K}} 0$, then all accumulation points of $\{x_k\}_{k \in \mathcal{K}}$ are KKT points for (P).*

Proof. Suppose $\mathcal{K}' \subseteq \mathcal{K}$ is an infinite index set on which $x_k \xrightarrow{k \in \mathcal{K}'} x^* \in X$. In view of Lemma 6, assume, without loss of generality that $H_k \xrightarrow{k \in \mathcal{K}'} H^* > 0$ and $\eta_k \xrightarrow{k \in \mathcal{K}'} \eta^* \geq 0$. Lemma 5 shows that $(0, 0)$ is the unique solution of $\widehat{QP}(x^*, H^*, \eta^*)$. Thus, in view of Lemma 2, x^* is a KKT point for (P). □

We now state and prove the main result of this section.

Theorem 1. *Under the stated assumptions, Algorithm* **FSQP′** *generates a sequence* $\{x_k\}$ *for which all accumulation points are KKT points for* (P).

Proof. Suppose $\mathcal{K} \subseteq \mathbb{N}$ is an infinite index set such that $x_k \xrightarrow{k\in\mathcal{K}} x^*$. In view of Lemma 6, we may assume without loss of generality that $\hat{d}_k \xrightarrow{k\in\mathcal{K}} \hat{d}^*$, $\eta_k \xrightarrow{k\in\mathcal{K}} \eta^* \geq 0$, and $H_k \xrightarrow{k\in\mathcal{K}} H^* > 0$. The cases $\eta^* = 0$ and $\eta^* > 0$ are considered separately.

Suppose first that $\eta^* = 0$. Then, by *Step 3(iii)*, either $\widehat{d_k^0} \xrightarrow{k\in\mathcal{K}} 0$ with $\widehat{\lambda_k^0} \geq 0$, for all $k \in \mathcal{K}$, k large enough, or $\hat{d}_{k-1} \xrightarrow{k\in\mathcal{K}} 0$. If the latter case holds, it is then clear that $x_{k-1} \xrightarrow{k\in\mathcal{K}} x^*$, since $\|x_k - x_{k-1}\| \leq 2\|\hat{d}_{k-1}\| \xrightarrow{k\in\mathcal{K}} 0$. Thus, by Lemma 7, x^* is a KKT point for (P). Now suppose instead that $\widehat{d_k^0} \xrightarrow{k\in\mathcal{K}} 0$ with $\widehat{\lambda_k^0} \geq 0$, for all $k \in \mathcal{K}$, k large enough. Using an argument very similar to that used in Lemma 5, we can show that $\{\widehat{\lambda_k^0}\}_{k\in\mathcal{K}}$ is a bounded sequence and $\hat{I}_{k-1} \subseteq I(x^*)$, for all $k \in \mathcal{K}$, k sufficiently large. Thus, taking limits in (5) on an appropriate subsequence of $\mathcal{K}$ shows that x^* is a KKT point for (P).

Now consider the case $\eta^* > 0$. We will show that $\hat{d}_k \xrightarrow{k\in\mathcal{K}} 0$. Proceeding by contradiction, without loss of generality suppose there exists $\underline{d} > 0$ such that $\|\hat{d}_k\| \geq \underline{d}$ for all $k \in \mathcal{K}$. Thus, from non-positivity of the optimal value of the objective function in $\widehat{QP}(x_k, H_k, \eta_k)$ (since $(0,0)$ is always feasible) and Assumption 4, we see that

$$\hat{\gamma}_k \leq -\frac{1}{2}\sigma_1 \underline{d}^2 < 0, \quad \forall k \in \mathcal{K}.$$

Further, in view of (7) and since $\eta^* > 0$, there exists $\underline{\eta} > 0$ such that

$$\eta_k > \underline{\eta}, \quad \forall k \in \mathcal{K}.$$

From the constraints of $\widehat{QP}(x_k, H_k, \eta_k)$, it follows that

$$\langle \nabla f(x_k), \hat{d}_k \rangle \leq -\frac{1}{2}\sigma_1 \underline{d}^2 < 0, \quad \forall k \in \mathcal{K},$$

and

$$g_j(x_k) + \langle \nabla g_j(x_k), \hat{d}_k \rangle \leq -\frac{1}{2}\sigma_1 \underline{d}^2 \underline{\eta} < 0, \quad \forall k \in \mathcal{K},$$

$j = 1, \ldots, m$. Hence, using Assumption 2, it is easily shown that there exists $\delta > 0$ such that for all $k \in \mathcal{K}$, k large enough,

$$\langle \nabla f(x_k), \hat{d}_k \rangle \leq -\delta,$$

$$\langle \nabla g_j(x_k), \hat{d}_k \rangle \leq -\delta, \quad \forall j \in I(x^*)$$

$$g_j(x_k) \leq -\delta, \quad \forall j \in \{1, \ldots, m\} \setminus I(x^*).$$

The rest of the contradiction argument establishing $\hat{d}_k \xrightarrow{k \in \mathcal{K}} 0$ follows exactly the proof of Proposition 3.2 in [11]. Finally, it then follows from Lemma 7 that x^* is a KKT point for (P). $\square$

3.2 Local Convergence

While the details are often quite different, overall the analysis in this section is inspired by and occasionally follows that of Panier and Tits in [11, 12]. In order to establish a rate of convergence for the algorithm, we first strengthen the regularity assumptions.

Assumption 2′: The functions $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \ldots, m$, are three times continuously differentiable.

A point x^* is said to satisfy the *second order sufficiency conditions with strict complementary slackness* for (P) if there exists a multiplier vector $\lambda^* \in \mathbb{R}^m$ such that

- The pair (x^*, λ^*) satisfies (1), i.e. x^* is a KKT point for (P),
- $\nabla^2_{xx} L(x^*, \lambda^*)$ is positive definite on the subspace

$$\{h \mid \langle \nabla g_j(x^*), h \rangle = 0, \ \forall j \in I(x^*)\},$$

- and $\lambda^{*,j} > 0$ for all $j \in I(x^*)$ (strict complementary slackness).

In order to guarantee that the entire sequence $\{x_k\}$ converges to a KKT point x^*, we make the following assumption. Recall that we have already established, under weaker assumptions, that *every* accumulation point of $\{x_k\}$ is a KKT point for (P).

Assumption 5: The sequence $\{x_k\}$ has an accumulation point x^* which satisfies the second order sufficiency conditions with strict complementary slackness.

It is well known, and not difficult to show, that Assumption 5 guarantees the entire sequence converges. For a proof see, e.g., Proposition 4.1 in [11]. We state the result here without proof.

Lemma 8. *The entire sequence generated by Algorithm* **FSQP′** *converges to a point x^* satisfying the second order sufficiency conditions with strict complementary slackness.*

From this point forward, λ^* will denote the (unique) multiplier vector satisfying the KKT conditions for (P) at x^*. Further, we need to strengthen the assumptions concerning the sequence $\{H_k\}$.

Assumption 6: The sequence $\{H_k\}$ converges to some $H^* = H^{*T} > 0$.

In order to establish a rate of convergence, we will show that our sequence of tilted SQP directions approaches the true SQP direction, for which asymptotic rates of convergence are well known, sufficiently fast. In order to do so, define $d_k^0 = d^0(x_k, H_k)$, where x_k and H_k are as computed by Algorithm **FSQP'** . Further, for each k, define $\lambda_k^0 \in \mathbb{R}^m$ as a multiplier vector satisfying (4) at d_k^0 and let $I_k^0 \triangleq \{\ j \mid g_j(x_k) + \langle \nabla g_j(x_k), d_k^0 \rangle = 0\ \}$. The following Lemma is proved in [11, 12] under identical assumptions.

Lemma 9.

(i) $d_k^0 \to 0$,

(ii) $\lambda_k^0 \to \lambda^*$.

(iii) For all k sufficiently large, the following two equalities hold

$$I_k^0 = \{\ j \mid \lambda_k^{0,j} > 0\ \} = I(x^*).$$

Before proceeding, we state one more well-known result that will be called upon several times throughout the balance of the analysis. First, we make the definitions

$$\begin{aligned} R_k &\triangleq [\ \nabla g_j(x_k)\ :\ j \in I(x^*)\ ], \\ g_k &\triangleq [\ g_j(x_k)\ :\ j \in I(x^*)\ ]^T. \end{aligned}$$

Lemma 10. *Under the stated assumptions, the matrix*

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix}$$

is uniformly invertible, i.e. it is invertible for all k and its singular values are bounded away from 0 for all k sufficiently large.

We now establish that the entire tilted SQP direction sequence converges to 0. In order to do so, we establish that $\hat{d}(x, H, \eta)$ is continuous in a neighborhood of (x^*, H^*, η^*), for any $\eta^* \geq 0$. Complicating the analysis is

the fact that we have yet to establish that the sequence $\{\eta_k\}$ does, in fact, converge. Given $\eta^* \geq 0$, define the set

$$N^*(\eta^*) \triangleq \left\{ \begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix}, \begin{pmatrix} \nabla g_j(x^*) \\ -\eta^* \end{pmatrix}, j \in I(x^*) \right\}.$$

Lemma 11. *Given any $\eta^* \geq 0$, the set $N^*(\eta^*)$ is linearly independent.*

Proof. Note that, in view of Lemma 2, $\hat{d}^* = \hat{d}(x^*, H^*, \eta^*) = 0$. Now suppose the Lemma does not hold, i.e. suppose there exists scalars λ^j, $j \in \{0\} \cup I(x^*)$, not all zero, such that

$$\lambda^0 \begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix} + \sum_{j \in I(x^*)} \lambda^j \begin{pmatrix} \nabla g_j(x^*) \\ -\eta^* \end{pmatrix} = 0. \tag{9}$$

In view of Assumption 3, $\lambda^0 \neq 0$ and the scalars λ^j are unique modulo a scaling factor. This uniqueness, the fact that $\hat{d}^* = 0$, and the optimality conditions (2) imply that $\hat{\mu}^* = 1$ and

$$\hat{\lambda}^{*,j} = \begin{cases} \dfrac{\lambda^j}{\lambda^0} & j \in I(x^*) \\ 0 & \text{else,} \end{cases}$$

$j = 1, \ldots, m$ are KKT multipliers for $\widehat{QP}(x^*, H^*, \eta^*)$. Thus, in view of (3),

$$\eta^* \cdot \sum_{j \in I(x^*)} \frac{\lambda^j}{\lambda^0} = 0.$$

But this contradicts (9), which gives

$$\eta^* \cdot \sum_{j \in I(x^*)} \frac{\lambda^j}{\lambda^0} = -1,$$

hence $N^*(\eta^*)$ is linearly independent. □

Lemma 12. *Let $\eta^* \geq 0$ be an accumulation point of $\{\eta_k\}$. Then $(\hat{d}^*, \hat{\gamma}^*) = (0, 0)$ is the unique solution of $\widehat{QP}(x^*, H^*, \eta^*)$ and the second order sufficiency conditions hold, with strict complementary slackness.*

Proof. In view of Lemma 2, $\widehat{QP}(x^*, H^*, \eta^*)$ has $(\hat{d}^*, \hat{\gamma}^*) = (0, 0)$ as its unique solution. Define the Lagrangian function $\hat{L}^* : \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}$ for $\widehat{QP}(x^*, H^*, \eta^*)$ as

$$\begin{aligned} \hat{L}^*(\hat{d}, \hat{\gamma}, \hat{\mu}, \hat{\lambda}) &= \frac{1}{2}\langle \hat{d}, H^* \hat{d}\rangle + \hat{\gamma} + \hat{\mu}\left(\langle \nabla f(x^*), \hat{d}\rangle - \hat{\gamma}\right) \\ &\quad + \sum_{j=1}^{m} \hat{\lambda}^j \left(g_j(x^*) + \langle \nabla g_j(x^*), \hat{d}\rangle - \hat{\gamma}\eta^*\right). \end{aligned}$$

Suppose $\hat{\mu}^* \in \mathbb{R}$ and $\hat{\lambda}^* \in \mathbb{R}^m$ are multipliers satisfying (2) at $(\hat{d}^*, \hat{\gamma}^*)$. Let $j = 0$ be the index for the first constraint in $\widehat{QP}(x^*, H^*, \eta^*)$, i.e. $\langle \nabla f(x^*), \hat{d}\rangle \leq \hat{\gamma}$. Note that since $(\hat{d}^*, \hat{\gamma}^*) = (0, 0)$, the active constraint index set[3] $\hat{I}^*$ for $\widehat{QP}(x^*, H^*, \eta^*)$ is equal to $I(x^*)$, the active constraint index set for (P) at x^*, in addition to $j = 0$. Thus the set of active constraint gradients for $\widehat{QP}(x^*, H^*, \eta^*)$ is $N^*(\eta^*)$.

Now consider the Hessian of the Lagrangian for $\widehat{QP}(x^*, H^*, \eta^*)$, i.e. the second derivative with respect to the first two variables $(\hat{d}, \hat{\gamma})$,

$$\nabla^2 \hat{L}^*(0, 0, \hat{\lambda}^*, \hat{\mu}^*) = \begin{bmatrix} H^* & 0 \\ 0 & 0 \end{bmatrix},$$

and given an arbitrary $h \in \mathbb{R}^{n+1}$, decompose it as $h = (y^T, \alpha)^T$, where $y \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$. Then clearly,

$$h^T \nabla^2 \hat{L}^*(0, 0, \hat{\lambda}^*, \hat{\mu}^*) h \geq 0, \quad \forall h$$

and for $h \neq 0$, $h^T \nabla^2 \hat{L}^*(0, 0, \hat{\lambda}^*, \hat{\mu}^*) h = y^T H^* y$ is zero if, and only if, $y = 0$ and $\alpha \neq 0$. Since

$$\begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix}^T \begin{pmatrix} 0 \\ \alpha \end{pmatrix} = -\alpha \neq 0,$$

it then follows that $\nabla^2 \hat{L}^*(0, 0, \hat{\lambda}^*, \hat{\mu}^*)$ is positive definite on $N^*(\eta^*)^\perp$, the tangent space to the active constraints for $\widehat{QP}(x^*, H^*, \eta^*)$ at $(0, 0)$. Thus, it is established that the second order sufficiency conditions hold. We next show that strict complementary slackness holds.

First, $\hat{\mu}^* > 0$. Indeed, suppose to the contrary that $\hat{\mu}^* = 0$. In view of (3), this implies there exists an index $j' \in \hat{I}^*$ such that $\hat{\lambda}^{*,j'} > 0$. Recalling that $\hat{I}^* = I(x^*) \cup \{0\}$ and invoking complementary slackness for

[3] We are temporarily abandoning our convention of omitting the objective descent constraint in $\hat{I}$ for this argument only.

$\widehat{QP}(x^*, H^*, \eta^*)$, the first equation in (2) gives

$$\sum_{j \in I(x^*)} \hat{\lambda}^{*,j} \nabla g_j(x^*) = 0.$$

As $\hat{\lambda}^{*,j'} > 0$ for some $j' \in \hat{I}^*$, this contradicts Assumption 3. Next, a well-known consequence of Assumption 3 is that the KKT multipliers $\lambda^{*,j}$ for (P) at x^* are unique. Thus, it follows from the optimality conditions (2) and (1) that $\hat{\lambda}^{*,j} = \hat{\mu}^* \cdot \lambda^{*,j}$, $j = 1, \ldots, m$. Further, it follows from Assumption 5 that $\hat{\lambda}^{*,j} > 0$, $j \in I(x^*)$, i.e. strict complementary slackness is satisfied by $\widehat{QP}(x^*, H^*, \eta^*)$ at $(0, 0)$. □

Lemma 13. *If $\mathcal{K}$ is a subsequence on which $\{\eta_k\}$ converges, say to $\eta^* \geq 0$, then $\hat{\mu}_k \xrightarrow{k \in \mathcal{K}} \hat{\mu}^* > 0$ and $\hat{\lambda}_k \xrightarrow{k \in \mathcal{K}} \hat{\mu}^* \cdot \lambda^*$, where $\hat{\mu}^* = \hat{\mu}^*(\eta^*)$ is the KKT multiplier for the first constraint of $\widehat{QP}(x^*, H^*, \eta^*)$. Finally, $\hat{d}_k \to 0$ and $\hat{\gamma}_k \to 0$.*

Proof. In view of Lemmas 11 and 12, we may invoke a result due to Robinson (Theorem 2.1 in [18]) to conclude

$$(\hat{d}_k, \hat{\gamma}_k) \xrightarrow{k \in \mathcal{K}} (0, 0), \quad \hat{\mu}_k \xrightarrow{k \in \mathcal{K}} \hat{\mu}^*, \quad \text{and} \quad \frac{\hat{\lambda}_k}{\hat{\mu}^*} \xrightarrow{k \in \mathcal{K}} \lambda^*.$$

It is important to note that $\hat{\mu}^*$ is a function of η^*, i.e. $\hat{\mu}^* = \hat{\mu}^*(\eta^*)$. Now suppose that the last claim of the lemma does not hold. If $\hat{d}_k \not\to 0$, there exists an infinite index set $\mathcal{K} \subseteq \mathbb{N}$ and $\underline{d} > 0$ such that $\|\hat{d}_k\| \geq \underline{d}$, for all $k \in \mathcal{K}$. As $\{\eta_k\}_{k \in \mathcal{K}}$ is bounded, there exists an infinite index set $\mathcal{K}' \subseteq \mathcal{K}$ and $\eta^* \geq 0$ such that $\eta_k \xrightarrow{k \in \mathcal{K}'} \eta^*$. By what we showed above, $\hat{d}_k \xrightarrow{k \in \mathcal{K}'} 0$, which is a contradiction, hence $\hat{d}_k \to 0$. It immediately follows from the first constraint of $\widehat{QP}(x_k, H_k, \eta_k)$ that $\hat{\gamma}_k \to 0$. □

Lemma 14. *For all k sufficiently large, $\hat{I}_k = I(x^*)$.*

Proof. Since $\{\eta_k\}$ is bounded and $(\hat{d}_k, \hat{\gamma}_k) \to (0, 0)$, in view of Lemma 5, $\hat{I}_k \subseteq I(x^*)$, for all k sufficiently large. Now suppose it does not hold that $\hat{I}_k = I(x^*)$ for all k sufficiently large. Thus, there exists $j' \in I(x^*)$ and an infinite index set $\mathcal{K} \subseteq \mathbb{N}$ such that $j' \notin \hat{I}_k$, for all $k \in \mathcal{K}$. Now, in view of Lemma 6, there exists an infinite index set $\mathcal{K}' \subseteq \mathcal{K}$ and $\eta^* \geq 0$ such that $\eta_k \xrightarrow{k \in \mathcal{K}'} \eta^*$. Since $j' \in I(x^*)$, Assumption 5 guarantees $\lambda^{*,j'} > 0$. Further, Lemma 13 shows that $\hat{\lambda}_k^{j'} \xrightarrow{k \in \mathcal{K}'} \hat{\mu}^*(\eta^*) \cdot \lambda^{*,j'} > 0$. Therefore, $\hat{\lambda}_k^{j'} > 0$ for all k sufficiently large, $k \in \mathcal{K}'$, which, by complementary slackness, implies $j' \in \hat{I}_k$

for all $k \in \mathcal{K}'$ large enough. Since $\mathcal{K}' \subseteq \mathcal{K}$, this is a contradiction, hence $\hat{I}_k = I(x^*)$, for all k sufficiently large. $\square$

Given a vector $\lambda \in \mathbb{R}^m$, define the notation

$$\lambda^+ \triangleq [\ \lambda^j \ : \ j \in I(x^*)\ ]^T.$$

Note that, in view of Lemma 9(*iii*), for k large enough, the optimality conditions (4), yield

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^0 \\ (\lambda_k^0)^+ \end{pmatrix} = - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix}. \tag{10}$$

Lemma 15. *For all k sufficiently large, $\widehat{d_k^0} = d_k^0$.*

Proof. In view of Lemma 14 and the optimality conditions (5), the estimate $\widehat{d_k^0}$ and its corresponding multiplier vector $\widehat{\lambda_k^0}$ (recall that for ease of notation we defined $\widehat{\lambda_k^0} \in \mathbb{R}^m$) satisfy

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} \widehat{d_k^0} \\ (\widehat{\lambda_k^0})^+ \end{pmatrix} = - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix}, \tag{11}$$

for all k sufficiently large. In view of (10), the result then follows from Lemma 10. $\square$

Lemma 16.

(i) $\eta_k \to 0$,

(ii) $\hat{\mu}_k \to 1$, *and* $\hat{\lambda}_k \to \lambda^*$.

(iii) For all k sufficiently large, $\hat{I}_k = \{\ j \mid \hat{\lambda}_k^j > 0\ \}$.

Proof. Claim (*i*) follows from *Step 3(iii)* of Algorithm **FSQP′**, since in view of Lemma 13, Lemma 15, and Lemma 9, $\{\hat{d}_k\}$ and $\{\widehat{d_k^0}\}$ both converge to 0. In view of (*i*), Lemma 13 establishes that $\hat{\mu}_k \to \hat{\mu}^*(0)$, and $\hat{\lambda}_k \to \hat{\mu}^*(0) \cdot \lambda^*$. That $\hat{\mu}^*(0) = 1$ follows from (3), hence claim (*ii*) is proved. Finally, claim (*iii*) follows from claim (*ii*), Lemma 14, and Assumption 5. $\square$

We now focus our attention on establishing relationships between $\hat{d}_k$, $\tilde{d}_k$, and the true SQP direction d_k^0.

Lemma 17.

(i) $\eta_k = O(\|d_k^0\|^2)$,

(ii) $\hat{d}_k = d_k^0 + O(\|d_k^0\|^2)$.

(iii) $\hat{\gamma}_k = O(\|d_k^0\|)$.

Proof. In view of Lemma 15, $\widehat{d_k^0}$ exists and $\widehat{d_k^0} = d_k^0$ for all k sufficiently large. Lemmas 13 and 9 ensure that *Step 3(iii)* of Algorithm **FSQP′** chooses $\eta_k = C_k \cdot \|\widehat{d_k^0}\|^2$ for all k sufficiently large, thus (i) follows. It is clear from Lemma 14 and the optimality conditions (2) that $\hat{d}_k$ and $\hat{\lambda}_k$ satisfy

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} \hat{d}_k \\ \hat{\lambda}_k^+ \end{pmatrix} = - \begin{pmatrix} \hat{\mu}_k \cdot \nabla f(x_k) \\ g_k - \eta_k \cdot \hat{\gamma}_k \cdot \mathbf{1}_{|I(x^*)|} \end{pmatrix}$$

$$= - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix} + \eta_k \cdot \begin{pmatrix} \left(\sum_{j \in I(x^*)} \hat{\lambda}_k^j \right) \cdot \nabla f(x_k) \\ \hat{\gamma}_k \cdot \mathbf{1}_{|I(x^*)|} \end{pmatrix}, \quad (12)$$

for all k sufficiently large, where $\mathbf{1}_{|I(x^*)|}$ is a vector of $|I(x^*)|$ ones. It thus follows from (10) that

$$\hat{d}_k = d_k^0 + O(\eta_k),$$

and in view of claim (i), claim (ii) follows. Finally, since (from the QP constraint and Lemma 2) $\langle \nabla f(x_k), \hat{d}_k \rangle \leq \hat{\gamma}_k < 0$, it is clear that $\hat{\gamma}_k = O(\|\hat{d}_k\|) = O(\|d_k^0\|)$. □

Lemma 18. $\tilde{d}_k = O(\|d_k^0\|^2)$.

Proof. Let

$$c_k \triangleq [-g_j(x_k + \hat{d}_k) - \|\hat{d}_k\|^\tau \ : \ j \in I(x^*)]^T.$$

Expanding $g_j(\cdot)$, $j \in I(x^*)$, about x_k we see that, for some $\xi^j \in (0,1)$, $j \in I(x^*)$,

$$c_k = [\ \overbrace{-g_j(x_k) - \langle \nabla g_j(x_k), \hat{d}_k \rangle}^{=-\eta_k \cdot \hat{\gamma}_k} + \frac{1}{2} \langle \hat{d}_k, \nabla^2 g_j(x_k + \xi^j \hat{d}_k) \hat{d}_k \rangle - \|\hat{d}_k\|^\tau \ : \ j \in I(x^*)\ ]^T.$$

Since $\tau > 2$, from Lemma 17, we conclude $c_k = O(\|d_k^0\|^2)$. Now, for all k sufficiently large, $\hat{I}_k = I(x^*)$, $\tilde{d}_k$ is well-defined and satisfies

$$g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k), \tilde{d}_k \rangle = -\|\hat{d}_k\|^\tau, \quad j \in I(x^*), \tag{13}$$

thus, we have established

$$R_k^T \tilde{d}_k = O(\|d_k^0\|^2). \tag{14}$$

The first order KKT conditions for $\widetilde{LS}(x_k, \hat{d}_k, H_k, \hat{I}_k)$ tell us there exists a multiplier $\tilde{\lambda}_k \in \mathbb{R}^{|I(x^*)|}$ satisfying

$$\begin{cases} H_k(\hat{d}_k + \tilde{d}_k) + \nabla f(x_k) + R_k \tilde{\lambda}_k = 0, \\ R_k^T \tilde{d}_k = c_k. \end{cases}$$

Also, from the optimality conditions (12) we have

$$H_k \hat{d}_k + \nabla f(x_k) = q_k - R_k \hat{\lambda}_k^+,$$

where

$$q_k \triangleq \eta_k \cdot \left(\sum_{j \in I(x^*)} \hat{\lambda}_k^j \right) \cdot \nabla f(x_k).$$

So, $\tilde{d}_k$ and $\tilde{\lambda}_k$ satisfy

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} \tilde{d}_k \\ \tilde{\lambda}_k \end{pmatrix} = \begin{pmatrix} R_k \hat{\lambda}_k^+ - q_k \\ c_k \end{pmatrix}.$$

Solving for $\tilde{d}_k$, after a little algebra we obtain

$$\begin{aligned} \tilde{d}_k &= H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} c_k \\ &\quad + \left[H_k^{-1} - H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} R_k^T H_k^{-1} \right] (R_k \hat{\lambda}_k^+ - q_k) \\ &= H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} c_k \\ &\quad + \left[H_k^{-1} - H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} R_k^T H_k^{-1} \right] (-q_k). \end{aligned}$$

Further, in view of Lemma 17 and since all sequences are bounded, $q_k = O(\|d_k^0\|^2)$. Thus, $\tilde{d}_k$ equivalently satisfies

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} \tilde{d}_k \\ \lambda_k' \end{pmatrix} = \begin{pmatrix} -q_k \\ c_k \end{pmatrix} = O(\|d_k^0\|^2),$$

for some $\lambda_k' \in \mathbb{R}^{|I(x^*)|}$. The result then follows from Lemma 10. $\square$

We now add one additional assumption to ensure that the matrices $\{H_k\}$ suitably approximate the Hessian of the Lagrangian at the solution. Define the projection

$$P_k \triangleq I - R_k(R_k^T R_k)^{-1} R_k^T.$$

Assumption 7:

$$\lim_{k\to\infty} \frac{\|P_k(H_k - \nabla^2_{xx} L(x^*, \lambda^*))P_k \hat{d}_k\|}{\|\hat{d}_k\|} = 0.$$

The following technical lemma will be needed in order to establish that eventually the step of one is always accepted by the line search.

Lemma 19. *There exists constants $\nu_1,\ \nu_2,\ \nu_3 > 0$ such that*

(i) $\langle \nabla f(x_k), \hat{d}_k \rangle \leq -\nu_1 \|d_k^0\|^2$,

(ii) for all k sufficiently large

$$\sum_{j=1}^{m} \hat{\lambda}_k^j g_j(x_k) \leq -\nu_2 \|g_k\|,$$

(iii) $\hat{d}_k = P_k \hat{d}_k + d_k^1$, *where*

$$\|d_k^1\| \leq \nu_3 \|g_k\| + O(\|d_k^0\|^3),$$

for all k sufficiently large.

Proof. To show part (i), note that in view of the first QP constraint, negativity of the optimal value of the QP objective, and Assumption 4,

$$\begin{aligned} \langle \nabla f(x_k), \hat{d}_k \rangle &\leq \hat{\gamma}_k \\ &\leq -\tfrac{1}{2}\langle \hat{d}_k, H_k \hat{d}_k \rangle \\ &\leq -\tfrac{\sigma_1}{2}\|\hat{d}_k\|^2 = -\tfrac{\sigma_1}{2}\|d_k^0\|^2 + O(\|d_k^0\|^4). \end{aligned}$$

The proof of part (ii) is identical to that of Lemma 4.4 in [11]. To show (iii), note that from (12) for all k sufficiently large, $\hat{d}_k$ satisfies

$$R_k^T \hat{d}_k = -g_k - \hat{\gamma}_k \eta_k \cdot \mathbf{1}_{|I(x^*)|}.$$

Thus, we can write $\hat{d}_k = P_k \hat{d}_k + d_k^1$, where

$$d_k^1 = -R_k(R_k^T R_k)^{-1}(g_k + \hat{\gamma}_k \eta_k \cdot \mathbf{1}_{|I(x^*)|}).$$

The result follows from Assumption 3. □

Lemma 20. *For all k sufficiently large, $t_k = 1$.*

Proof. Following [11], consider an expansion of $g_j(\cdot)$ about $x_k + \hat{d}_k$ for $j \in I(x^*)$, for all k sufficiently large,

$$\begin{aligned} g_j(x_k + \hat{d}_k + \tilde{d}_k) &= g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k + \hat{d}_k), \tilde{d}_k \rangle + O(\|d_k^0\|^4) \\ &= g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k), \tilde{d}_k \rangle + O(\|d_k^0\|^3) \\ &= -\|\hat{d}_k\|^\tau + O(\|d_k^0\|^3) \\ &= -\|d_k^0\|^\tau + O(\|d_k^0\|^3), \end{aligned}$$

where we have used Lemmas 17 and 18, boundedness of all sequences, and the constraints from $\widetilde{LS}(x_k, \hat{d}_k, H_k, \hat{I}_k)$ ($\hat{I}_k = I(x^*)$ for all k sufficiently large by Lemma 14). As $\tau < 3$, it follows that $g_j(x_k + \hat{d}_k + \tilde{d}_k) \leq 0$, $j \in I(x^*)$, for all k sufficiently large. The same result trivially holds for $j \notin I(x^*)$. Further, we have

$$g_j(x_k + \hat{d}_k + \tilde{d}_k) = O(\|d_k^0\|^\tau), \quad j \in I(x^*). \tag{15}$$

In view of Assumption $2'$ and Lemmas 17 and 18,

$$\begin{aligned} f(x_k + \hat{d}_k + \tilde{d}_k) &= f(x_k) + \langle \nabla f(x_k), \hat{d}_k \rangle + \langle \nabla f(x_k), \tilde{d}_k \rangle \\ &\quad + \tfrac{1}{2}\langle \hat{d}_k, \nabla^2 f(x_k) \hat{d}_k \rangle + O(\|d_k^0\|^3). \end{aligned}$$

From the optimality conditions (2), Lemma 17(i), and boundedness of all sequences, we see

$$H_k \hat{d}_k + \nabla f(x_k) + \sum_{j=1}^m \hat{\lambda}_k^j \nabla g_j(x_k) = O(\|d_k^0\|^2). \tag{16}$$

Complementary slackness for $\widehat{QP}(x_k, H_k, \eta_k)$ and Lemma 17 yield

$$\hat{\lambda}_k^j \langle \nabla g_j(x_k), \hat{d}_k \rangle = -\hat{\lambda}_k^j g_j(x_k) + O(\|d_k^0\|^3). \tag{17}$$

Taking the inner product of (16) with $\hat{d}_k$, then adding and subtracting the quantity $\sum_j \hat{\lambda}_k^j \langle \nabla g_j(x_k), \hat{d}_k \rangle$, using (17), and finally multiplying the result by $\frac{1}{2}$ gives

$$\begin{aligned} \tfrac{1}{2}\langle \nabla f(x_k), \hat{d}_k \rangle &= -\frac{1}{2}\langle \hat{d}_k, H_k \hat{d}_k \rangle - \sum_{j=1}^m \hat{\lambda}_k^j \langle \nabla g_j(x_k), \hat{d}_k \rangle \\ &\quad - \frac{1}{2}\sum_{j=1}^m \hat{\lambda}_k^j g_j(x_k) + O(\|d_k^0\|^3). \end{aligned}$$

Further, Lemmas 17 and 18 and (16) give

$$\langle \nabla f(x_k), \tilde{d}_k \rangle = -\sum_{j=1}^{m} \hat{\lambda}_k^j \langle \nabla g_j(x_k), \tilde{d}_k \rangle + O(\|d_k^0\|^3).$$

Combining results, we have

$$f(x_k + \hat{d}_k + \tilde{d}_k) - f(x_k) = \tag{18}$$

$$\frac{1}{2}\langle \nabla f(x_k), \hat{d}_k \rangle - \frac{1}{2}\langle \hat{d}_k, H_k \hat{d}_k \rangle - \frac{1}{2}\sum_{j=1}^{m} \hat{\lambda}_k^j g_j(x_k) \tag{19}$$

$$-\sum_{j=1}^{m} \hat{\lambda}_k^j \langle \nabla g_j(x_k), \hat{d}_k \rangle - \sum_{j=1}^{m} \hat{\lambda}_k^j \langle \nabla g_j(x_k), \tilde{d}_k \rangle \tag{20}$$

$$+\frac{1}{2}\langle \hat{d}_k, \nabla^2 f(x_k) \hat{d}_k \rangle + O(\|d_k^0\|^3). \tag{21}$$

Expanding about x_k and using Lemmas 17(ii) and 18 and equation (15) we have

$$g_j(x_k) + \langle \nabla g_j(x_k), \hat{d}_k \rangle + \langle \nabla g_j(x_k), \tilde{d}_k \rangle =$$
$$-\frac{1}{2}\langle \hat{d}_k, \nabla^2 g_j(x_k) \hat{d}_k \rangle + O(\|d_k^0\|^\tau), \qquad j \in I(x^*),$$

since $\tau < 3$. Rearranging to give an expression for $g_j(x_k)$ and then substituting into the third term on the right-hand side of (18) for each j gives

$$f(x_k + \hat{d}_k + \tilde{d}_k) - f(x_k) =$$
$$\frac{1}{2}\langle \nabla f(x_k), \hat{d}_k \rangle + \frac{1}{2}\sum_{j=1}^{m} \hat{\lambda}_k^j g_j(x_k)$$
$$+\frac{1}{2}\hat{d}_k^T \left(\nabla^2 f(x_k) + \sum_{j=1}^{m} \hat{\lambda}_k^j \nabla^2 g_j(x_k) - H_k \right) \hat{d}_k$$
$$+O(\|d_k^0\|^\tau).$$

Subtracting $\alpha \langle \nabla f(x_k), \hat{d}_k \rangle$ from both sides and invoking Lemma 19 shows

there exists constants $\nu_2,\ \nu_3 > 0$ such that, since $\tau > 2$,

$$
\begin{aligned}
&f(x_k + \hat{d}_k + \tilde{d}_k) - f(x_k) - \alpha\langle \nabla f(x_k), \hat{d}_k\rangle \leq \\
&\quad (\frac{1}{2} - \alpha)\langle \nabla f(x_k), \hat{d}_k\rangle + \frac{1}{2}\hat{d}_k^T P_k \left(\nabla^2 f(x_k) + \sum_{j=1}^{m} \hat{\lambda}_k^j \nabla^2 g_j(x_k) - H_k \right) P_k \hat{d}_k \\
&\quad - \left(\nu_2 - \nu_3 \left(\|\hat{d}_k\| + \nu_3 \|g_k\| \right) \left\| \nabla^2 f(x_k) + \sum_{j=1}^{m} \hat{\lambda}_k^j \nabla^2 g_j(x_k) - H_k \right\| \right) \cdot \|g_k\| \\
&\quad + o(\|d_k^0\|^2).
\end{aligned}
$$

Since $\hat{d}_k \to 0$ and $g_k \to 0$ and all sequences are bounded, the third term on the right-hand side is negative for all k sufficiently large, hence

$$
\begin{aligned}
f(x_k + \hat{d}_k + \tilde{d}_k) - f(x_k) - \alpha\langle \nabla f(x_k), \hat{d}_k\rangle \leq \\
(\frac{1}{2} - \alpha)\langle \nabla f(x_k), \hat{d}_k\rangle + \frac{1}{2}\hat{d}_k^T P_k \left(\nabla^2_{xx} L(x_k, \hat{\lambda}_k) - H_k \right) P_k \hat{d}_k \\
+ o(\|d_k^0\|^2).
\end{aligned}
$$

Assumption 7 says that $P_k(\nabla^2_{xx} L(x_k, \hat{\lambda}_k) - H_k) P_k \hat{d}_k = o(\|\hat{d}_k\|)$. This, along with Lemma 19 implies

$$
\begin{aligned}
f(x_k + \hat{d}_k + \tilde{d}_k) - f(x_k) - \alpha\langle \nabla f(x_k), \hat{d}_k\rangle \\
&\leq -\nu_1(\frac{1}{2} - \alpha)\|d_k^0\|^2 + o(\|d_k^0\|^2) \\
&\leq 0,
\end{aligned}
$$

for all k sufficiently large. Thus we have shown that the conditions of the line search in *Step 2* are satisfied with $t_k = 1$ for all k sufficiently large. □

A consequence of Lemmas 17, 18, and 20 is that the algorithm generates a convergent sequence of iterates satisfying

$$
x_{k+1} - x_k = d_k^0 + O(\|d_k^0\|^2). \tag{22}
$$

This allows us to apply, with some modification, the argument used by Powell in [15] to establish a 2-step superlinear rate of convergence, the main result of this section. The modification of Powell's argument to our case is given in the appendix.

Theorem 2. *Algorithm* **FSQP′** *generates a sequence* $\{x_k\}$ *which converges 2-step superlinearly to* x^**, i.e.*

$$\lim_{k\to\infty} \frac{\|x_{k+2} - x^*\|}{\|x_k - x^*\|} = 0.$$

4 Numerical Results

In our implementation of Algorithm **FSQP′** we allow for some classification of the constraints in order to exploit structure. In particular, the implementation contains special provisions for linear (affine) constraints and simple bounds on the variables. The general problem solved is

$$\begin{array}{lll} \min & f(x) & \\ \text{s.t.} & g_j(x) \le 0, & j = 1, \dots, m_n, \\ & \langle a_j, x\rangle + b_j \le 0, & j = 1, \dots, m_a, \\ & x^\ell \le x \le x^u, & \end{array}$$

where $a_j \in \mathbb{R}^n$, $b_j \in \mathbb{R}$, $j = 1, \dots, m_a$, and x^ℓ, $x^u \in \mathbb{R}^n$ with $x^\ell < x^u$ (componentwise). The linear constraints and bounds require no "tilting" and may be directly incorporated into $\widehat{QP}(x_k, H_k, \eta_k)$, i.e.

$$\begin{array}{lll} \min & \frac{1}{2}\langle \hat{d}, H_k \hat{d}\rangle + \hat{\gamma} & \\ \text{s.t.} & \langle \nabla f(x_k), \hat{d}\rangle \le \hat{\gamma}, & \\ & g_j(x) + \langle \nabla g_j(x), \hat{d}\rangle \le \hat{\gamma} \cdot \eta_k^j, & j = 1, \dots, m_n, \\ & \langle a_j, x_k + \hat{d}\rangle + b_j \le 0, & j = 1, \dots, m_a, \\ & x^\ell - x_k \le \hat{d} \le x^u - x_k. & \end{array}$$

Note that a distinct value of η_k is maintained for each nonlinear constraint, i.e η_k^j, $j = 1, \dots, m_n$. This helps significantly in practice while not affecting the analysis. We define the active sets in the implementation as

$$\hat{I}_k^n = \{\ j \mid g_j(x_k) + \langle \nabla g_j(x_k), \hat{d}_k\rangle - \hat{\gamma}_k \cdot \eta_k^j > -\sqrt{\epsilon_m}\ \}$$
$$\hat{I}_k^a = \{\ j \mid \langle a_j, x_k + \hat{d}_k\rangle + b_j > -\sqrt{\epsilon_m}\ \}$$

where ϵ_m is the machine precision. As before, let $\hat{\lambda}_k^j \in \mathbb{R}^{m_n}$ be the QP multipliers corresponding to the nonlinear constraints. Define $\hat{\lambda}_k^a \in \mathbb{R}^{m_a}$, $\zeta_k^u \in \mathbb{R}^n$, and $\zeta_k^l \in \mathbb{R}^n$ as the QP multipliers corresponding the the affine

constraints, the upper bounds, and the lower bounds respectively. The *binding* sets are defined as

$$\hat{I}_k^{b,n} = \{\ j \mid \hat{\lambda}_k^j > 0\ \}, \qquad \hat{I}_k^{b,a} = \{\ j \mid \hat{\lambda}_k^{a,j} > 0\ \},$$
$$\hat{I}_k^{b,l} = \{\ j \mid \zeta_k^{l,j} > 0\ \}, \qquad \hat{I}_k^{b,u} = \{\ j \mid \zeta_k^{u,j} > 0\ \}.$$

Of course, no bending is required from $\tilde{d}_k$ for affine constraints and simple bounds, hence if $\hat{I}_k^n = \emptyset$, we simply set $\tilde{d}_k = 0$, otherwise the implementation attempts to compute $\tilde{d}_k$ as the solution of

$$\begin{array}{lll}
\min & \langle \hat{d}_k + \tilde{d}, H_k(\hat{d}_k + \tilde{d})\rangle + \langle \nabla f(x_k), \hat{d}_k + \tilde{d}\rangle & \\
\text{s.t.} & g_j(x_k + \hat{d}_k) + \langle \nabla g_j(x_k), \tilde{d}\rangle = -\min\{10^{-2}\|\hat{d}_k\|, \|\hat{d}_k\|^\tau\}, & j \in \hat{I}_k^n, \\
& \langle a_j, x_k + \hat{d}_k + \tilde{d}\rangle + b_j = 0, & j \in \hat{I}_k^a, \\
& \tilde{d}^j = x^u - x_k^j - \hat{d}_k^j, & j \in \hat{I}_k^{b,u}, \\
& \tilde{d}^j = x^l - x_k^j - \hat{d}_k^j, & j \in \hat{I}_k^{b,l}.
\end{array}$$

Since not all simple bounds are included in the computation of $\tilde{d}_k$, it is possible that $x_k + \hat{d}_k + \tilde{d}_k$ will not satisfy all bounds. To take care of this, we simply "clip" $\tilde{d}_k$ so that the bounds are satisfied. Specifically, for the upper bounds, we perform the following:

for $j \notin \hat{I}_k^{b,u}$ **do**
  if $(\tilde{d}_k^j \geq x^u - x_k^j - \hat{d}_k^j)$ **then**
    $\tilde{d}_k^j \leftarrow x^u - x_k^j - \hat{d}_k^j$
end

The same procedure, mutatis mutandis, is executed for the lower bounds. We note that such a procedure has no affect on the convergence analysis of Section 3 since, locally, the active set is correctly identified and a full step along $\hat{d}_k + \tilde{d}_k$ is always accepted.

Due to convexity of affine constraints, in the line search of *Step 2* we first generate an upper bound on the step size $\bar{t}_k \leq 1$ using the affine constraints that were not used in the computation of $\tilde{d}_k$. Once these constraints are satisfied, they need not be checked again. Finally, the least squares problem used to compute $\widehat{d_k^0}$ is modified similarly. In the implementation, $\widehat{d_k^0}$ is only

computed if $m_n > 0$, in which case we use

$$\begin{array}{lll} \min & \frac{1}{2}\langle \widehat{d^0}, H_k \widehat{d^0}\rangle + \langle \nabla f(x_k), \widehat{d^0}\rangle & \\ \text{s.t.} & g_j(x_k) + \langle \nabla g_j(x_k), \widehat{d^0}\rangle = 0, & j \in \hat{I}^{b,n}_{k-1}, \\ & \langle a_j, x_k + \widehat{d^0}\rangle + b_j = 0, & j \in \hat{I}^{b,a}_{k-1}, \\ & \widehat{d^0}^j = x^u - x^j_k, & j \in \hat{I}^{b,u}_{k-1}, \\ & \widehat{d^0}^j = x^l - x^j_k, & j \in \hat{I}^{b,l}_{k-1}. \end{array}$$

It was mentioned above that, in the implementation, we maintain a separate tilting parameter η^j_k for each nonlinear constraint. In particular, the η^j_k's are different because we use a different scaling C^j_k for each nonlinear constraint. In the algorithm description and in the analysis all that was required of C_k was that it remain bounded and bounded away from zero. In practice, though, performance of the algorithm is critically dependent upon the choice of C_k. For our implementation, an adaptive scheme was chosen in which C^j_k is increased if $g_j(\cdot)$ caused a failure in the line search. Otherwise, if $f(\cdot)$ caused a failure in the line search, C_k is decreased. Specifically, our update rule is as follows,

if ($g_j(\cdot)$ caused line search failure) **then** $C^j_{k+1} \leftarrow C^j_k \cdot \delta_c$
else if ($f(\cdot)$ caused line search failure) **then** $C^j_{k+1} \leftarrow C^j_k / \delta_c$
if ($C^j_{k+1} < \underline{C}$) **then** $C^j_{k+1} \leftarrow \underline{C}$
if ($C^j_{k+1} > \overline{C}$) **then** $C^j_{k+1} \leftarrow \overline{C}$

where $\delta_c > 1$.

Another aspect of the algorithm which was purposefully left vague in Sections 2 and 3 was the updating scheme for the Hessian estimates H_k. In the implementation, we use the BFGS update with Powell's modification [16]. Specifically, define

$$\delta_{k+1} \triangleq x_{k+1} - x_k$$

$$\gamma_{k+1} \triangleq \nabla_x L(x_{k+1}, \hat{\lambda}_k) - \nabla_x L(x_k, \hat{\lambda}_k),$$

where, in an attempt to better approximate the true multipliers, if $\hat{\mu}_k > \sqrt{\epsilon_m}$ we normalize as follows

$$\hat{\lambda}^j_k \leftarrow \frac{\hat{\lambda}^j_k}{\hat{\mu}_k}, \quad j = 1, \ldots, m_n.$$

A scalar $\theta_{k+1} \in (0,1]$ is then defined by

$$\theta_{k+1} \triangleq \begin{cases} 1, & \text{if } \delta_{k+1}^T \gamma_{k+1} \geq 0.2 \cdot \delta_{k+1}^T H_k \delta_{k+1}, \\ \dfrac{0.8 \cdot \delta_{k+1}^T H_k \delta_{k+1}}{\delta_{k+1}^T H_k \delta_{k+1} - \delta_{k+1}^T \gamma_{k+1}}, & \text{otherwise.} \end{cases}$$

Defining $\xi_{k+1} \in \mathbb{R}^n$ as

$$\xi_{k+1} \triangleq \theta_{k+1} \cdot \gamma_{k+1} + (1 - \theta_{k+1}) \cdot H_k \delta_{k+1},$$

the rank two Hessian update is

$$H_{k+1} = H_k - \frac{H_k \delta_{k+1} \delta_{k+1}^T H_k}{\delta_{k+1}^T H_k \delta_{k+1}} + \frac{\xi_{k+1} \xi_{k+1}^T}{\delta_{k+1}^T \xi_{k+1}}.$$

Note that while it is not clear whether the resultant sequence $\{H_k\}$ will, in fact, satisfy Assumption 7, this update scheme is known to perform very well in practice.

Our implementation calls the Goldfarb-Idnani based active set QP solver QLD due to Powell and Schittkowski [19]. QLD uses dense linear algebra and does not allow "warm starts", i.e. does not allow the user to supply an initial guess for the QP multipliers. For simplicity, we not only used QLD to solve $\widehat{QP}(x_k, H_k, \eta_k)$, but also the least squares problems. Of course, this was likely not too inefficient since the active set is known automatically for these problems. In order to guarantee that the algorithm terminates after a finite number of iterations with an approximate solution, the stopping criterion of *Step 1* is changed to

$$\textbf{if} \;\; (\|\hat{d}_k\| \leq \epsilon) \;\; \textbf{stop},$$

where $\epsilon > 0$ is small. Finally, note that during the line search of *Step 2*, as soon as it is determined that the given trial point does not satisfy the descent criterion or a particular constraint, no more constraints are evaluated. In this case, a new trial point is immediately computed and the trial evaluations start over from the beginning. In order to reduce the number of constraint function evaluations, the constraint which caused the failure is always checked first at the new trial point, as it is most likely to be infeasible.

In order to test the implementation, we selected several problems from [7] which provided feasible initial points and contained no equality constraints.

The results are reported in Table 1. For all problems we used the parameter values

$$\begin{array}{lll} \alpha = 0.1, & \beta = 0.5, & \tau = 2.5, \\ \epsilon_\ell = \min\{1, \sqrt{\epsilon}\}, & \underline{C} = 1 \times 10^{-3}, & \overline{C} = 1 \times 10^{3}, \\ \delta_c = 10, & \bar{D} = 10 \cdot \epsilon_\ell. & \end{array}$$

Further, we always set $H_0 = I$ and $\eta_0^j = 1 \times 10^{-2}$, $C_0^j = 1$, $j = 1, \ldots, m_n$.

In Table 1 we compare our implementation with CFSQP [10], the implementation of Algorithm **FSQP** as described in [12]. The column labeled # lists the problem number as given in [7], the column labeled `ALGO` tells which algorithm was used to solve the given problem (the names are self-explanatory). The next three columns give the size of the problem following the conventions of this section. The columns labeled `NF`, `NG`, and `IT` give the number of objective function evaluations, nonlinear constraint function evaluations, and iterations required to solve the problem, respectively. Finally, $f(x^*)$ is the objective function value at the final iterate and ϵ is the tolerance for the size of the search direction (the stopping criterion). The value of ϵ was chosen in order to obtain approximately the same precision as reported in [7] for each problem.

The results reported in Table 1 are very encouraging. The performance of our implementation of Algorithm **FSQP**$'$ is essentially identical to that of CFSQP (Algorithm **FSQP**). Of course, Algorithm **FSQP**$'$ requires substantially less work per iteration than Algorithm **FSQP**, thus in the case that the work to generate a new iterate dominates the work to evaluate the objectives and constraints, the new algorithm is at a clear advantage.

5 Conclusions

We have presented here a new SQP-type algorithm generating feasible iterates. The main advantage of the algorithm presented here is a dramatic reduction in the amount of computation required in order to generate a new iterate. While this may not be very important for applications where function evaluations dominate the actual amount of work to compute a new iterate, it is very useful in many contexts. In any case, we saw in the previous section that preliminary results seem to indicate that decreasing the amount of computation per iteration did not come at the cost of increasing the number of function evaluations required to find a solution.

#	ALGO	n	m_a	m_n	NF	NG	IT	$f(x^*)$	ϵ
12	NEW	2	0	1	7	14	7	-3.0000000E+01	1.E–6
	CFSQP				7	14	7	-3.0000000E+01	
29	NEW	3	0	1	11	20	10	-2.2627417E+01	1.E–5
	CFSQP				11	20	10	-2.2627417E+01	
30	NEW	3	0	1	18	35	18	1.0000000E+00	1.E–7
	CFSQP				18	35	18	1.0000000E+00	
31	NEW	3	0	1	9	25	8	6.0000000E+00	1.E–5
	CFSQP				9	19	7	6.0000000E+00	
33	NEW	3	0	2	4	11	4	-4.0000000E+00	1.E–8
	CFSQP				4	11	4	-4.0000000E+00	
34	NEW	3	0	2	8	32	8	-8.3403245E–01	1.E–8
	CFSQP				7	28	7	-8.3403244E–01	
43	NEW	4	0	3	9	45	8	-4.4000000E+01	1.E–5
	CFSQP				10	46	8	-4.4000000E+01	
66	NEW	3	0	2	8	30	8	5.1816327E–01	1.E–8
	CFSQP				8	30	8	5.1816327E–01	
84	NEW	5	0	6	4	32	4	-5.2803351E+06	1.E–8
	CFSQP				4	30	4	-5.2803351E+06	
93	NEW	6	0	2	14	55	12	1.3507596E+02	1.E–5
	CFSQP				16	62	13	1.3507596E+02	
113	NEW	10	3	5	13	116	13	2.4306210E+01	1.E–3
	CFSQP				12	108	12	2.4306377E+01	
117	NEW	15	0	5	19	179	17	3.2348679E+01	1.E–4
	CFSQP				20	219	19	3.2348679E+01	

Table 1: Numerical results.

A number of significant extensions of Algorithm **FSQP′** are being examined. It is not too difficult to extend the algorithm to handle mini-max problems. The only real issue that arises is how to handle the mini-max objectives in the least squares sub-problems. Several possibilities, each with the desired global and local convergence properties, are being examined. Another extension that is important for engineering design is the incorporation of a scheme to efficiently handle very large sets of constraints and/or objectives. We will examine schemes along the lines of those developed in [9, 22]. Further, work remains to be done to exploit the close relationship between the two least squares problems and the quadratic program. A careful implementation should be able to use these relationships to great advantage

computationally. For starters, updating the Cholesky factors of H_k instead of H_k itself at each iteration would save a factorization in each of the subproblems. Finally, it is possible to extend the class of problems (P) which are handled by the algorithm to include nonlinear equality constraints. Of course, we will not be able to generate feasible iterates for such constraints, but a scheme such as that studied in [8] could be used in order to guarantee asymptotic feasibility while maintaining feasibility for all inequality constraints.

Appendix

In this appendix we discuss how the arguments given by Powell in Sections 2 and 3 of [15] may be used, with some modification, to prove Theorem 2. To avoid confusion, we will refer to Lemmas from [15] as Lemma P.n, where n is the number as it appears in [15]. We begin by noting that all of Powell's assumptions outlined at the beginning of Section 2 in [15] hold in our case (under the strengthened assumptions of Section 3.2). Also, Lemmas P.1 and P.2 are already established by our Lemmas 14 and 16. These Lemmas show that the active set is exactly identified by the QP multipliers for all k sufficiently large. In view of this, and since Lemma 20 shows that $t_k = 1$ for all k sufficiently large, the inactive constraints eventually have no effect on the computation of a new iterate. Thus, without loss of generality, it may be assumed here that we are generating iterates converging to a solution of the problem

$$\begin{array}{lll} \min & f(x) & \\ \text{s.t.} & g_j(x) = 0, & j \in I(x^*), \end{array} \qquad (P^+)$$

Let $L^+ : \mathbb{R}^n \times \mathbb{R}^{|I(x^*)|} \to \mathbb{R}$ be the corresponding Lagrangian function and, recalling our notation introduced in Section 3.2, let λ^{*+} be the optimal multiplier for (P^+).

Lemma P.3, which establishes that the SQP direction d_k^0 is unchanged when the matrix H_k is perturbed by a symmetric matrix whose kernel includes the orthogonal complement of the constraint gradients, is algorithm independent, hence automatically holds. Following Powell's notation, define

$$h_k \triangleq P_k \nabla f(x_k),$$

and interpret the symbol "$\sim$" as meaning the ratio of the expression on the left-hand side to the right-hand side is both bounded above and bounded

away from zero, as $k \to \infty$. Using the same argument as in Lemma P.4, we can show (recall the definition of g_k from Section 3.2)

$$\|d_k^0\| \sim \|g_k\| + \|h_k\|.$$

In view of (22), this implies Lemma P.4 still holds in our case.

Unfortunately, the proof of Lemma P.5 will not work in our context. Thus, we establish this result here.

Lemma 21. $\|x_k - x^*\| \sim \|g_k\| + \|h_k\|$.

Proof. We begin by showing that $\nabla^2 L^+(x^*, \lambda^{*+})$ (by which we mean the second derivative with respect to both x and λ) is non-singular. Let $R^* \triangleq \lim_{k\to\infty} R_k$. Suppose there exists $z = (y^T, u^T)^T \in \mathbb{R}^{n+|I(x^*)|}$ such that $\nabla^2 L^+(x^*, \lambda^*)z = 0$. Then, using complementary slackness we can substitute $\nabla^2_{xx} L(x^*, \lambda^*)$ for $\nabla^2_{xx} L^+(x^*, \lambda^{*+})$, obtaining

$$\begin{bmatrix} \nabla^2_{xx} L(x^*, \lambda^*) & R^* \\ R^{*T} & 0 \end{bmatrix} \begin{pmatrix} y \\ u \end{pmatrix} = 0.$$

So, $R^{*T} y = 0$ and $y^T \nabla^2_{xx} L(x^*, \lambda^*) y = -(R^{*T} y)^T u = 0$, which, in view of Assumption 5, implies $y = 0$. This, in turn, implies $R^* u = 0$, which, by Assumption 3 requires $u = 0$. Thus, we have shown that $\nabla^2 L^+(x^*, \lambda^{*+})$ is non-singular.

Note that we may write

$$\begin{aligned} &\nabla L^+(x_k, \lambda_k^{0+}) \\ &\quad = \int_0^1 \nabla^2 L^+(x^* + t(x_k - x^*), \lambda^{*+} + t(\lambda_k^{0+} - \lambda^{*+})) \begin{pmatrix} x_k - x^* \\ \lambda_k^{0+} - \lambda^{*+} \end{pmatrix} dt \\ &\quad \triangleq \overline{D}_k \begin{pmatrix} x_k - x^* \\ \lambda_k^{0+} - \lambda^{*+} \end{pmatrix}. \end{aligned}$$

Since $x_k \to x^*$ and $\lambda_k^{0+} \to \lambda^{*+}$, it follows from our regularity Assumption 2′ that $\overline{D}_k \to \nabla^2 L^+(x^*, \lambda^{*+})$. Non-singularity of $\nabla^2 L^+(x^*, \lambda^{*+})$ implies that for all k sufficiently large, $\overline{D}_k$ is non-singular and there exists $\overline{M} > 0$ such that

$$\|\overline{D}_k^{-1}\| \le \overline{M},$$

for k large enough. Thus,

$$\begin{aligned} \|x_k - x^*\| &\le \left(\|x_k - x^*\|^2 + \|\lambda_k^{0+} - \lambda^{*+}\|^2\right)^{\frac{1}{2}} \\ &= \left\|\overline{D}_k^{-1} \nabla L^+(x_k, \lambda_k^{0+})\right\|, \end{aligned}$$

(where we are using the Euclidean norm) which implies

$$\|x_k - x^*\| \leq \overline{M} \left\| \nabla L^+(x_k, \lambda_k^{0+}) \right\|, \tag{23}$$

for all k sufficiently large (note that we are using the Euclidean norm). Recall now that, for k large enough, the SQP direction satisfies

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^0 \\ \lambda_k^{0+} \end{pmatrix} = - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix}.$$

This can be solved for d_k^0, yielding

$$\begin{aligned} d_k^0 &= -H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} g_k \\ &\quad - \left[H_k^{-1} - H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} R_k^T H_k^{-1} \right] \nabla f(x_k) \\ &= -H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} g_k \\ &\quad - \left[H_k^{-1} - H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} R_k^T H_k^{-1} \right] \\ &\qquad \cdot (P_k + R_k (R_k^T R_k)^{-1} R_k^T) \nabla f(x_k) \\ &= -H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} g_k \\ &\quad - \left[H_k^{-1} - H_k^{-1} R_k (R_k^T H_k^{-1} R_k)^{-1} R_k^T H_k^{-1} \right] P_k \nabla f(x_k) \\ &\triangleq B_k g_k + E_k h_k, \end{aligned}$$

where B_k and E_k are bounded for large k, and we have used the trivial identity $P_k + R_k (R_k^T R_k)^{-1} R_k^T = I$. Now, in view of the optimality conditions (4),

$$\begin{aligned} \nabla_x L^+(x_k, \lambda_k^{0+}) &= -H_k d_k^0 \\ &= -H_k B_k g_k - H_k E_k h_k. \end{aligned}$$

Thus, there exists $K_1,\ K_2 > 0$ such that for large k

$$\|\nabla_x L^+(x_k, \lambda_k^{0+})\| \leq K_1 \|g_k\| + K_2 \|h_k\|. \tag{24}$$

Finally, since $\nabla_\lambda L^+(x_k, \lambda_k^{0+}) = g_k$, we conclude from (23) and (24) that there exists $K_3 > 0$ such that for large k

$$\|x_k - x^*\| \leq K_3 \cdot (\|g_k\| + \|h_k\|).$$

To go the other direction, expanding $g(\cdot)$ about x^* (recall that for this argument $g : \mathbb{R}^n \to \mathbb{R}^{|I(x^*)|}$) and noting that $P_k \nabla g_j(x_k) = 0$ for all k, we

have

$$\begin{aligned}
\|g_k\| + \|h_k\| &= \|g(x^*) + R_k^T(x_k - x^*) + O(\|x_k - x^*\|^2)\| \\
&\quad + \|P_k \nabla_x L^+(x_k, \lambda^{*+})\| \\
&= \|R_k^T(x_k - x^*) + O(\|x_k - x^*\|^2)\| \\
&\quad + \|P_k(\nabla_x L^+(x^*, \lambda^{*+}) + \nabla_{xx}^2 L^+(x^*, \lambda^{*+})(x_k - x^*) \\
&\quad + O(\|x_k - x^*\|^2)\| \\
&= \|R_k^T(x_k - x^*)\| + \|P_k \nabla_{xx}^2 L^+(x^*, \lambda^{*+})(x_k - x^*)\| \\
&\quad + O(\|x_k - x^*\|^2) \\
&\leq K_4\|x_k - x^*\| + O(\|x_k - x^*\|^2),
\end{aligned}$$

for some constant $K_4 > 0$, and the result follows. □

Lemma P.6 requires some additional explanation in our case. In particular, we need to justify/modify equations (3.3), (3.8), and (3.9) in [15]. To begin with, consider for all k sufficiently large (and recall that we are only interested in $j \in I(x^*)$ here)

$$\begin{aligned}
g_j(x_{k+1}) &= g_j(x_k + d_k^0 + O(\|d_k^0\|^2) \\
&= g_j(x_k) + \langle \nabla g_j(x_k), d_k^0 \rangle + O(\|d_k^0\|^2) \\
&= O(\|x_{k+1} - x_k\|^2).
\end{aligned}$$

Thus equation (3.3) holds. If $O(\|x_{k+1} - x_k\|^2)$ is added to the right hand side of equation (3.8), and to both sides of equation (3.9), then the same argument holds for the sequences generated by Algorithm **FSQP′**. Finally, Theorem P.1 is the same as our Theorem 2 and the argument used in [15] may be used to prove Theorem 2.

References

[1] J. Birge, L. Qi, and Z. Wei. A variant of the Topkis-Veinott method for solving inequality constrained optimization problems. Technical Report AMR 97/29, School of Mathematics, The University of New South Wales, Sydney, Australia, 1997. `http://www.maths.unsw.edu.au/appweb/preprints/amr97_29.ps`.

[2] P. T. Boggs and J. W. Tolle. Sequential quadratic programming. *Acta Numerica*, pages 1–51, 1995.

[3] R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. Technical Report 97-05, Optimization Technology Center, Argonne National Laboratory, 1997. `http://www.eecs.nwu.edu/ nocedal/psfiles/nitro.ps`.

[4] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang. On the formulation and theory of the Newton interior-point method for nonlinear programming. *J. Opt. Theory Appl.*, 89:507–541, 1996.

[5] D. M. Gay, M. L. Overton, and M. H. Wright. A primal-dual interior method for nonconvex nonlinear programming. Technical Report 97-4-08, Bell Laboratories, Computing Sciences Research Center, 1997. `http://cm.bell-labs.com/cm/cs/doc/97/4-08.ps.gz`.

[6] J. N. Herskovits and L. A. V. Carvalho. A successive quadratic programming based feasible directions algorithm. In A. Bensoussan and J.L. Lions, editors, *Proc. of the Seventh International Conference on Analysis and Optimization of Systems — Antibes, June 25-27, 1986, Lecture Notes in Control and Information Sciences 83*, pages 93–101. Springer Verlag, Berlin-Heidelberg-New York-Tokyo, 1986.

[7] W. Hock and K. Schittkowski. *Test Examples For Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems No. 187*. Springer-Verlag, Berlin, 1981.

[8] C. T. Lawrence and A. L. Tits. Nonlinear equality constraints in feasible sequential quadratic programming. *Optimization Methods and Software*, 6:265–282, 1996.

[9] C. T. Lawrence and A. L. Tits. Feasible sequential quadratic programming for finely discretized problems from SIP. In R. Reemtsen and J.-J. Rückmann, editors, *Semi-infinite Programming*, pages 159–193. Kluwer Academic Publishers B.V., 1998. In the series Nonconvex Optimization and its Applications.

[10] C. T. Lawrence, J. L. Zhou, and A. L. Tits. *User's Guide for CFSQP Version 2.5: A C Code for Solving (Large Scale) Constrained Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality Constraints*, 1998. ISR TR-94-16r1, Institute for Systems Research, University of Maryland (College Park, MD) `http://www.isr.umd.edu/Labs/CACSE/FSQP/c_manual.ps`.

[11] E. R. Panier and A. L. Tits. A superlinearly convergent feasible method for the solution of inequality constrained optimization problems. *SIAM Journal on Control and Optimization*, 25(4):934–950, 1987.

[12] E. R. Panier and A. L. Tits. On combining feasibility, descent and superlinear convergence in inequality constrained optimization. *Math. Programming*, 59:261–276, 1993.

[13] E. R. Panier, A. L. Tits, and J. N. Herskovits. A QP-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization. *SIAM J. Control and Optimization*, 26(4):788–811, 1988.

[14] E. Polak. *Computational Methods in Optimization.* Academic Press, New York, 1971.

[15] M. J. D. Powell. Convergence of variable metric methods for nonlinearly constrained optimization calculations. In O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, editors, *Nonlinear Programming 3*, pages 27–63. Academic Press, New York, 1978.

[16] M. J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In G. A. Watson, editor, *Numerical Analysis, Dundee, 1977, Lecture Notes in Mathematics 630*, pages 144–157. Springer Verlag, 1978.

[17] L. Qi and Z. Wei. Constant positive linear independence, KKT points, and convergence of feasible SQP methods. Technical Report AMR 97/4, School of Mathematics, The University of New South Wales, Sydney, Australia, 1997. `http://www.maths.unsw.edu.au/appweb/preprints/amr97_4.ps`.

[18] S. M. Robinson. Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear-programming algorithms. *Math. Programming*, 7:1–16, 1974.

[19] K. Schittkowski. *QLD: A Fortran Code for Quadratic Programming, User's Guide.* Mathematisches Institut, Universität Bayreuth, Germany, 1986.

[20] T. Urban, A. L. Tits, and C. T. Lawrence. A primal-dual interior-point method for nonconvex optimization with multiple logarithmic barrier parameters and with strong convergence properties. Technical Report

98-27, Institute for Systems Research, University of Maryland, College Park, 1998. http://www.isr.umd.edu/ andre/pdip.ps.

[21] R. J. Vanderbei and D. F. Shanno. An interior point algorithm for nonconvex nonlinear programming. Technical Report SOR-97-21, Princeton University, Dept. of Statistics and Operations Research, 1997. http://www.sor.princeton.edu/ rvdb/ps/nonlin.ps.gz.

[22] J. L. Zhou and A. L. Tits. An SQP algorithm for finely discretized continuous minimax problems and other minimax problems with many objective functions. *SIAM J. on Optimization*, pages 461–487, May 1996.

[23] J. L. Zhou, A. L. Tits, and C. T. Lawrence. *User's Guide for FSQP Version 3.7: A FORTRAN Code for Solving Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality and Linear Constraints*, 1997. ISR TR-92-107r2, Institute for Systems Research, University of Maryland (College Park, MD) http://www.isr.umd.edu/Labs/CACSE/FSQP/f_manual.ps.

[24] G. Zoutendijk. *Methods of Feasible Directions.* Elsevier Science, Amsterdam, The Netherlands, 1960.