AN INEXACT FIRST-ORDER METHOD FOR CONSTRAINED NONLINEAR OPTIMIZATION

HAO WANG*, JIASHAN WANG†, YUYANG RONG‡, AND HUDIE ZHOU§

Abstract. The primary focus of this paper is on designing inexact first-order methods for solving large-scale constrained nonlinear optimization problems. By controlling the inexactness of the subproblem solution, we can significantly reduce the computational cost needed for each iteration. A penalty parameter updating strategy during the subproblem solve enables the algorithm to automatically detect infeasibility. Global convergence for both feasible and infeasible cases are proved. Complexity analysis for the KKT residual is also derived under loose assumptions. Numerical experiments exhibit the ability of the proposed algorithm to rapidly find inexact optimal solution through cheap computational cost.

Key words. nonlinear optimization, sequential linear optimization, large-scale constrained problems, exact penalty functions, convex composite optimization, first-order methods

AMS subject classifications. 49M20, 49M29, 49M37, 65K05, 65K10, 90C06, 90C20, 90C25

1. Introduction.

In the last few years, a number of advances on first-order optimization methods have been made for unconstrained/constrained optimization problems in a wide range of applications including machine learning, compressed sensing and signal processing. This is largely due to their relatively low iteration computational cost, as well as their implementation easiness. Numerous works have emerged for solving unconstrained optimization problems, e.g. the stochastic gradient descent methods [5, 6, 7, 19] and mirror descent methods [3, 20] solving machine learning problems, soft-thresholding type algorithms [4, 15] for solving sparse reconstruction problems. For certain structured constrained optimization problems, many first-order methods have also captured researchers' attention, such as conditional gradient methods (also known as Frank-Wolfe methods) for solving principle component analysis problems [24], gradient projection methods for solving various problems with structured constraints [16, 27], and gradient methods on Riemannian manifolds [1, 26].

On the contrary, little attention has been paid on first-order methods for solving general constrained optimization problems in the past decades. This is mainly due to the slow tail convergence of first-order methods, since it can cause heavy computational burden for obtaining accurate solutions. Most of the research efforts can date back to the successive linear programming (SLP) algorithms [2, 23] designed in 1960s-1980s for solving pooling problems in oil refinery. Among various SLP algorithms, the most famous SLP algorithm is proposed by Fletcher and Maza in [17], which analyzes the global convergence as well as the local convergence under strict complementarity, second-order sufficiency and regularity conditions. The other well-known work is the active-set algorithmic option implemented in the off-the-shelf solver `Knitro` [12], which sequentially solves a linear optimization subproblem and an equality constrained quadratic optimization subproblem.

The primary focus of this paper is to design an algorithmic framework of first-order methods for nonlinear constrained optimization. Despite of their weakness on tail convergence, first-order methods are widely used for quickly computing relatively

*Sch. of Inf. Sci. and Tech., ShanghaiTech University; wanghao1@shanghaitech.edu.cn
†Dept. of Mathematics, University of Washington; jsw1119@gmail.com
‡Sch. of Inf. Sci. and Tech., ShanghaiTech University; rongyy@shanghaitech.edu.cn
§Sch. of Phy. Sci. and Tech., ShanghaiTech University; zhouhd@shanghaitech.edu.cn.

inaccurate solutions. Unlike second-order methods, the subproblems in first-order methods are often easier to tackle. If only inexact subproblem solutions are required to enforce the global convergence, the computational cost per iteration could further be reduced, which may be able to compensate for the inefficiency of the entire algorithm. To achieve this, one must carefully handle the possible infeasibility of the subproblem constraints. Many nonlinear solvers need sway away from the main algorithm and turn to a so-called *feasibility restoration phase* to improve the constraint satisfaction. In a penalty method, the penalty parameter needs to be properly tuned so that the feasibility of the nonlinear problem is not deteriorated, such penalty parameter updating strategy have been studies in [9, 10, 11, 13] for sequential quadratic programming methods.

1.1. Contributions. The major contribution of this paper is an algorithmic framework of inexact first-order penalty methods. The novelties of the proposed methods include three aspects. First, only inexact solve of each subproblem is needed, which can significantly reduce computational effort for each subproblem. Indeed, if the subproblem is a linear optimization problem, then only a few pivots by simplex methods can be witnessed, making the fast computation of relatively inaccurate solutions possible. The second novelty of our proposed methods is the ability of automatic detection of potential constraint inconsistencies, so that the algorithm can automatically solve for optimality if the problem is feasible or find an infeasible stationary point if the problem is (locally) infeasible. The last novel feature of our work is the worst-case complexity analysis for the proposed algorithm under loose assumptions. We show that the KKT residuals for optimality problem and feasibility condition need at most $O(1/\epsilon^2)$ iterations to reach below ϵ and for feasible cases the constraint violation locally needs at most $O(1/\epsilon^2)$ iterations—a novelty rarely seen in general nonlinear constrained optimization methods.

1.2. Organization. In § 2, we describe the proposed framework of inexact first-order penalty method. The global convergence and worst-case complexity analysis of the proposed methods are analyzed in § 3. Subproblems algorithms are discussed in § 4. Implementations of the proposed methods and the numerical results are discussed in § 5. Finally, concluding remarks are provided in § 6.

2. A Framework of Inexact First-Order Methods. In this section, we formulate our problem of interest and outline a framework of inexact first-order penalty method. We present our algorithm in the context of the generic nonlinear optimization problem with equality and inequality constraints

$$\text{(NLP)} \qquad \begin{aligned} \min_{x\in\mathbb{R}^n}\ & f(x) \\ \text{s.t. } & c_i(x) = 0 \quad \text{for all} \quad i \in \mathcal{E}; \\ & c_i(x) \le 0 \quad \text{for all} \quad i \in \mathcal{I}. \end{aligned}$$

where the functions $f : \mathbb{R}^n \to \mathbb{R}$ and $c_i : \mathbb{R}^n \to \mathbb{R}$ for $i \in \mathcal{E} \cup \mathcal{I}$ are continuously differentiable.

We define utility functions to characterize first-order stationary solutions in our penalty method. Our algorithms either converge to a feasible stationary point of (NLP) or an infeasible stationary point. In both cases, the algorithms will converge to a stationary point of *feasibility problem*

$$\text{(FP)} \qquad \min_{x\in\mathbb{R}^n}\ v(x)$$

where v is the constraint violation defined by

$$v(x) = \sum_{i \in \mathcal{E} \cup \mathcal{I}} v_i(c_i(x)) \tag{2.1}$$

with

$$v_i(z) = |z|\,,\ i \in \mathcal{E} \quad \text{and} \quad v_i(z) = (z)_+,\ i \in \mathcal{I}.$$

If this minimizer of (2.1) violates the constraints of (NLP), then it provides a certificate of infeasibility for (NLP). It can be shown that the Clarke's generalized gradients [14] of v, denoted by $\bar{\partial} v(x)$, is given by

$$\bar{\partial} v(x) = \{ \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \nabla c_i(x) \mid \lambda_i \in \partial_c v_i(c_i(x))\},$$

where

$$\partial_c v_i(c_i) = \begin{cases} [-1,1] & \text{if } i \in \mathcal{E} \quad \text{and } c_i = 0 \\ [0,1] & \text{if } i \in \mathcal{I} \quad \text{and } c_i = 0 \\ \{-1\} & \text{if } i \in \mathcal{E} \quad \text{and } c_i < 0 \\ \{0\} & \text{if } i \in \mathcal{I} \quad \text{and } c_i < 0 \\ \{1\} & \text{if } i \in \mathcal{E} \cup \mathcal{I} \quad \text{and } c_i > 0. \end{cases} \tag{2.2}$$

A stationary point x for (FP) must satisfy

$$0 \in \bar{\partial} v(x), \tag{2.3}$$

and it is called an infeasible stationary point if $v(x) > 0$. Despite the possibility that problem (NLP) may be feasible elsewhere, it is deemed that no further progress on minimizing constraint violation locally can be made.

If the algorithm converges to a feasible point of problem (NLP), this point will be a stationary point of the ℓ_1 exact penalty function

$$\phi(x;\rho) := \rho f(x) + v(x), \tag{2.4}$$

with $v(x) = \phi(x;0) = 0$ for final penalty parameter $\rho > 0$. The Clarke's generalized gradients of ϕ, is then given by $\bar{\partial}\phi(x;\rho) = \rho \nabla f(x) + \bar{\partial} v(x)$, and a stationary point of $\phi(x;\rho)$ is characterized by

$$0 \in \rho \nabla f(x) + \bar{\partial} v(x).$$

Such a stationary point must satisfy the the first-order optimality condition

$$\rho \nabla f(x) + \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \nabla c_i(x) = 0, \tag{2.5a}$$

$$\lambda_i \in [-1,1], i \in \mathcal{E};\ \lambda_i \in [0,1], i \in \mathcal{I}, \tag{2.5b}$$

$$\sum_{c_i > 0} (1 - \lambda_i) v_i(c_i(x)) + \sum_{i \in \mathcal{E}, c_i < 0} (1 + \lambda_i) v_i(c_i(x)) + \sum_{i \in \mathcal{I}, c_i < 0} \lambda_i |c_i(x)| = 0. \tag{2.5c}$$

A first-order stationary point x for (NLP) thus can be presented as a stationary point of the penalty function with $\rho > 0$ and satisfying $v(x) = 0$. Let $\lambda = [\lambda_{\mathcal{E}}^T, \lambda_{\mathcal{I}}^T]^T$ be the multipliers which make the first two summations in (2.5c) vanish. Such a point (x, λ) is called stationary for (NLP) since it corresponds to a Karush-Kuhn-Tucker (KKT) point $(x, \lambda/\rho)$ for (NLP) [21, 22]. Also notice that (2.5) with $\rho = 0$ can be deemed as an equivalent statement of condition (2.3).

2.1. Subproblems. We now describe our technique for search direction computation which involves the inexact solution of subproblems that are constructed using merely the first-order information of (NLP). Once the details and properties of these subproblems have been specified, we will describe the penalty parameter ρ update strategy while iteratively solving the subproblems.

At the kth iteration, the algorithm seeks to measure the possible improvement in minimizing the linearized model of the penalty function $\phi(x;\rho)$ at x^k. The linear model $l(d;\rho,x^k)$ of penalty function at x^k is defined as

$$l(d;\rho,x^k) := \rho\langle\nabla f(x^k),d\rangle + \sum_{i\in\mathcal{E}\cup\mathcal{I}} v_i(c_i(x^k)+\langle\nabla c_i(x^k),d\rangle), \tag{2.6}$$

where the constant $\rho f(x^k)$ is omitted for the ease of presentation. The local model $l(\cdot;\rho,x^k)$ is convex and the subdifferential of $l(\cdot;\rho,x^k)$ at d is given by

$$\partial l(d;\rho,x^k) = \rho\nabla f(x^k) + \sum_{i\in\mathcal{E}\cup\mathcal{I}} \nabla c_i(x^k)\partial_c v_i(c_i(x^k)+\langle\nabla c_i(x^k),d\rangle).$$

In particular, its subdifferential at $d=0$ coincides with the Clarke's generalized subdifferential of $\phi(\cdot;\rho)$ at $x=x^k$

$$\partial l(0;\rho,x^k) = \bar{\partial}\phi(x^k;\rho) \quad \text{and} \quad \partial l(0;0,x^k) = \bar{\partial}v(x^k). \tag{2.7}$$

The subproblem solver aims to find a direction d^k that yields a nonnegative reduction in $l(\cdot;\rho,x^k)$ and $l(\cdot;0,x^k)$, i.e.,

$$\begin{aligned}\Delta l(d^k;\rho,x^k) &:= l(0;\rho,x^k) - l(d^k;\rho,x^k) \geq 0 \quad \text{and}\\ \Delta l(d^k;0,x^k) &:= l(0;0,x^k) - l(d^k;0,x^k) \geq 0.\end{aligned}$$

The idea underlying this goal is that if one cannot achieve any further improvement in finding a nonzero d to reduce $l(\cdot;\rho,x^k)$ and $l(\cdot;0,x^k)$, then x^k is a stationary point for $\phi(x^k;\rho)$ and $v(x^k)$ by (2.7). Such a direction d^k can be found as an *approximate* minimizer of $l(\cdot;\rho_k,x^k)$ for appropriate $\rho_k \in (0,\rho_{k-1}]$ over a convex set $X\subseteq\mathbb{R}^n$ containing $\{0\}$, i.e.,

$$(\mathscr{P}) \qquad d^k := \arg\min_{d\in X}\ l(d;\rho_k,x^k) \ \text{ for some } \ \rho_k\in(0,\rho_{k-1}].$$

To prevent infinite steps, we introduce the set X for imposing a trust region, which is defined as $X := \{d : \|d\|\leq\delta\}$ for trust region radius $\delta>0$ and $\|\cdot\|$ such that

$$\|x\|_2 \leq \kappa_0\|x\|, \quad \forall x\in\mathbb{R}^n \tag{2.8}$$

and constant $\kappa_0>0$. The most popular choice for the norm here is the ℓ_2-norm and ℓ_∞-norm. The latter one results in a linear optimization subproblem, but it is not a requirement in our algorithm. We refer to $(\mathscr{P})$ with $\rho>0$ as a *penalty subproblem* and $(\mathscr{P})$ with $\rho=0$ as the *feasibility subproblem*. In the remainder of this paper, let $d^*(\rho,x)$ denote a minimizer of $l(d;\rho,x)$.

To alleviate the computational burden of solving subproblems $(\mathscr{P})$ exactly, our algorithm accepts an inexact solution d^k of $(\mathscr{P})$ as long as it yields sufficient reduction in l compared with the optimal solution of $(\mathscr{P})$. In particular, it is required that the model reductions in $l(d;\rho,x^k)$ and $l(d;0,x^k)$ induced by d^k is at least a portion of reduction of the optimal $d^*(\rho,x^k)$ and $d^*(0,x^k)$ respectively:

$$\Delta l(d^k;\rho,x^k) + \gamma_k \geq \beta_\phi[\Delta l(d^*(\rho,x^k);\rho,x^k)+\gamma_k], \tag{2.9a}$$

$$\Delta l(d^k; 0, x^k) + \gamma_k \geq \beta_v[\Delta l(d^*(\rho, x^k); 0, x^k) + \gamma_k] \tag{2.9b}$$

where $\beta_\phi, \beta_v \in (0,1)$ with $\beta_v < \beta_\phi$ are prescribed constants and $\gamma_k \in \mathbb{R}_+$ is the relaxation error. Further details of γ_k will be discussed in § 2.2.

It is often impractical to verify conditions (2.9a) and (2.9b), since it requires the exact optimal solution of subproblems. Instead, we will use a relaxed version to enforce (2.9a) and (2.9b) are satisfied. The Lagrangian dual of ($\mathscr{P}$) is

$$(\mathscr{D}) \qquad \max_{\lambda_\mathcal{E}\in\mathbb{B}, \lambda_\mathcal{I}\in\mathbb{B}_+} \; p(\lambda; \rho, x^k) := -\delta\|\rho\nabla f(x^k) + \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i \nabla c_i(x^k)\|_* + \langle c(x^k), \lambda\rangle,$$

with $\lambda = [\lambda_\mathcal{E}^T, \lambda_\mathcal{I}^T]^T$ and $c(x) = [c_\mathcal{E}(x)^T, c_\mathcal{I}(x)^T]^T$. If λ is dual feasible, then by weak duality, we have for any primal-dual feasible pair (d, λ) that

$$p(\lambda; 0, x_k) \leq l(d; 0, x_k) \;\text{ and }\; p(\lambda; \rho, x_k) \leq l(d; \rho, x_k). \tag{2.10}$$

Using the dual values, we can require the direction d^k to satisfy

$$\Delta l(d^k; \rho, x^k) + \gamma_k \geq \beta_\phi[l(0; \rho, x^k) - p(\lambda^k; \rho, x^k) + \gamma_k] \tag{2.11a}$$

$$\Delta l(d^k; 0, x^k) + \gamma_k \geq \beta_v[l(0; 0, x^k) - p(\lambda^k; 0, x^k) + \gamma_k], \tag{2.11b}$$

for current dual feasible estimate λ^k, so that (2.9a) and (2.9b) naturally hold by weak duality.

An interesting aspect to notice is that $p(\lambda; \rho, x^k)$ consists of two parts. The first term is in fact the KKT optimality residual at (x^k, λ) scaled by the trust region radius δ, while for $\lambda_i \in \partial_c v_i(c_i(x^k))$, the second term $\langle c(x^k), \lambda\rangle = v(x^k)$ describes the complementarity, so that the problem ($\mathscr{D}$) is indeed seeking to minimize the KKT residual at x^k.

Before proceeding to the design of penalty parameter updating strategy, we first provide a couple of results related to our subproblems and their solutions.

LEMMA 1. *The following hold at any x with $\delta > 0$.*

(i) $\Delta l(d^*(0, x); 0, x) \geq 0$ *where the equality holds if and only if x is stationary for $v(\cdot)$.*

(ii) $\Delta l(d^*(\rho, x); \rho, x) \geq 0$ *where the equality holds if and only if x is stationary for $\phi(\cdot; \rho)$.*

(iii) If $\Delta l(d^*(\rho, x); \rho, x) = 0$ *for $\rho > 0$ and $v(x) = 0$, then x is a KKT point for* (NLP).

Proof. Note that $\Delta l(d^*(0, x); 0, x) = v(x) - l(d^*(0, x); 0, x) \geq v(x) - l(0; 0, x) = 0$. Now we investigate the case when the equality holds:

$$\begin{aligned} l(0; 0, x) = l(d^*(0, x); 0, x) &\iff 0 \text{ is stationary for } l(d; 0, x) \\ &\iff 0 \in \partial l(0; 0, x) \\ &\iff 0 \in \bar{\partial} v(x) \\ &\iff x \text{ is stationary for } v(x), \end{aligned}$$

where the last equivalence is by (2.7). This proves (i).

Following the same argument for $l(d; \rho, x)$ and $\phi(x; \rho)$ using (2.7), (ii) holds true.

To prove (iii), we know from (ii) that the condition in (iii) means x is stationary for $\phi(x; \rho)$. Therefore, there must exist $\lambda_i \in \partial_c v(c_i(x))$ such that $0 = \rho\nabla f(x) + \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i \nabla c_i(x)$, which is equivalent to

$$\nabla f(x) + \sum_{i\in\mathcal{E}\cup\mathcal{I}} (\lambda_i/\rho)\nabla c_i(x) = 0.$$

Furthermore, it follows from $\lambda_i \in \partial_c v(c_i(x))$ that

$$\lambda_i \geq 0, i \in \mathcal{I}, \quad \text{with} \quad \lambda_i = 0, \text{ if } c_i(x) < 0 \text{ and } i \in \mathcal{I},$$

meaning the complementary condition is satisfied. The constraints are all satisfied since $v(x) = 0$. Overall, we have shown that x is a KKT point with multipliers λ_i/ρ, $i \in \mathcal{E} \cup \mathcal{I}$, as desired. ∎

Overall, the kth iteration of our proposed method proceeds as in Algorithm 1. First of all, a direction and penalty parameter pair (d^k, ρ_k) is computed by a subproblem solver such that d^k yields reductions that satisfy our conditions (2.11a) and (2.11b). Then a line search is executed to find a step size α_k. Finally, the new iterate is set as $x^{k+1} \leftarrow x^k + \alpha_k d^k$ and the algorithm proceeds to the next iteration. The proposed first-order method for nonlinear constrained optimization, hereinafter nicknamed `FoNCO`, is presented as Algorithm 2.1.

Algorithm 1 *F*irst-*o*rder methods for *c*onstrained *o*ptimization (`FoNCO`)

Require $\{\gamma_0, \theta_\alpha, \beta_\alpha\} \in (0,1)$, $\rho_{-1} \in (0, \infty)$.
Choose $x^0 \in \mathbb{R}^n$.
for $k \in \mathbb{N}$ **do**
  Solve $(\mathscr{P})$ (approximately) to obtain $(d^k, \rho_k) \in \mathbb{R}^n \times (0, \rho_{k-1}]$
    or stop if a stationarity certificate is satisfied.
  Let α^k be the largest value in $\{\theta_\alpha^0, \theta_\alpha^1, \theta_\alpha^2, \ldots\}$ such that

$$\phi(x^k; \rho_k) - \phi(x^k + \alpha_k d^k; \rho_k) \geq \beta_\alpha \alpha_k \Delta l(d^k; x^k, \rho_k). \tag{2.12}$$

  Set $x^{k+1} = x^k + \alpha_k d^k$, choose $\gamma_{k+1} \in (0, \gamma_k]$

2.2. Penalty parameter update. At the kth iteration, the value $\rho_k \in (0, \rho_{k-1}]$ needs to be updated to ensure the direction satisfies (2.11a) and (2.11b) as described in the previous subsection can successfully be found. The updating strategy for penalty parameter ρ consists of two phases. Phase I occurs within the subproblem solve and Phase II happens after solving the subproblem.

In the first place, let us consider the Phase I updating strategy employed within one subproblem solve. For ease of presentation, we drop the iteration number k and utilize the following shorthand notation

$$g = \nabla f(x^k), \; a^i = \nabla c_i(x^k), \; b_i = c_i(x^k), \; A = [a^1, \cdots, a^m]^T, \tag{2.13}$$

$l(d; \rho)$ and $p(\lambda; \rho)$ for the kth primal and dual subproblem objectives, respectively. Now problem $(\mathscr{P})$ can be written as

$$(\mathscr{P}') \qquad \min_{d \in X} \; l(d; \rho), \quad \text{where} \quad l(d; \rho) = \langle \rho g, d \rangle + l(d; 0),$$

with its dual $(\mathscr{D})$ written as

$$(\mathscr{D}') \qquad \max_{\lambda_\mathcal{E} \in \mathbb{B}, \lambda_\mathcal{I} \in \mathbb{B}_+} \; p(\lambda; \rho) := -\delta \|\rho g + \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i a^i\|_* + \langle b, \lambda \rangle.$$

After dropping the iteration number k, for a given $\rho > 0$, we can use $(d_\rho^*, \lambda_\rho^*)$ to represent an optimal primal-dual pair for the penalty subproblem corresponding to

ρ; in particular, (d_0^*, λ_0^*) represents an optimal primal-dual pair for the feasibility subproblem.

We are now ready to introduce the penalty parameter ρ updating strategy. Suppose our subproblem solver generates a sequence of $\{d^{(j)}, \lambda^{(j)}, \nu^{(j)}\}$ where $d^{(j)}$ represents the feasible solution estimate for the primal penalty subproblem, $\lambda^{(j)}$ and $\nu^{(j)}$ are the dual feasible solution estimate for the dual penalty and feasibility subproblems, respectively. First of all, we assume the subproblem solver finds primal solution estimate $d^{(j)}$ no worse than a trivial zero step

$$l(d^{(j)}; \rho_{(j)}) \le l(0; \rho_{(j)}), \tag{2.14}$$

the feasibility dual solution $\nu^{(j)}$ is no worse than the penalty dual solution $\lambda^{(j)}$ and both of them are no worse than the initial penalty dual solution

$$p(\nu^{(j)}; 0) \ge p(\lambda^{(j)}; 0) \ge p(\lambda^{(0)}; 0) > -\infty. \tag{2.15}$$

Both of these are reasonable assumptions since if (2.14) (resp. (2.15)) were not to hold, meaning there is no solution can achieve larger reduction in the objective than $d = 0$. Lemma 1 indicates that this must be a certificate of stationarity for optimality or infeasibility. For dual iterates, one can simply use $\nu^{(j)} = \lambda^{(j)} = \lambda^{(0)}$ if there is no better dual estimate than the initial dual estimate.

Now, corresponding to the jth subproblem solver iterate, two ratios are defined:

$$r_v^{(j)} := \frac{l_\gamma^{(0)} - l(d^{(j)}; 0)}{l_\gamma^{(0)} - (p(\nu^{(j)}; 0))_+} \quad \text{and} \quad r_\phi^{(j)} := \frac{l_\gamma^{(0)} - l(d^{(j)}; \rho_{(j)})}{l_\gamma^{(0)} - p(\lambda^{(j)}; \rho_{(j)})}, \tag{2.16}$$

here $l_\gamma^{(0)} := l^{(0)} + \gamma$ with $\gamma \in (0, \infty)$ and

$$l^{(0)} := l(0; \rho) = l(0; 0) = \sum_{i \in \mathcal{E} \cup \mathcal{I}} v_i(b_i) = v(x^k) \ge 0$$

being the primal penalty and feasibility subproblem objective at $d = 0$ for any $\rho \in (0, \infty)$. Note that the numerators of ratios $r_v^{(j)}$ and $r_\phi^{(j)}$ are positive due to the presence of γ. If at the jth iteration, we have

$$r_v^{(j)} \ge \beta_v, \tag{R_v}$$

then the model reduction must satisfy

$$\begin{aligned} l_\gamma^{(0)} - l(d^{(j)}; 0) &\ge \beta_v (l_\gamma^{(0)} - (p(\nu^{(j)}; 0))_+) \\ &\ge \beta_v (l_\gamma^{(0)} - p(\lambda_0^*; 0)) = \beta_v (l_\gamma^{(0)} - l(d_0^*; 0)), \end{aligned} \tag{2.17}$$

where the first and second inequality follows by (R_v) and the optimality of λ_0^* with respect to the feasibility subproblem (which is known that $p(\lambda_0^*, 0) \ge 0$) respectively, and the third one follows by strong duality. Thus condition (2.9a) is satisfied. In a similar way, if at the jth iteration, we have

$$r_\phi^{(j)} \ge \beta_\phi, \tag{R_ϕ}$$

then it follows that

$$\begin{aligned} l_\gamma^{(0)} - l(d^{(j)}; \rho_{(j)}) &\ge \beta_\phi (l_\gamma^{(0)} - p(u^{(j)}; \rho_{(j)})) \\ &\ge \beta_\phi (l_\gamma^{(0)} - p(\lambda^*_{\rho_{(j)}}; \rho_{(j)})) = \beta_\phi (l_\gamma^{(0)} - l(d^*_{\rho_{(j)}}; \rho_{(j)})). \end{aligned} \tag{2.18}$$

Thus condition (2.9b) is satisfied.

As discussed above, the values of ratios $r_v^{(j)}$ and $r_\phi^{(j)}$ reflect the inexactness of current primal-dual $\{d^{(j)}, \lambda^{(j)}, \nu^{(j)}\}$. We need another ratio to measure the satisfaction of the complementarity. Define the index sets

$$\begin{aligned} \mathcal{E}_+(d) &:= \{i \in \mathcal{E} : \langle a^i, d\rangle + b_i > 0\}, \\ \mathcal{E}_-(d) &:= \{i \in \mathcal{E} : \langle a^i, d\rangle + b_i < 0\}, \\ \text{and } \ \mathcal{I}_+(d) &:= \{i \in \mathcal{I} : \langle a^i, d\rangle + b_i > 0\}. \end{aligned}$$

The complementarity measure can be defined accordingly:

$$\chi(d, \lambda) := \sum_{i \in \mathcal{E}_+(d) \cup \mathcal{I}_+(d)} (1 - \lambda_i) v_i(\langle a^i, d\rangle + b_i) + \sum_{i \in \mathcal{E}_-(d)} (1 + \lambda_i) v_i(\langle a^i, d\rangle + b_i).$$

With an optimal primal-dual solution $(d_\rho^*, \lambda_\rho^*)$ for a penalty subproblem, one has $\lambda_i(\lambda_\rho^*) = 1$ for $i \in \mathcal{E}_+(d_\rho^*)$, $\lambda_i(\lambda_\rho^*) = -1$ for $i \in \mathcal{E}_-(d_\rho^*)$, and $\lambda_i(\lambda_\rho^*) = 1$ for $i \in \mathcal{I}_+(d_\rho^*)$, from which it follows that $\chi(d_\rho^*, \lambda_\rho^*) = 0$. Therefore, for a given $\gamma \in (0, \infty)$, the condition (R_c) will hold for sufficiently accurate primal-dual solutions of the penalty subproblem. For an inexact solution, we require that $(d^{(j)}, \lambda^{(j)})$ satisfies

$$\chi^{(j)} := \chi(d^{(j)}, \lambda^{(j)}) \le (1 - \beta_v)^2 l_\gamma^{(0)},$$

or, equivalently,

$$(\mathrm{R}_c) \qquad r_c^{(j)} := 1 - \sqrt{\frac{\chi^{(j)}}{l_\gamma^{(0)}}} \ge \beta_v.$$

Note that the numerator in $r_c^{(j)}$ is always positive due to the presence of $\gamma > 0$.

Now we introduce the dynamic updating strategy (DUST) [9] stated as:

(DUST)

> Given $\rho_{(j)}$ and the jth iterate $(d^{(j)}, \lambda^{(j)}, \nu^{(j)})$, perform the following:
>
> - if (R_ϕ), (R_c), and (R_v) hold, then terminate;
> - else if (R_ϕ) and (R_c) hold, but (R_v) does not, then apply (2.20);
> - else set $\rho_{(j+1)} \leftarrow \rho_{(j)}$.

For a fixed ρ, conditions (R_ϕ) and (R_c) will eventually be satisfied as the subproblem algorithm proceeds. However, this may not be the case for condition (R_v). When this happens, d^k is deemed to be a "successful" inexact direction for minimizing the penalty function, but an "unsuccessful" direction for improving feasibility. The intuition underlying this phenomenon is that a large penalty parameter places too much emphasis on the objective function—a reason to reduce the penalty parameter. Thus we can update the parameter while solving the subproblem as follows. Given

$$(2.19) \qquad 0 < \beta_v < \beta_\phi < 1,$$

we initialize $\rho_{(0)} \leftarrow \rho_{k-1}$ (from the preceding iteration of the outer iteration) and apply the subproblem solver to ($\mathscr{P}'$) to generate $\{(d^{(j)}, \lambda^{(j)}, \nu^{(j)})\}$. We continue to iterate toward solving ($\mathscr{P}'$) until (R_ϕ) and (R_c) are satisfied. Then we terminate

the subproblem algorithm if (R_v) is also satisfied or reduce the penalty parameter by setting

$$\rho_{(j+1)} \leftarrow \theta_\rho \rho_{(j)} \tag{2.20}$$

for some prescribed $\theta_\rho \in (0,1)$.

It is possible that (R_ϕ), (R_c), and (R_v) all hold with $d^{(j)} = 0$ causing the subproblem solver takes a null step. In such a case, we have the subproblem solver terminate with $d^{(j)} = 0$, causing the outer iteration to take a null step in the primal space. This would be followed by a decrease in γ, prompting the outer iteration to eventually make further progress through solving the subproblem or terminate with a stationarity certificate.

On the other hand, if x^k is not stationary with respect to $\phi(\cdot, \rho)$ for any $\rho \in (0, \rho_{k-1}]$, but is stationary with respect to v, then for (d_0^*, λ_0^*) one has

$$\frac{l_\gamma^{(0)} - l(d_0^*; 0)}{l_\gamma^{(0)} - (p(\lambda_0^*; 0))_+} = \frac{\gamma}{\gamma} = 1,$$

meaning that $r_v^{(j)} > \beta_v$ for $(d^{(j)}, \nu^{(j)})$ in a neighborhood of (d_0^*, λ_0^*). One should expect that (DUST) would only reduce the penalty parameter a finite number of times during one subproblem solve. If x^k is near an infeasible stationary point, this may happen consecutively for many subproblems. This may quickly drive the penalty to 0, leading to an infeasible stationary point.

After solving the subproblem, we also consider an additional check of the penalty parameter. Let $\tilde{\rho}_k$ be the value of the penalty parameter obtained by applying (DUST) within the kth subproblem solve. Then, given a constant $\beta_l \in (0, \beta_\phi(1 - \beta_v)]$, we require $\rho_k \in (0, \tilde{\rho}_k]$ so that

$$\Delta l(d^k; \rho_k, x^k) + \gamma_k \geq \beta_l(\Delta l(d^k; 0, x^k) + \gamma_k), \tag{2.21}$$

where the right-hand side of this inequality is guaranteed to be positive due to (R_v). This can be guaranteed by the following *P*osterior *S*ubproblem *ST*rategy:

(PSST)
$$\boxed{\rho_k \leftarrow \begin{cases} \tilde{\rho}_k & \text{if this yields (2.21)} \\ \dfrac{(1-\beta_l)(\Delta l(d^k; 0, x^k) + \gamma_k)}{\langle \nabla f(x^k), d^k \rangle} & \text{otherwise.} \end{cases}}$$

It is possible that $\langle \nabla f(x^k), d^k \rangle \leq 0$, implying

$$\Delta l(d^k; \tilde{\rho}_k, x^k) = -\langle \tilde{\rho}_k \nabla f(x^k), d^k \rangle + \Delta l(d^k; 0, x^k) \geq \Delta l(d^k; 0, x^k),$$

so that (2.21) is always true by setting $\rho_k = \tilde{\rho}_k$. Thus the denominator in the latter formula (PSST) is always positive. On the other hand, if the choice $\rho_k = \tilde{\rho}_k$ does not yield (2.21), then, by setting ρ_k according to the latter formula in (PSST), it follows that

$$\rho_k \langle \nabla f(x^k), d^k \rangle \leq (1 - \beta_l)(\Delta l(d^k; 0, x^k) + \gamma_k),$$

which means that

$$\Delta l(d^k; \rho_k, x^k) + \gamma_k = \Delta l(d^k; 0, x^k) - \rho_k \langle \nabla f(x^k), d^k \rangle + \gamma_k \geq \beta_l(\Delta l(d^k; 0, x^k) + \gamma_k),$$

implying that (2.21) holds. This idea is similar to the updating strategy in various algorithms employing a merit function such as [8]. The difference is that this model reduction condition is imposed inexactly (due to the presence of $\gamma_k > 0$), making (PSST) more suitable for an inexact algorithmic framework. We now summarize the framework of a subproblem solver employing the (DUST) and (PSST) in Algorithm 2.

Algorithm 2 A Framework of Subproblem Algorithm for Solving ($\mathscr{P}$).

Require $(\gamma_k, \beta_\phi) \in (0,1)$, $\beta_v \in (0, \beta_\phi)$, $\beta_l \in (0, \beta_\phi(1-\beta_v))$ and $(\rho_{-1}, \gamma_0) \in (0, \infty)$
Set $\rho_{(0)} \leftarrow \rho_{k-1}$
for $j \in \mathbb{N}$ **do**
    Generate a primal-dual feasible solution estimate $(d^{(j)}, \lambda^{(j)}, \nu^{(j)})$
    Set $\rho_{(j+1)}$ by applying (DUST)
Set $d^k \leftarrow d^{(j)}$ and $\tilde{\rho}_k \leftarrow \rho_{(j)}$.
Set ρ_k by applying (PSST)

3. Convergence analysis. In this section, we analyze the behavior of our proposed algorithmic framework. We first show that if (DUST) is employed within an algorithm for solving ($\mathscr{P}'$), then, under reasonable assumptions on the subproblem data, it is only triggered finite number of times. The second part of this section focuses on the global convergence, which suggests that our proposed algorithm will either converge to a stationary point if (NLP) is feasible or an infeasible stationary point if (NLP) is infeasible under general assumptions.

One of the contributions in this paper is the complexity analysis for the proposed method. We derive the worst-case complexity analysis of the KKT residuals for both the nonlinear optimization problem (NLP) and the feasibility problem (FP). Local complexity analysis for constraint violation is also proved at the last part of this section.

3.1. Worse-case complexity for a single subproblem. The goal of this subsection is to show that the subproblem solver terminates after reducing the penalty parameter for a finite number of times by employing the (DUST) within the subproblem solver for solving ($\mathscr{P}'$). Specifically, we can show that there exists a sufficiently small $\tilde{\rho}$ such that for any $\rho \in (0, \tilde{\rho}]$, if ($\mathrm{R}_\phi$) and ($\mathrm{R}_c$) are satisfied, then ($\mathrm{R}_v$) is also satisfied—a criterion that (DUST) will not be triggered and the subproblem solver should be terminated at this moment. Our complete subproblem algorithm utilizing (DUST) and (PSST) is stated as Algorithm 2. It should be clear that in the inner loop (over j) one is solving a subproblem with quantities dependent on the kth iterate of the main algorithm; see (2.13).

The assumption needed for this analysis is simply the primal-dual feasibility of the iterates, which is formulated as the following.

ASSUMPTION 2. *For all $j \in \mathbb{N}$, the sequence $\{(d^{(j)}, \lambda^{(j)}), \nu^{(j)}\}$ has $d^{(j)} \in X$, $\lambda^{(j)}$ and $\nu^{(j)}$ are feasible for ($\mathscr{D}'$), and (2.14) and (2.15) hold.*

We first show that the differences between the primal and dual values of the optimality and feasibility subproblems are bounded with respect to ρ. Therefore, as ρ tends sufficiently small, the optimality primal (dual) subproblem will approaches the feasibility primal (dual) subproblem.

LEMMA 3. *Under Assumptions 2, it follows that, for any $j \in \mathbb{N}$,*

$$|l(d^{(j)}; \rho_{(j)}) - l(d^{(j)}; 0)| \le \kappa_2 \rho_{(j)} \tag{3.1a}$$

$$\text{and} \quad |p(\lambda^{(j)}; \rho_{(j)}) - p(\lambda^{(j)}; 0)| \le \kappa_3 \rho_{(j)}, \tag{3.1b}$$

with $\kappa_2 := \kappa_0 \delta \|g\|_2$ and $\kappa_3 = \delta \|g\|$. In particular, $\kappa_2 = \delta \|g\|_2$ if $\|\cdot\| = \|\cdot\|_2$ and $\kappa_2 = \sqrt{n}\delta \|g\|_2$ if $\|\cdot\| = \|\cdot\|_\infty$.

Proof. For the primal values, it holds true that

$$|l(d^{(j)};\rho_{(j)}) - l(d^{(j)};0)| = |\rho_{(j)}\langle g, d^{(j)}\rangle| \le \rho_{(j)}\|g\|_2\|d^{(j)}\|_2 \le \rho_{(j)}\kappa_0\|g\|_2\|d^{(j)}\|,$$

where the second inequality is from the requirement (2.8). It then follows that

$$|l(d^{(j)};\rho_{(j)}) - l(d^{(j)};0)| \le \rho_{(j)}\kappa_0\delta\|g\|_2 = \kappa_2\rho_{(j)},$$

proving (3.1a).

The difference between the dual values is given by

$$\begin{aligned}
&|p(\lambda^{(j)};\rho_{(j)}) - p(\lambda^{(j)};0)| \\
&= |-\delta\|\rho_{(j)}g + \sum_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i a_i\| + \delta\|\sum_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i a_i\||, \\
&\le \delta\|\sum_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i a_i - (\rho_{(j)}g + \sum_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i a_i)\|, \\
&= \rho_{(j)}\delta\|g\|,
\end{aligned}$$

completing the proof of (3.1b). ∎

Now we are ready to prove our main result in this section, which needs the following definition

$$\mathcal{R} = \{j : (d^{(j)}, \lambda^{(j)}) \text{ satisfies } (\mathrm{R}_\phi) \text{ and } (\mathrm{R}_c) \text{ but not } (\mathrm{R}_v)\},$$

meaning that $\mathcal{R}$ is the set of subproblem iterations in which (2.20) is triggered.

THEOREM 4. *Suppose Assumptions 2 holds and let*

$$\kappa_4 := \inf_{j\in\mathcal{R}}\{l^{(0)} - l(d^{(j)};\rho_{(j)})\} \ge 0 \quad \textit{and} \quad \kappa_5 := \inf_{j\in\mathcal{R}}\{l^{(0)} - p(\lambda^{(j)};0)\} \ge 0.$$

Then we have the following two cases.

(i) If $g = 0$, then (DUST) is never triggered during the subproblem solve.

(ii) If $g \neq 0$, for $\rho_{(j)} \in (0, \hat{\rho}]$, where

$$\hat{\rho} := \frac{\gamma + \min\{\kappa_4, \kappa_5\}}{\max\{\kappa_2, \kappa_3\}}\left(1 - \sqrt{\beta_v/\beta_\phi}\right), \tag{3.2}$$

if $(d^{(j)}, \lambda^{(j)})$ satisfies (R_ϕ) and (R_c), then $(d^{(j)}, \nu^{(j)})$ satisfies (R_v). In other words, for any $\rho_{(j)} \in (0, \hat{\rho}]$, the update (2.20) is never triggered by (DUST).

Proof. We first prove (i). If $g = 0$, we know that $l(d^{(j)};\rho) = l(d^{(j)};0)$ and $p(\lambda^{(j)};\rho) = p(\lambda^{(j)};0)$, implying $r_\phi^{(j)} = r_v^{(j)}$. Therefore, if (R_ϕ) and (R_c) are satisfied, then $r_v^{(j)} = r_\phi^{(j)} \ge \beta_\phi > \beta_v$ satisfying (R_v), as desired.

As for (ii), the denominator of $\tilde{\rho}$ is positive since $g \neq 0$. We prove (ii) by contradiction and assume that $\mathcal{R}$ is infinite, indicating that the subproblem solver is never terminated and $\rho_{(j)}$ is reduced infinite many times and driven to 0. We have from (3.1a) that

$$-\kappa_2\rho_{(j)} \le l(d^{(j)};\rho_{(j)}) - l(d^{(j)};0) \le \kappa_2\rho_{(j)} \quad \text{for any } j \in \mathcal{R},$$

which, after adding and dividing through by $l_\gamma^{(0)} - l(d^{(j)};\rho_{(j)})$, yields for $j \in \mathcal{R}$ that

$$1 - \frac{\kappa_2\rho_{(j)}}{l_\gamma^{(0)} - l(d^{(j)};\rho_{(j)})} \le \frac{l_\gamma^{(0)} - l(d^{(j)};0)}{l_\gamma^{(0)} - l(d^{(j)};\rho_{(j)})} \le 1 + \frac{\kappa_2\rho_{(j)}}{l_\gamma^{(0)} - l(d^{(j)};\rho_{(j)})}. \tag{3.3}$$

Thus, for any

$$\rho_{(j)} \leq \frac{\gamma + \kappa_4}{\kappa_2}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right) \leq \frac{l_\gamma^{(0)} - l(d^{(j)}; \rho_{(j)})}{\kappa_2}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right),$$

it follows from the first inequality of (3.3) that

$$\frac{l_\gamma^{(0)} - l(d^{(j)}; 0)}{l_\gamma^{(0)} - l(d^{(j)}; \rho_{(j)})} \geq \sqrt{\frac{\beta_v}{\beta_\phi}}. \tag{3.4}$$

Following a similar argument from (3.1b), it follows that for any

$$\rho_{(j)} \leq \frac{\gamma + \kappa_5}{\kappa_3}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right) \leq \frac{l_\gamma^{(0)} - p(\lambda^{(j)}; 0)}{\kappa_3}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right),$$

one finds that

$$\frac{l_\gamma^{(0)} - p(\lambda^{(j)}; \rho_{(j)})}{l_\gamma^{(0)} - p(\lambda^{(j)}; 0)} \geq \sqrt{\frac{\beta_v}{\beta_\phi}}. \tag{3.5}$$

Overall, we have shown that for any $\rho_{(j)} \leq \tilde{\rho}$ with $\tilde{\rho}$ defined in (3.2), it follows that (3.4) and (3.5) both hold true.

Since our supposition that $\mathcal{R}$ is infinite implies that $\rho_{(j)} \to 0$, we may now proceed under the assumption that $j \in \mathcal{R}$ with $\rho_{(j)} \in (0, \tilde{\rho}]$. Let us now define the ratios

$$\bar{r}_v^{(j)} := \frac{l_\gamma^{(0)} - l(d^{(j)}; 0)}{l_\gamma^{(0)} - p(\lambda^{(j)}; 0)},$$

where it must be true that $r_v^{(j)} \geq \bar{r}_v^{(j)}$ by the definition of operator $(\cdot)_+$. From (3.4)

$$\frac{\bar{r}_v^{(j)}}{r_\phi^{(j)}} = \frac{l_\gamma^{(0)} - l(d^{(j)}; 0)}{l_\gamma^{(0)} - l(d^{(j)}; \rho_{(j)})} \frac{l_\gamma^{(0)} - p(\lambda^{(j)}; \rho_{(j)})}{l_\gamma^{(0)} - p(\lambda^{(j)}; 0)} \geq \frac{\beta_v}{\beta_\phi},$$

yielding

$$r_v^{(j)} \geq \bar{r}_v^{(j)} \geq \frac{\beta_v}{\beta_\phi} r_\phi^{(j)} \geq \beta_v.$$

However, this contradicts the fact that $j \in \mathcal{R}$. Overall, since we have reached a contradiction, we may conclude that $\mathcal{R}$ is finite. ∎

We can use Theorem 4 to estimate the number of reductions occured during a single subproblem solve, as well as a lower bound of the penalty parameter after solving the subproblem, which is summarized in the following theorem. Since it describes results about the main algorithm, we add back the k index to denote the kth iteration of main algorithm.

THEOREM 5. *Suppose Assumptions 2 holds, then after solving the kth subproblem, we have*

$$\tilde{\rho}_k \geq \min\left(\rho_{k-1}, \frac{\theta_\rho \gamma_k}{\max(\kappa_0^2, 1)\delta}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right)\frac{1}{\|\nabla f(x^k)\|}\right). \tag{3.6}$$

Moreover, (DUST) is triggered at most

$$\left\lceil \frac{1}{\ln \theta_\rho} \ln\left(\frac{\gamma_k}{\max(\kappa_0^2, 1)\delta}\left(1 - \sqrt{\frac{\beta_v}{\beta_\phi}}\right)\frac{1}{\rho_{k-1}\|\nabla f(x^k)\|}\right)\right\rceil \tag{3.7}$$

times during solving the kth subproblem.

Proof. From Theorem 4, we know that if $\hat{\rho}_k \geq \rho_{k-1}$, then (DUST) is never triggered. If this is not the case, since ρ is reduced by a fraction whenever it is updated, from Theorem 4, we know the final ρ returned by the subproblem solver must satisfy

$$\tilde{\rho}_k \geq \theta_\rho \hat{\rho}_k \geq \frac{\theta_\rho \gamma_k}{\max\{\kappa_0^2 \delta \|\nabla f(x^k)\|, \delta \|\nabla f(x^k)\|\}} \left(1 - \sqrt{\beta_v/\beta_\phi}\right).$$

by noticing

$$\|\nabla f(x^k)\|_2 \leq \kappa_0 \|\nabla f(x^k)\|.$$

This completes the proof of (3.6).

For (3.7), suppose $\hat{\rho}_k < \rho_{k-1}$ so that (DUST) is triggered during the subproblem solve. Also suppose after $\hat{j}$ reductions, we have $\theta_\rho^{\hat{j}} \rho_{k-1} \leq \hat{\rho}_k$. Taking logarithm of both sides, after simple rearrangement, we have

$$\hat{j} \leq \frac{\ln(\hat{\rho}_k/\rho_{k-1})}{\ln \theta_\rho}.$$

Notice that both the denominator and the numerator are negative. This inequality, combined with (3.6), proves (3.7). ∎

From (3.6) and (3.7) in Theorem 5, it would be worth noticing that many factors may affect the the number of times that (DUST) is triggered within a single subproblem.

- Smaller $\|\nabla f(x^k)\|$ will result in fewer (DUST) updates. Intuitively, in this case, $l(d; \rho, x^k)$ is close to $l(d; 0, x^k)$, so that any direction successful for optimality may be also successful for feasibility. As for larger $\|\nabla f(x^k)\|$, we may need more updates.
- The accuracy tolerance γ_k also affects the number of updates needed. If we have aggressively small γ_k, meaning the subproblem needs to be solved more accurately, then (DUST) updates may happen more frequently.
- The trust region radius also plays a role in the number of (DUST) updates, and smaller δ may lead to fewer updates. This is reasonable since the difference between $l(d; \rho, x^k)$ and $l(d; 0, x^k)$ should be smaller in this case within trust region X.
- We can also see the influence of the algorithmic parameters here. A smaller θ_ρ naturally leads to fewer updates but possibly smaller ρ since it is reduced more aggressively each time. It would be interesting to see that if one chooses $\beta_v \to \beta_\phi$ (note $\beta_v < \beta_\phi$), then (DUST) may occur a lot more times. The intuition of this case is that we require a direction successful for optimality should also be the same successful for feasibility, which could only happen for very small ρ.

3.2. Global Convergence. In this subsection, we show that if (DUST) and (PSST) are used to solve (NLP) in a penalty-SLP algorithm. Then, the algorithm can converge from any starting point if we have reasonable assumptions. Specifically, if (DUST) and (PSST) are only triggered a finite number of times, then every limit point of the iterates is either infeasible stationary or first-order stationary for (NLP). Otherwise, if (DUST) and (PSST) are triggered an infinite number of times, driving the penalty parameter to zero, then every limit point of the iterates is either an infeasible stationary point or a feasible point at which a constraint qualification fails to hold.

For the analysis in this section, we extend our use of the sub/superscript k to represent the value of quantities associated with iteration $k \in \mathbb{N}$. For instance, $\mathcal{R}^k$ denotes the set $\mathcal{R}$ defined in §3.1 at the kth iteration.

In the whole process of analysis, we assume the following.

ASSUMPTION 6. *Functions f and c_i for all $i \in \mathcal{E} \cup \mathcal{I}$, and their first- and second-order derivatives, are all bounded in an open convex set containing $\{x^k\}$ and $\{x^k + d^k\}$. Also assume that $\gamma_k \to 0$.*

Define the index set

$$\mathcal{U} := \{k \in \mathbb{N} : \mathcal{R}^k \neq \emptyset\}.$$

Moreover, for every $k \in \mathcal{U}$, let j_k be the subproblem iteration number corresponding to the value of the smallest ratio r_v, i.e.,

$$r_v^{(j_k)} \leq r_v^{(i_k)} \quad \text{for any} \quad i_k \in \mathcal{R}^k.$$

Also, define the index set

$$\mathcal{T} := \{k \in \mathbb{N} : \rho_k \text{ is reduced by (PSST)}\}.$$

From these definitions, it follows that $\rho_k < \rho_{k-1}$ if and only if $k \in \mathcal{U} \cup \mathcal{T}$.

We also have the following fact about the subproblem solutions, the proof of which is skipped here since it can be easily derived by noticing $\|d^k\|_2 \leq \kappa_0 \|d^k\| \leq \kappa_0 \delta$ in the proof of [9, Lemma 10].

LEMMA 7. *Under Assumption 2 and 6, it follows that, for all $k \in \mathbb{N}$, the stepsize satisfies*

$$\alpha_k \geq \frac{\theta_\alpha(1-\beta_\alpha)}{\delta \kappa_0 \kappa_1} \Delta l(d^k; \rho_k, x^k).$$

We now prove that, in the limit, the reductions in the models of the constraints violation measure and the penalty function vanish. For this purpose, it will be convenient to work with the shifted penalty function

$$\varphi(x, \rho) := \rho(f(x) - \underline{f}) + v(x) \geq 0,$$

where $\underline{f}$(its existance follows from Assumption 6) is the infimum of f over the smallest convex set containing $\{x^k\}$. In the following lemma, it proves that the function φ possesses a useful monotonicaity property.

LEMMA 8. *Under Assumption 2 and 6, it holds that, for all $k \in \mathbb{N}$,*

$$\varphi(x^{k+1}, \rho_{k+1}) \leq \varphi(x^k, \rho_k) - \frac{\theta_\alpha(1-\beta_\alpha)\beta_\alpha}{\delta \kappa_0 \kappa_1} [\Delta l(d^k; \rho_k, x^k)]^2, \tag{3.8}$$

Proof. From the line search condition (2.12)

$$\varphi(x^{k+1}, \rho_{k+1}) \leq \varphi(x^k, \rho_k) - (\rho_k - \rho_{k+1})(f(x^{k+1}) - \underline{f}) - \beta_\alpha \alpha_k \Delta l(d^k; \rho_k, x^k).$$

Then (3.8) follows from this inequality, Lemma 7, the fact that $\{\rho_k\}$ is monotonically decreasing, and $f(x^{k+1}) \geq \underline{f}$ for all $k \in \mathbb{N}$. ∎

We now show the model reductions and duality gap all vanish asymptotically.

LEMMA 9. *Under Assumption 2 and 6, the following limits hold.*
(i) $0 = \lim_{k\to\infty} \Delta l(d^k; \rho_k, x^k) = \lim_{k\to\infty} \Delta l(d^k; 0, x^k)$,

(ii) $0 = \lim_{k\to\infty} [l(0;\rho_k,x^k) - p(\lambda^k;\rho_k,x^k)] = \lim_{k\to\infty} [l(0;0,x^k) - p(\nu^k;0,x^k)]$,
(iii) $0 = \lim_{k\to\infty} \Delta l(d^*(0,x^k);0,x^k) = \lim_{k\to\infty} \Delta l(d^*(\rho_k,x^k);\rho_k,x^k)$,
(iv) $0 = \Delta l(d^*(0,x^*);0,x^*) = \Delta l(d^*(\rho_*,x^*);\rho_*,x^*)$ *with* $\rho_* := \lim_{k\to\infty} \rho_k$ *for any limit point* x^* *of* $\{x^k\}$.

Proof. Let us first prove (i) by contradiction. Suppose that $\Delta l(d^k;\rho_k,x^k)$ does not converge to 0. Then, there exists a constant $\epsilon > 0$ and an infinite $\mathcal{K} \subseteq \mathbb{N}$ such that $\Delta l(d^k;\rho_k,x^k) \geq \epsilon$ for all $k \in \mathcal{K}$. It then follows from Lemma 8 that $\varphi(x^k;\rho_k) \to -\infty$, which contradicts the fact that $\{\varphi(x^k,\rho_k)\}$ is bounded below by zero. Therefore, $\Delta l(d^k;\rho_k,x^k) \to 0$. The second limit in (i) follows from (2.21) and $\gamma_k \to 0$.

The limits in (ii) and (iii) follow directly from the limits in (i) and the inequalities in (2.17) and (2.18) along with $\gamma_k \to 0$; (iv) follows directly from (iii). ∎

We now provide our first global convergence theorem.

THEOREM 10. *Under Assumption 2 and 6, the following statements hold.*
(i) Any limit point of $\{x^k\}$ *is first-order stationary for* v*, i.e., it is feasible or an infeasible stationary point for* (NLP).
(ii) If $\rho_k \to \rho_*$ *for some constant* $\rho_* > 0$ *and* $v(x^k) \to 0$*, then any limit point* x^* *of* $\{x^k\}$ *with* $v(x^*) = 0$ *is a KKT point for* (NLP).
(iii) If $\rho_k \to 0$*, then either all limit points of* $\{x^k\}$ *are feasible for* (NLP) *or all are infeasible.*

Proof. Part (i) and part (ii) follow by combining Lemma 9(iv) with Lemma 1(i) and Lemma 9(iv) with Lemma 1(ii) respectively.

We prove (iii) by contradiction. Suppose there exist infinite $\mathcal{K}^* \subseteq \mathbb{N}$ and $\mathcal{K}^\times \subseteq \mathbb{N}$ such that $\{x^k\}_{k\in\mathcal{K}^*} \to x^*$ with $v(x^*) = 0$ and $\{x^k\}_{k\in\mathcal{K}^\times} \to x^\times$ with $v(x^\times) = \epsilon > 0$. Since $\rho_k \to 0$, there exists $k^* \geq 0$ such that for all $k \in \mathcal{K}^*$ and $k \geq k^*$ one has that $\rho_k(f(x^k) - \underline{f}) < \epsilon/4$ and $v(x^k) < \epsilon/4$, meaning that $\varphi(x^k,\rho_k) < \epsilon/2$. On the other hand, it follows that $\rho_k(f(x^k) - \underline{f}) \geq 0$ for all $k \in \mathbb{N}$ and there exists $k^\times \in \mathbb{N}$ such that $v(x^k) \geq \epsilon/2$ for all $k \geq k^\times$ with $k \in \mathcal{K}^\times$, meaning that $\varphi(x^k,\rho_k) \geq \epsilon/2$. This contradicts Lemma 8, which shows that $\varphi(x^k,\rho_k)$ is monotonically decreasing. Therefore, the set of limit points of $\{x^k\}$ must be all feasible or all infeasible. ∎

The result of Theorem 10 is satisfactory in the case when $\rho_k \to \rho_* > 0$, in which case it is proved that any limit point of the primal sequence is a KKT point for (NLP). However, more should be said in the case that $\rho_k \to 0$; in particular, in the following results, we look further at this case and aim to show that it will only occur if any limit point of the algorithm is an infeasible stationary point or a feasible point at which a constraint qualification fails to hold. For this analysis, we first provide the following.

LEMMA 11. *Suppose Assumption 2 and 6 hold and* $\rho_k \to 0$*. Let* x^* *be a limit point of* $\{x^k\}_{k\in\mathcal{U}\cup\mathcal{T}}$ *that is feasible for* (NLP) *with infinite* $\mathcal{S} \subseteq \mathcal{U} \cup \mathcal{T}$ *such that* $\{x^k\}_{k\in\mathcal{S}} \to x^*$*. Then, the following hold true.*
(i) $|\mathcal{S} \cap \mathcal{U}|$ *is finite or* $\{\Delta l(d^{(j_k)};\rho_{(j_k)},x^k)\}_{k\in\mathcal{S}\cap\mathcal{U}} \to 0$*;*
(ii) any limit point of $\{\lambda^{(j_k)}\}_{k\in\mathcal{S}\cap\mathcal{U}} \cup \{\lambda^k\}_{k\in\mathcal{S}\cap\mathcal{T}}$ *is optimal for* $p(\cdot;0,x^*)$*;*
(iii) $\{\lambda^{(j_k)}\}_{k\in\mathcal{S}\cap\mathcal{U}} \cup \{\lambda^k\}_{k\in\mathcal{S}\cap\mathcal{T}}$ *has a nonzero limit point.*

Proof. For part (i), if $|\mathcal{S} \cap \mathcal{U}|$ is finite, then there is nothing left to prove. As a

result, we assume that $|\mathcal{S} \cap \mathcal{U}| = \infty$. By observing that, for all $k \in \mathbb{N}$, it holds that

$$\begin{aligned} 0 &\leq \Delta l(d^{(j_k)}; \rho_{(j_k)}, x^k) \\ &= v(x^k) - \rho_{(j_k)} \langle \nabla f(x^k), d^{(j_k)} \rangle - l(d^{(j_k)}; 0, x^k) \\ &\leq v(x^k) - \rho_{(j_k)} \langle \nabla f(x^k), d^{(j_k)} \rangle, \end{aligned}$$

where the first inequality is by (2.14) and the second one is by the definition of l, which ensures that $l(d^{(j_k)}; 0, x^k) \geq 0$. Moreover, $\{d^{(j_k)}\}$ is bounded because of the trust-region constraint. Accordingly, the limit in part (i) holds due to $|\mathcal{S} \cap \mathcal{U}| = \infty$ and $\{v(x^k)\}_{k \in \mathcal{S} \cap \mathcal{U}} \to 0$ with $\rho^{(j_k)} \to 0$.

Now consider part (ii). If $|\mathcal{S} \cap \mathcal{U}|$ is infinite, then for a limit point λ^* there must exist an infinite $\mathcal{S}_\mathcal{U} \subseteq \mathcal{S} \cap \mathcal{U}$ such that $\{\lambda^{(j_k)}\}_{k \in \mathcal{S}_\mathcal{U}} \to \lambda^*$. Then, it follows that

$$\begin{aligned} 0 &\leq l(0; 0, x^*) - p(\lambda^*; 0, x^*) \\ &= \lim_{\substack{k \in \mathcal{S}_\mathcal{U} \\ k \to \infty}} [l(0; 0, x^k) + \gamma_k - p(\lambda^{(j_k)}; \rho_{(j_k)}, x^k)] \\ &\leq \lim_{\substack{k \in \mathcal{S}_\mathcal{U} \\ k \to \infty}} \frac{1}{\beta_\phi} [l(0; 0, x^k) + \gamma_k - l(d^{(j_k)}; \rho_{(j_k)}, x^k)] \\ &= 0, \end{aligned} \tag{3.9}$$

where the first and the second inequality follows from weak duality, (R_ϕ) and $\gamma_k \to 0$ separately. And the last equality follows from part(i) and $\gamma_k \to 0$. This means that λ^* is optimal for $p(\cdot; 0, x^*)$. On the other hand, if $|\mathcal{S} \cap \mathcal{U}|$ is finite, then $|\mathcal{S} \cap \mathcal{T}|$ must be infinite, in which case for a limit point λ^* there must exist an infinite $\mathcal{S}_\mathcal{T} \subseteq \mathcal{S} \cap \mathcal{T}$ such that $\{\lambda^k\}_{k \in \mathcal{S}_\mathcal{T}} \to \lambda^*$. Then, again

$$l(0; 0, x^*) - p(\lambda^*; 0, x^*) = \lim_{\substack{k \in \mathcal{S}_\mathcal{T} \\ k \to \infty}} [l(0; 0, x^k) - p(\lambda^k; \rho_k, x^k)] = 0$$

from Lemma 9(ii), meaning that λ^* is optimal for $p(\cdot; 0, x^*)$.

For part (iii), first we can observe that

$$l(d; 0, x^k) = \sum_{i \in \mathcal{E}_+^k(d) \cup \mathcal{E}_-^k(d) \cup \mathcal{I}_+^k(d)} v_i(c_i(x^k) + \langle \nabla c_i(x^k), d \rangle),$$

and that $\chi(d, \lambda; x^k)$ can be viewed as a weighted variant of this sum with weights

$$1 - \lambda_i \quad \text{for all} \quad i \in \mathcal{E}_+^k(d) \cup \mathcal{I}_+^k(d) \quad \text{and} \quad 1 + \lambda_i \quad \text{for all} \quad i \in \mathcal{E}_-^k(d).$$

We can also observe that (R_c) holds at any primal-dual point

$$(d, \lambda) \in \{(d^{(j_k)}, \lambda^{(j_k)})\}_{k \in \mathcal{S} \cap \mathcal{U}} \cup \{(d^k, \lambda^k)\}_{k \in \mathcal{S} \cap \mathcal{T}}$$

owing to the facts that

$$\chi(d^{(j_k)}, \lambda^{(j_k)}; x^k) \leq (1 - \beta_v)^2 (v(x^k) + \gamma_k) \quad \text{for all} \quad k \in \mathcal{S} \cap \mathcal{U} \quad \text{and} \tag{3.10}$$

$$\chi(d^k, \lambda^k; x^k) \leq (1 - \beta_v)^2 (v(x^k) + \gamma_k) \quad \text{for all} \quad k \in \mathcal{S} \cap \mathcal{T}. \tag{3.11}$$

Three cases are considered in the following.

Case (a): Assume there exists an infinite $\mathcal{S}_{\mathcal{U}} \subseteq \mathcal{S} \cap \mathcal{U}$ such that

$$l(d^{(j_k)}; 0, x^k) > (1 - \beta_v)(v(x^k) + \gamma_k) \quad \text{for all} \quad k \in \mathcal{S}_{\mathcal{U}}. \tag{3.12}$$

Then, it must be true that $\|\lambda^{(j_k)}\|_\infty \geq \beta_v$ for all $k \in \mathcal{S}_{\mathcal{U}}$; actually, if this were not the case, then for some $k \in \mathcal{S}_{\mathcal{U}}$ one would find from the definition of χ and (3.12) that

$$\chi(d^{(j_k)}, \lambda^{(j_k)}; x^k) \geq (1 - \beta_v) l(d^{(j_k)}; 0, x^k) > (1 - \beta_v)^2 (v(x^k) + \gamma_k),$$

contradicting (3.10). In this case, combining the boundedness of $\{\lambda^{(j_k)}\}$, Assumption 6(iv) and the fact that $\|\lambda^{(j_k)}\|_\infty \geq \beta_v$ for all $k \in \mathcal{S}_{\mathcal{U}}$ shows that $\{\lambda^{(j_k)}\}_{k \in \mathcal{S} \cap \mathcal{U}}$ has a nonzero limit point, proving part (iii), as desired.

Case (b): Assume there exists an infinite $\mathcal{S}_{\mathcal{T}} \subseteq \mathcal{S} \cap \mathcal{T}$ such that

$$l(d^k; 0, x^k) > (1 - \beta_v)(v(x^k) + \gamma_k) \quad \text{for all} \quad k \in \mathcal{S}_{\mathcal{T}}. \tag{3.13}$$

Then, it must be true that $\|\lambda^k\|_\infty \geq \beta_v$ for all $k \in \mathcal{S}_{\mathcal{T}}$; actually, if this were not the case, then for some $k \in \mathcal{S}_{\mathcal{T}}$ one would find from the definition of χ and (3.13) that

$$\chi(d^k, \lambda^k; x^k) \geq (1 - \beta_v) l(d^k; 0, x^k) > (1 - \beta_v)^2 (v(x^k) + \gamma_k),$$

contradicting (3.11). In this case, combining the boundedness of $\{x^k\}_{k \in \mathcal{T}}$, Assumption 6$(iv)$, and the fact that $\|\lambda^k\|_\infty \geq \beta_v$ for all $k \in \mathcal{S}_{\mathcal{T}}$ shows that $\{\lambda^k\}_{k \in \mathcal{S} \cap \mathcal{T}}$ has a nonzero limit point, proving part (iii), as desired.

Case (c): Suppose that (3.12) and (3.13) only hold for finite subsets of $\mathcal{S} \cap \mathcal{U}$ and $\mathcal{S} \cap \mathcal{T}$. In this case, there exists a sufficiently large $\bar{k} \in \mathbb{N}$ such that

$$l(d^{(j_k)}; 0, x^k) \leq (1 - \beta_v)(v(x^k) + \gamma_k) \text{ for all } k \in \mathcal{S} \cap \mathcal{U} \text{ with } k \geq \bar{k}; \tag{3.14}$$

$$l(d^k; 0, x^k) \leq (1 - \beta_v)(v(x^k) + \gamma_k) \text{ for all } k \in \mathcal{S} \cap \mathcal{T} \text{ with } k \geq \bar{k}. \tag{3.15}$$

We can further assume that

$$\begin{aligned} \|\lambda^{(j_k)}\|_\infty < \beta_v \text{ for all } k \in \mathcal{S} \cap \mathcal{U} \text{ with } k \geq \bar{k} \text{ and} \\ \|\lambda^k\|_\infty < \beta_v \text{ for all } k \in \mathcal{S} \cap \mathcal{T} \text{ with } k \geq \bar{k}; \end{aligned}$$

since otherwise, as in Cases (a) and (b), respectively, part (iii) would hold. Now, for $k \geq \bar{k}$ with $k \in \mathcal{S} \cap \mathcal{U}$, it follows from (3.14) that

$$\begin{aligned} & l(0; 0, x^k) + \gamma_k - l(d^{(j_k)}; 0, x^k) \\ \geq\ & v(x^k) + \gamma_k - (1 - \beta_v)(v(x^k) + \gamma_k) \\ =\ & \beta_v (v(x^k) + \gamma_k) \\ \geq\ & \beta_v [v(x^k) + \gamma_k - (p(\nu^{(j_k)}, 0; x^k))_+], \end{aligned}$$

from which it follows that

$$r_v^{(j_k)} = \frac{l(0; 0, x^k) + \gamma_k - l(d^{(j_k)}; 0, x^k)}{v(x^k) + \gamma_k - (p(\nu^{(j_k)}; 0, x^k))_+} \geq \beta_v.$$

This indicates that (DUST) is not triggered at any iteration $k \geq \bar{k}$ with $k \in \mathcal{S} \cap \mathcal{U}$. The definition of $\mathcal{U}$ implies that $\mathcal{S} \cap \mathcal{U}$ is finite. On the other hand, for $k \in \mathcal{S} \cap \mathcal{T}$ with $k \geq \bar{k}$, it holds that

$$
\begin{aligned}
& l(0;0,x^k) - p(\lambda^k;\rho_k,x^k) \\
\geq\ & v(x^k) - \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i^k c_i(x^k) \\
=\ & \sum_{i\in\mathcal{E}\cup\mathcal{I}} v_i(c_i(x^k)) - \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i^k c_i(x^k) \\
=\ & \sum_{i\in\mathcal{E}} [|c_i(x^k)| - \lambda_i^k c_i(x^k)] + \sum_{i\in\mathcal{I}} [(c_i(x^k))_+ - \lambda_i^k c_i(x^k)] \\
\geq\ & \sum_{i\in\mathcal{E}} (1-|\lambda_i^k|)|c_i(x^k)| + \sum_{i\in\mathcal{I}} (1-|\lambda_i^k|)(c_i(x^k))_+ \\
\geq\ & (1-\beta_v) \sum_{i\in\mathcal{E}} v_i(c_i(x^k)) \\
=\ & (1-\beta_v) v(x^k),
\end{aligned}
\tag{3.16}
$$

where the first inequality is from the definition of $p(\lambda^k;\rho_k,x^k)$. Since (R_ϕ) is satisfied, the first inequality in (2.18) and (3.16) imply

$$
\begin{aligned}
& \Delta l(d^k;\rho_k,x^k) + \gamma_k \\
=\ & l(0;0,x^k) - l(d^k;\rho_k,x^k) + \gamma_k \\
\geq\ & \beta_\phi [l(0;0,x^k) - p(\lambda^k;\rho_k,x^k) + \gamma_k] \\
\geq\ & \beta_\phi [(1-\beta_v) v(x^k) + \gamma_k] \\
\geq\ & \beta_\phi (1-\beta_v)(v(x^k) + \gamma_k) \\
\geq\ & \beta_l (v(x^k) + \gamma_k) \\
\geq\ & \beta_l (\Delta l(d^k;0,x^k) + \gamma_k).
\end{aligned}
$$

yielding

$$
\Delta l(d^k;\rho^k,x^k) + \gamma_k \geq \Delta l(d^k;\rho_k,x^k) + \gamma_k \geq \beta_l (\Delta l(d^k;0,x^k) + \gamma_k).
$$

Therefore, we know that (PSST) is not triggered in any iteration $k \in \mathcal{S} \cap \mathcal{T}$ with $k \geq \bar{k}$. The definition of $\mathcal{T}$ means that $\mathcal{S} \cap \mathcal{T}$ is finite. In general, in this case, $\mathcal{S} \cap \mathcal{U}$ and $\mathcal{S} \cap \mathcal{T}$ are finite which means that $\mathcal{S}$ is finite. However, this contradicts the statement of the lemma, which defines $\mathcal{S}$ to be finite.

Overall, since Case (c) leads to a contradiction, it follows that either Case (a) or (b) must occur, which proves part (iii). ∎

Now we are ready to prove a theorem about the behavior of the algorithm when the penalty parameter is driven to zero. The theorem involves a statement about points satisfying the well-known Mangasarian-Fromovitz constraint qualificatioin (MFCQ). Defining

$$
\mathcal{A}(x) = \{i \in \mathcal{I} : c_i(x) = 0\} \ \text{ and } \ \mathcal{N}(x) = \{i \in \mathcal{I} : c_i(x) < 0\},
$$

the MFCQ is defined as follows.

DEFINITION 12. *A point x satisfies the MFCQ for problem (NLP) if $v(x) = 0$, $\{\nabla c_i(x) : i \in \mathcal{E}\}$ are linearly independent, and there exists $d \in \mathbb{R}^n$ such that*

$$\begin{aligned} & c_i(x) + \langle \nabla c_i(x), d\rangle = 0 \quad \text{for all} \quad i \in \mathcal{E} \\ \text{and} \quad & c_i(x) + \langle \nabla c_i(x), d\rangle < 0 \quad \text{for all} \quad i \in \mathcal{I}. \end{aligned}$$

The equivalent form, namely the dual form [25] of MFCQ, states that $\lambda = 0$ is the unique solution of

$$\sum_{i \in \mathcal{E} \cup \mathcal{A}(x^*)} \lambda_i \nabla c_i(x^*) = 0. \tag{3.17}$$

We are now ready to state and prove the main result.

THEOREM 13. *Suppose Assumption 2 and 6 hold and $\rho_k \to 0$. Then, every limit point of $\{x^k\}_{k \in \mathcal{U} \cup \mathcal{T}}$ is either an infeasible stationary point or a feasible point where the MFCQ does not hold.*

Proof. By Theorem 10(i), any limit point of $\{x^k\}_{k \in \mathcal{U} \cup \mathcal{T}}$ is either feasible or an infeasible stationary point. If any such point is infeasible, then there is nothing left to prove. Therefore, we can let x^* denote a feasible limit point of $\{x^k\}_{k \in \mathcal{U} \cup \mathcal{T}}$. Our goal is to show that the MFCQ fails to hold at x^*.

Let $\mathcal{S} \subseteq \mathcal{U} \cup \mathcal{T}$ be an infinite set such that $\{x^k\}_{k \in \mathcal{S}} \to x^*$. Theorem 11($iii$) says there exists a nonzero limit point λ^* of $\{\lambda^{(j_k)}\}_{k \in \mathcal{S} \cap \mathcal{U}} \cup \{\lambda^k\}_{k \in \mathcal{S} \cap \mathcal{T}}$. In addition, from Lemma 9, it follows that

$$\begin{aligned} 0 = v(x^*) &= l(0; 0, x^*) = p(\lambda^*; 0, x^*) \\ &= \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i^* c_i(x^*) - \delta \| \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i^* \nabla c_i(x^*) \|_* \\ &\leq \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i^* c_i(x^*) \\ &= \sum_{i \in \mathcal{N}(x^*)} \lambda_i^* c_i(x^*). \end{aligned} \tag{3.18}$$

Since $\sum_{i \in \mathcal{N}(x^*)} \lambda_i^* c_i(x^*) \leq 0$, it follows that $\sum_{i \in \mathcal{N}(x^*)} \lambda_i^* c_i(x^*) = 0$, yielding $\lambda_*^i = 0$ for all $i \in \mathcal{N}(x^*)$. Therefore, (3.18) implies

$$\sum_{i \in \mathcal{E} \cup \mathcal{A}(x^*)} \lambda_i^* \nabla c_i(x^*) = 0. \tag{3.19}$$

Note that $\lambda^* \neq 0$ by Lemma 11. Therefore, (3.19) violates the dual form of MFCQ, completing the proof. ∎

In the following corollary, we summarize the results of all of our theorems .

COROLLARY 14. *Suppose Assumption 2 and 6 hold. Then, exactly one of the following occurs*

- *(i) $\rho_k \to \rho_*$ for some constant $\rho_* > 0$ and each limit point of $\{x^k\}$ either corresponds to a KKT point or an infeasible stationary point for problem (NLP).*
- *(ii) $\rho_k \to 0$ and all limit points of $\{x^k\}$ are infeasible stationary points for (NLP).*
- *(iii) $\rho_k \to 0$, all limit points of $\{x^k\}$ are feasible for (NLP), and the MFCQ fails to hold at all limit points of $\{x^k\}_{k \in \mathcal{U} \cup \mathcal{T}}$.*

3.3. Worst-case complexity for KKT residuals. In this subsection, we aim to show the worst-case complexity of the KKT residuals for both the penalty problem and the feasibility problem, which are denoted as

$$E_{opt}(x,\lambda,\rho) = \|\rho\nabla f(x) + \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i \nabla c_i(x)\|_*$$
$$\text{and} \quad E_{fea}(x,\nu) = \|\sum_{i\in\mathcal{E}\cup\mathcal{I}} \nu_i \nabla c_i(x)\|_*,$$

The subproblem always chooses dual feasible variables λ^k and ν^k satisfying (2.5b). Therefore, we verify the satisfaction of complementarity (2.5c) by defining the complementary residual as

$$E_c(x,\lambda) = \sum_{c_i>0}(1-\lambda_i)v_i(c_i(x)) + \sum_{i\in\mathcal{E},c_i<0}(1+\lambda_i)v_i(c_i(x)) + \sum_{i\in\mathcal{I},c_i<0}\lambda_i|c_i(x)|.$$

If $E_{opt}(x^k,\lambda^k,\rho_k)=0$, $E_c(x^k,\lambda^k)=0$ and $v(x^k)=0$ with $\rho_k>0$, we know x^k is stationary for (NLP). If $E_{fea}(x,\nu)$, $E_c(x^k,\nu^k)=0$ and $v(x^k)>0$, then x^k is an infeasible stationary point.

Obviously, the KKT residual complexities depend on many factors especially the subproblem tolerance $\{\gamma_k\}$, since they represent how accurately the subproblems are solved. We make the following assumption about $\{\gamma_k\}$.

ASSUMPTION 15. *The subproblem tolerance $\{\gamma_k\}$ are selected such that $\gamma_k \le \eta k^{-\zeta/2}$ with constant $\eta>0$ and $\zeta\ge 1$.*

The parameters η and ζ control accuracy of the subproblem solution. Larger ζ or small η means more accurate subproblem solution is needed.

The following lemma establishes the relationship between the KKT residual and complementarity residual for feasibility and optimality problems.

LEMMA 16. *Under Assumption 2, 6 and 15, it holds that, for all $k\in\mathbb{N}$,*

$$E_{opt}(x^k,\lambda^k,\rho^k) \le \frac{1}{\delta\beta_\phi}\Delta l(d^k;\rho_k,x^k) + \frac{1-\beta_\phi}{\delta\beta_\phi}\gamma_k, \tag{3.20}$$

$$E_c(x^k,\lambda^k) \le \frac{1}{\beta_\phi}\Delta l(d^k;\rho_k,x^k) + \frac{1-\beta_\phi}{\beta_\phi}\gamma_k, \tag{3.21}$$

$$E_{fea}(x^k,\nu^k) \le \frac{1}{\delta\beta_v}\Delta l(d^k;0,x^k) + \frac{1-\beta_v}{\delta\beta_v}\gamma_k, \tag{3.22}$$

$$\text{and} \quad E_c(x^k,\nu^k) \le \frac{1}{\beta_\phi}\Delta l(d^k;0,x^k) + \frac{1-\beta_v}{\beta_v}\gamma_k. \tag{3.23}$$

Proof. For dual feasible λ^k

$$v(x^k) - \sum_{i\in\mathcal{E}\cup\mathcal{I}} \lambda_i^k c_i(x^k) = E_c(x^k,\lambda^k) \tag{3.24}$$

which follows from the fact that (where we temporarily use $c_i^k = c_i(x^k)$ due to space limit)

$$\begin{cases} v_i(c_i^k) - \lambda_i^k c_i^k = v_i(c_i^k) - \lambda_i^k v_i(c_i^k) = (1-\lambda_i^k)v_i(c_i^k) & \text{if } c_i^k>0, \\ v_i(c_i^k) - \lambda_i^k c_i^k = v_i(c_i^k) + \lambda_i^k v_i(c_i^k) = (1+\lambda_i^k)v_i(c_i^k) & \text{if } c_i^k<0, i\in\mathcal{E} \\ v_i(c_i^k) - \lambda_i^k c_i^k = 0 - \lambda_i^k c_i^k = \lambda_i^k|c_i^k|, & \text{if } c_i^k<0, i\in\mathcal{I}. \end{cases}$$

Then

$$l(0;\rho_k,x^k)-p(\lambda^k;\rho_k,x^k)=\delta E_{opt}(x^k,\lambda^k,\rho_k)+v(x^k)-\sum_{i\in\mathcal{E}\cup\mathcal{I}}\lambda_i^k c_i(x^k)$$
$$=\delta E_{opt}(x^k,\lambda^k,\rho_k)+E_c(x^k,\lambda^k).$$

On the other hand, it holds that

$$l(0;\rho_k,x^k)-p(\lambda^k;\rho_k,x^k)\le\frac{1}{\beta_\phi}\Delta l(d^k;\rho_k,x^k)+\frac{1-\beta_\phi}{\beta_\phi}\gamma_k$$

from (R_ϕ). Combining the above yields (3.20) and (3.21). The same argument applied to $l(0;0,x^k)-p(\lambda^k;0,x^k)$ and (R_v) proves (3.22) and (3.23). ∎

As the sequence $\{\varphi(x^k,\rho_k)\}$ has been shown in Lemma 8 to be monotonically decreasing, we can denote the initial penalty function value $\varphi^0:=\varphi(x^0,\rho_0)$ and the limit $\varphi^*:=\lim_{k\to\infty}\varphi(x^k,\rho_k)$ and derive the following complexity results for model reductions.

LEMMA 17. *Under Assumption 2, 6 and 15, for any $\epsilon>0$, the following statements hold true*

(i) It needs at most

$$\frac{\delta\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\theta_\alpha\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}$$

iterations to reach $\inf_{i=0}^k\Delta l(d^i;\rho_i,x^i)\le\epsilon$.

(ii) It needs at most

$$\max\left\{\frac{4\delta\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\theta_\alpha\beta_l^2\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2},\left[\frac{2\eta(1-\beta_l)}{\beta_l\epsilon}\right]^{\frac{2}{\zeta}}\right\}$$

iterations to reach $\inf_{i=0}^k\Delta l(d^i;0,x^i)\le\epsilon$.

Proof. For Part (i), from Lemma 8, summing up both sides of (3.8) from 0 to k gives

$$\sum_{t=0}^k[\Delta l(d^t;\rho_t,x^t)]^2\le\frac{\delta\kappa_0\kappa_1}{\theta_\alpha(1-\beta_\alpha)\beta_\alpha}[\varphi(x^0,\rho_0)-\varphi(x^{k+1},\rho_{k+1})]\ \forall k\in\mathbb{N}. \tag{3.25}$$

Therefore,

$$\inf_{i=0}^k[\Delta l(d^i;\rho_i,x^i)]^2\le\frac{\delta\kappa_0\kappa_1}{k\theta_\alpha(1-\beta_\alpha)\beta_\alpha}[\varphi(x^0,\rho_0)-\varphi^*],$$

completing the proof of (i).

It follows from (2.21) that

$$\Delta l(d^k;0,x^k)\le\tfrac{1}{\beta_l}\Delta l(d^k;\rho_k,x^k)+\tfrac{1-\beta_l}{\beta_l}\gamma_k.$$

Part (i) and Assumption 15 implies if

$$k\ge\max\left\{\frac{4\delta\kappa_0\kappa_1[\varphi(x^0,\rho_0)-\varphi^*]}{\theta_\alpha\beta_l^2\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2},\left[\frac{2\eta(1-\beta_l)}{\beta_l\epsilon}\right]^{\frac{2}{\zeta}}\right\},$$

then

$$\inf_{i=0}^{k} \tfrac{1}{\beta_l}\Delta l(d^i;\rho_i,x^i) \le \epsilon/2$$

and

$$\tfrac{1-\beta_l}{\beta_l}\gamma_k \le \epsilon/2,$$

completing the proof. ∎

Lemma 16 and 17 immediately lead to our main results.

THEOREM 18. *Under Assumption 2, 6 and 15, given $\epsilon > 0$, the following statements hold true.*

(i) It requires at most

$$\max\left\{\frac{4\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\delta\beta_\phi^2\theta_\alpha\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}, \lceil\frac{2\eta(1-\beta_\phi)}{\delta\beta_\phi}\frac{1}{\epsilon}\rceil^{\frac{2}{\zeta}}\right\}$$

iterations to reach $\inf_{i=0}^k E_{opt}(x^i,\lambda^i,\rho_i)\le\epsilon$.

(ii) It requires at most

$$\max\left\{\frac{4\delta\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\beta_\phi^2\theta_\alpha\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}, \lceil\frac{2\eta(1-\beta_\phi)}{\beta_\phi}\frac{1}{\epsilon}\rceil^{\frac{2}{\zeta}}\right\}$$

iterations to reach $\inf_{i=0}^k E_c(x^i,\lambda^i)\le\epsilon$.

(iii) It requires at most

$$\max\left\{\frac{16\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\delta\theta_\alpha\beta_v^2\beta_l\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}, \left\lceil\frac{2\eta(1-\beta_l)}{\delta\beta_v\beta_l\epsilon}\right\rceil^{\frac{2}{\zeta}}, \left\lceil\frac{2\eta(1-\beta_v)}{\delta\beta_v\epsilon}\right\rceil^{\frac{2}{\zeta}}\right\}$$

iterations to reach $\inf_{i=0}^k E_{fea}(x^i,\nu^i)\le\epsilon$.

(iv) It requires at most

$$\max\left\{\frac{16\kappa_0\kappa_1(\varphi^0-\varphi^*)}{\theta_\alpha\beta_\phi^2\beta_l\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}, \left\lceil\frac{2\eta(1-\beta_l)}{\beta_\phi\beta_l\epsilon}\right\rceil^{\frac{2}{\zeta}}, \left\lceil\frac{2\eta(1-\beta_v)}{\beta_v\epsilon}\right\rceil^{\frac{2}{\zeta}}\right\}$$

iterations to reach $\inf_{i=0}^k E_c(x^i,\nu^i)\le\epsilon$.

Proof. Part (i) can be derived by requiring $\frac{1}{\delta\beta_\phi}\inf_{i=0}^k\Delta l(d^i;\rho_i,x^i)\le\epsilon/2$ and $\frac{1-\beta_\phi}{\delta\beta_\phi}\gamma_k\le\epsilon/2$ and then combining (3.20) and Lemma 17(i).

Part (ii) can be derived by requiring $\frac{1}{\beta_\phi}\inf_{i=0}^k\Delta l(d^i;\rho_i,x^i)\le\epsilon/2$ and $\frac{1-\beta_\phi}{\beta_\phi}\gamma_k\le\epsilon/2$ and then combining (3.21) and Lemma 17(i).

Part (iii) is from (3.22), Lemma 17(ii) by requiring $\frac{1}{\delta\beta_v}\inf_{i=0}^k\Delta l(d^i;0,x^i)\le\epsilon/2$ and $\frac{1-\beta_v}{\delta\beta_v}\gamma_k\le\epsilon/2$.

Part (iv) is from (3.23), Lemma 17(ii) by requiring $\frac{1}{\beta_\phi}\inf_{i=0}^k\Delta l(d^i;0,x^i)\le\epsilon/2$ and $\frac{1-\beta_v}{\beta_v}\gamma_k\le\epsilon/2$.

3.4. Local complexity of constraint violation. We have summarized the (global) complexity of stationarity and complementarity for both feasible and infeasible cases in Theorem 18, and the dual feasibility is maintained all the time during the iteration of the algorithm. Therefore, we still need to analyze the complexity of primal feasibility when the iterates converge to an optimal solution. Notice that this

is not a concern in the infeasible case, since Theorem 18 is sufficient for the complexity of KKT residuals of the feasibility problems. Therefore, in this section, we assume that $\{x^k\}$ only has feasible limit points.

The analysis of the behavior $v(x)$ may rely on the monotonic behavior of the penalty function. However, from Corollary 14, one cannot expect $v(x)$ decreases steadily over the iterations. In early iterations, it could happen that the constraint violation continues deteriorating while the objective is improving. Instead we should focus on the local behavior of $v(x)$ around a limit point x^*. Our analysis for $v(x)$ applies to the case that $\{x^k\}$ converges to a feasible x^* where strict complementarity is satisfied.

We summarize the local complexity of constraint violation $v(x)$ of $\{x^k\}$ in the following theorem.

THEOREM 19. *Under Assumption 2, 6 and 15, suppose that* $\{(x^k, \lambda^k)\} \to (x^*, \lambda^*)$ *with* $v(x^*) = 0$ *and* $-e < \lambda^*_{\mathcal{E}} < e, 0 < \lambda^*_{\mathcal{I}} < e$. *Then for any* $0 < \tau < 1 - \|\lambda^*\|_\infty$, *there exists* $\bar{k} \in \mathbb{N}$ *such that the following statements hold true:*

(i) $v(x_k) \le E_c(x^k; \lambda^k)/\tau$ *for any* $k > \bar{k}$.

(ii) It requires at most

$$\max\Big\{\frac{4\delta\kappa_0\kappa_1[\varphi(x^{\bar{k}}, \rho_{\bar{k}}) - \varphi^*]}{\tau\beta_\phi^2\theta_\alpha\beta_\alpha(1-\beta_\alpha)}\frac{1}{\epsilon^2}, \frac{1}{\tau}\Big[\frac{2\eta(1-\beta_\phi)}{\beta_\phi}\frac{1}{\epsilon}\Big]^{\frac{2}{\zeta}}\Big\}$$

additional iterations to reach $\inf_{i=\bar{k}}^k v(x^i) \le \epsilon$ *for given* $\epsilon > 0$.

Proof. We first prove Part (i). Given $0 < \tau < 1 - \|\lambda^*\|_\infty$, there exists $\bar{k} \in \mathbb{N}$ such that for all $k \ge \bar{k}$, the following holds

$$\begin{aligned} -1 + \tau \le \lambda_i^k &\le 1 - \tau, \quad && i \in \mathcal{E} \\ 0 < \lambda_i^k &\le 1 - \tau, && i \in \mathcal{A}(x^*) \\ 0 \le \lambda_i^k &\le \tau, && i \in \mathcal{N}(x^*). \\ c_i(x^k) &\le 0, && i \in \mathcal{N}(x^*). \end{aligned}$$

Therefore,

$$\begin{aligned} v_i(c_i^k) - \lambda_i^k c_i^k &= |c_i^k| - \lambda_i^k c_i^k \ge |c_i^k| - (1-\tau)|c_i^k| \ge \tau|c_i^k|, && i \in \mathcal{E} \\ v_i(c_i^k) - \lambda_i^k c_i^k &= (c_i^k)_+ - \lambda_i^k c_i^k \ge (c_i^k)_+ - (1-\tau)(c_i^k)_+ \ge \tau(c_i^k)_+, && i \in \mathcal{A}(x^*) \\ v_i(c_i^k) - \lambda_i^k c_i^k &= (c_i^k)_+ - \lambda_i^k c_i^k = 0 - \lambda_i^k c_i^k \ge 0, && i \in \mathcal{N}(x^*). \end{aligned}$$

Hence

$$v(x^k) - \sum_{i \in \mathcal{E}\cup\mathcal{I}} \lambda_i^k c_i^k = \sum_{i \in \mathcal{E}\cup\mathcal{I}} (v_i(c_i^k) - \lambda_i^k c_i^k) \ge \tau v(x^k).$$

This, combined with (3.24), yields

$$E_c(x^k; \lambda^k) \ge \tau v(x^k)$$

completing the proof of Part (i).

Part (ii) follows naturally from Theorem 18(ii) by replacing starting point x^0 with $x^{\bar{k}}$. ∎

We emphasize that the local complexity result is derived under quite loose assumptions compared with other nonlinear optimization methods. The strictly complementary condition in Theorem 19 is commonly used in penalty-SQP methods for

analyzing the local convergence rate [11, 8]. Indeed, second-order methods (interior point methods or SQP methods) for constrained nonlinear optimization generally analyze local convergence by assuming strictly complementary condition, regular condition and second-order sufficient condition. These three conditions are also required to hold true in [17] for analyzing the local behavior of the SLP algorithm.

The other aspect to notice is about the constant ζ, which controls how fast γ_k tends to 0. The complexity results we have derived consists $O(\epsilon^{-2})$ and $O(\epsilon^{-2})$. If we choose $\zeta \geq 1$, meaning $\gamma_k \sim O(k^{-1})$, then overall we have $O(\epsilon^{-2})$ in the complexity results. On the contrary, if we want to drive γ_k to zero slower than $O(k^{-1})$, then we have $O(\epsilon^{-2/\zeta})$ in the complexity results.

4. Subproblem algorithms. In this section, we apply simplex methods to solve the subproblem by focus on ℓ_∞ norm trust region in $(\mathscr{P}')$. Since the discussion focus on the subproblem at the kth iteration, we drop x^k and the iteration number k and use the shorthand notation as introduced in § 2.2.

Using ℓ_∞ norm trust region in $(\mathscr{P}')$ results in subproblem

$$\begin{aligned} \min_{(d,r,s,t)} \ & \langle \rho g, d\rangle + \langle e, r+s\rangle + \langle e, t\rangle \\ \text{s.t. } & \langle a_i, d\rangle + b_i = r_i - s_i, \ i \in \mathcal{E}, \\ & \langle a_i, d\rangle + b_i \leq t_i, \qquad i \in \mathcal{I}, \\ & -\delta e \leq d \leq \delta e, \ (r,s,t) \geq 0. \end{aligned} \tag{4.1}$$

by adding auxiliary variables (r, s, t). To see how a primal simplex method could benefit from the structures of (4.1), we rewrite the standard form of (4.1) as

$$\begin{aligned} \min_{(d_+,d_-,r,s,t,z,u,v)} \ & \langle \rho g, d_+\rangle - \langle \rho g, d_-\rangle + \langle e, r\rangle + \langle e, s\rangle + \langle e, t\rangle \\ \text{s.t. } & \begin{bmatrix} a_{\mathcal{E}}^T & -a_{\mathcal{E}}^T & -I_{\mathcal{E}} & I_{\mathcal{E}} & 0 & 0 & 0 & 0 \\ a_{\mathcal{I}}^T & -a_{\mathcal{I}}^T & 0 & 0 & -I_{\mathcal{I}} & I_{\mathcal{I}} & 0 & 0 \\ I_n & -I_n & 0 & 0 & 0 & 0 & I & 0 \\ -I_n & I_n & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix} \begin{bmatrix} d_+ \\ d_- \\ r \\ s \\ t \\ z \\ u \\ v \end{bmatrix} = \begin{bmatrix} -b_{\mathcal{E}} \\ -b_{\mathcal{I}} \\ \delta e \\ \delta e \end{bmatrix} \\ & (d_+, d_-, r, s, t, u, v, w) \geq 0, \end{aligned} \tag{4.2}$$

by splitting d into (d_+, d_-) and adding slack variables (u, v, w). The benefits of using a simplex for solving such a linear optimization subproblem in our proposed method can be summarized as follows.

- The linear optimization subproblem (4.1) is always feasible and bounded due to the presence of slack variables and the trust region.
- There exists a basic feasible solution for the tableau

$$(d_+, d_-, r, s, t, z, u, v) = (0, 0, (b_{\mathcal{E}})_+, -(b_{\mathcal{E}})_-, (b_{\mathcal{I}})_+, -(b_{\mathcal{I}})_-, \delta e, \delta e),$$

  so that tableau can always be trivially initialized.
- During pivoting, the complementary condition is always satisfied, i.e.,

$$\begin{aligned} (1-\lambda_i) r_i = 0, \ (1+\lambda_i) s_i = 0, \ i \in \mathcal{E} \\ \lambda_i t_i = 0, \ (1-\lambda_i) z_i = 0, \ i \in \mathcal{I}, \end{aligned}$$

implies $\chi(d, \lambda) = 0$ is always true. Therefore $r_c = 1 > \beta_c$ simplifies our penalty parameter updating strategy.

- The quantities $l(d; \rho)$, $p(\lambda; \rho)$ and $v_i(\langle a_i, d\rangle + b_i)$ used for computing ratios r_ϕ, r_c and r_v can be easily extracted from the tableau. Moreover, d can be extracted easily from the last column of the tableau and λ can also be extract from the last row of the tableau.
- After reducing ρ during pivoting, it is only needed to change the row of the objective vector in the tableau. With a new ρ, the current iterate remains basic feasible, so that the simplex method can continue with a "warm-start" basic feasible initial point.

5. Numerical experiments. In this section, we list our findings for applying `FoNCO` on a collections of nonlinear problems.

5.1. Trust region radius updates. We fix the trust region radius to simplify the analysis. However, in practice, dynamically adjusting the radius helps to improve the algorithm efficiency. In our implementation, the radius is adjusted as described below. Define ratio

$$\sigma_k := \frac{\phi(x^k, \rho_k) - \phi(x^k + d^k, \rho_k)}{\Delta l(d^k; \rho_k, x^k)}.$$

The trust radius is updated as

$$\delta_{k+1} = \begin{cases} \min(2\delta_k, \delta_{\max}) & \text{if } \sigma_k > \bar{\sigma} \\ \max(\delta_k/2, \delta_{\min}) & \text{if } \sigma_k < \underline{\sigma} \\ \delta_k & \text{otherwise,} \end{cases}$$

where $0 < \underline{\sigma} < \bar{\sigma} < 1$ and $\delta_{\max} > \delta_{\min}$ are prescribed parameters.

We choose $\bar{\sigma} > \beta_\alpha$ such that the Armijo line search condition holds naturally true if $\sigma_k > \bar{\sigma}$. In this case, the back-tracking line search is skipped after solving the subproblem. We do not consider repeatedly rejecting the trust region radius and re-solving the subproblem if $\sigma_k < \underline{\sigma}$. If the trust region radius is reduced to be smaller than $\delta_{\min}$, we stop further reducing the trust region radius and continue with line search. In either case, our theoretical analysis still holds.

5.2. Implementation. Our code[1] is a prototype Python implementation using package `NumPy`. Define the relative KKT error as

$$\epsilon_{kkt} := \frac{\max(E_{opt}(x^k, \lambda^k, \rho_k), E_c(x^k, \lambda^k))}{\max(1, E_{opt}(x^0, \lambda^0, \rho_0), E_c(x^0, \lambda^0))}. \tag{5.1}$$

The algorithm is terminated if $\epsilon_{kkt} < 10^{-4}$ and constraint violation $v(x^k) < 10^{-4}$. Otherwise, the algorithm is deemed to fail within the maximum number of iterations. Denote $\texttt{Iter}^{ps}$ as the maximum number of iterations for the subproblem solver. The relaxation parameter γ_k is updated as

$$\gamma_k = \gamma_0 \theta_\gamma^{k-1}.$$

The parameter values used in our implementation are listed in the following Table 1.

[1] https://github.com/DataCorrupted/FoNCO.

TABLE 1
Parameters in `FoNCO`

Parameter	ρ_0	β_α	β_v	β_ϕ	β_l	γ_0	θ_γ	θ_α	δ_0
Value	1	10^{-4}	0.3	0.75	0.135	0.01	0.7	0.5	1
Parameter	$\bar{\sigma}$	$\underline{\sigma}$	$\delta_{\min}$	$\delta_{\max}$	`Iter`	$\texttt{Iter}^{ps}$			
Value	0.3	0.75	64	10^{-4}	1024	100			

We tested our implementation on 126 Hock–Schittkowski problems [18] on a ThinkPad T470 with i5-6700U processor. The detailed performance statistics of `FoNCO` is provided in Table 3, where column name is explained in Table 2.

We have the following observations from the experiment.

- Our algorithm solves 113 out of these 126 problems , attaining a success rate $\approx 89.7\%$. We noticed that some problems are sensitive to the selection of trust region radius. For examples, problems HS87, HS93 HS101, HS102 and HS103 failed with initial trust region radius 1. We re-ran those 5 problems with a smaller initial trust region radius $\delta_0 = 10^{-4}$. All these 5 problems are solved successfully. We believe the robustness of our proposed algorithm could be improved with a more sophisticated trust region radius updating strategy.
- Simplex method employed in our implementation is very efficient. Figure 1 shows the histogram of average number of pivots per iteration for 113 successful problems. We can see that for the majority of the cases, pivot per iteration is less than 5.

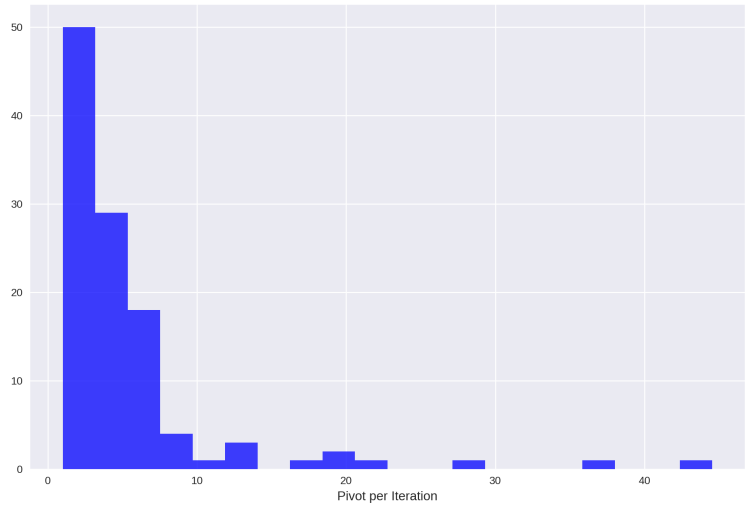

FIG. 1. *Pivots per iteration for 113 successful cases.*

- Compared with second-order methods, SLP may take more outer iterations to compute for a high accuracy solution. However, the lower computational cost of each subproblem might be able to compensate for more outer iterations.

6. Conclusion. In this paper, we have proposed, analyzed, and tested an algorithmic framework of first-order methods for solving nonlinear constrained optimization problems that possesses global convergence guarantees for both feasible and infeasible problems instances. The worst-case complexity of KKT residuals for feasible and infeasible cases have been studied as well as the local complexity for constraint violation for feasible cases.

Numerical results demonstrated that the proposed methods work on HS test problems. We remark, however, the selection of trust region radius and its updating strategy plays a key role in the robustness of the methods. It would be interesting to develop more efficient updating strategies and study how the complexity could be affected by the trust region.

TABLE 2
Column Explanation

Problem	The name of the problem
# iter	Number of iterations
# pivot	Total number of pivots
# f	Number of function evaluations
$f(x*)$	Final objective value
$v(x^*)$	Final constraint violation
KKT	Final relative KKT error defined in (5.1)
$\rho*$	Final ρ
Exit	1(Success) or -1(`Iter` exceeded)

Table 3: Numerical Experiment

Problem	# iter	# pivot	# f	$f(x^*)$	$v(x^*)$	KKT	ρ_*	Exit
HS1	260	504	440	7.490597E−02	0.0E+00	9.7E−05	1.6E−03	1
HS10	16	25	40	-1.000002E+00	4.8E−06	6.3E−06	1.0E+00	1
HS100	99	545	394	6.806299E+02	9.8E−05	6.2E−05	7.0E−01	1
HS100LNP	243	1653	1725	6.806300E+02	3.2E−05	9.8E−05	5.9E−01	1
HS100MOD	32	129	52	6.786796E+02	1.1E−05	6.9E−05	6.2E−01	1
HS101	1025	5089	8011	3.000007E+03	7.6E−03	3.6E−13	1.0E−04	-1
HS102	66	311	89	1.000775E+03	4.1E−06	7.4E−05	1.0E−04	1
HS103	71	382	100	7.201554E+02	6.2E−05	9.8E−05	4.3E−05	1
HS104	5	27	6	4.200000E+00	8.0E−08	8.8E−16	1.0E+00	1
HS105	244	1904	1045	1.044612E+03	0.0E+00	1.4E−05	2.1E−06	1
HS106	1025	15683	1117	2.146195E+03	1.1E+00	1.1E−03	7.2E−04	-1
HS107	1025	7191	1078	2.713610E+06	5.2E+00	2.3E+05	2.6E−10	-1
HS108	32	1198	135	-8.660254E−01	2.0E−07	6.1E−05	9.0E−01	1
HS109	1025	4727	1049	0.000000E+00	5.4E+03	1.0E+00	1.0E+00	-1
HS11	26	60	60	-8.498486E+00	7.2E−06	4.2E−05	2.6E−01	1
HS110	23	170	92	-4.577848E+01	0.0E+00	6.4E−05	1.0E+00	1
HS111	298	3823	515	-4.776118E+01	1.1E−05	8.4E−05	5.4E−02	1
HS111LNP	298	3823	515	-4.776118E+01	1.1E−05	8.4E−05	5.4E−02	1
HS112	54	691	216	-4.776109E+01	8.1E−16	6.3E−05	1.3E−03	1
HS113	34	338	74	2.430622E+01	1.5E−05	8.8E−05	2.8E−01	1
HS114	1025	25291	1073	-1.636123E+03	0.0E+00	1.0E+00	3.3E−47	-1
HS116	10	77	12	2.500000E+02	9.0E−07	3.6E−15	1.0E+00	1
HS117	352	8011	466	3.234873E+01	0.0E+00	8.2E−05	3.6E−04	1
HS118	18	319	19	9.329922E+02	0.0E+00	1.9E−16	7.4E−02	1
HS119	16	436	20	2.449598E+02	1.9E−15	7.3E−05	1.5E−01	1
HS12	10	20	15	-3.000000E+01	4.6E−10	7.7E−06	1.0E+00	1
HS13	15	26	16	4.000000E+00	0.0E+00	8.5E−05	1.0E−04	1
HS14	9	19	16	1.393465E+00	1.0E−14	3.0E−07	4.0E−01	1
HS15	78	167	132	3.065000E+02	0.0E+00	3.2E−17	1.0E−03	1
HS16	35	79	77	2.314466E+01	0.0E+00	1.9E−08	2.8E−02	1
HS17	17	42	38	1.000000E+00	6.1E−21	8.4E−08	4.7E−01	1
HS18	13	26	34	5.000000E+00	0.0E+00	9.4E−06	1.0E+00	1

Continued on next page

Table 3 – continued from previous page

Problem	# iter	# pivot	# f	$f(x^*)$	$v(x^*)$	KKT	ρ_*	Exit
HS19	173	359	189	-6.961814E+03	0.0E+00	1.9E−08	9.6E−05	1
HS2	15	26	54	4.941229E+00	0.0E+00	3.1E−05	9.0E−01	1
HS20	40	88	314	4.019873E+01	0.0E+00	2.3E−07	8.1E−03	1
HS21	5	7	6	-9.996000E+01	0.0E+00	0.0E+00	1.0E+00	1
HS21MOD	15	23	19	-9.596000E+01	0.0E+00	0.0E+00	1.7E−01	1
HS22	5	10	6	1.000000E+00	0.0E+00	3.7E−09	1.0E+00	1
HS23	18	37	225	2.000000E+00	0.0E+00	1.0E−07	3.1E−01	1
HS24	3	6	38	-1.000000E+00	0.0E+00	1.9E−16	1.0E+00	1
HS25	1	0	1	3.283500E+01	0.0E+00	1.9E−08	1.0E+00	1
HS26	86	254	415	8.505871E−06	5.4E−06	9.8E−05	7.2E−01	1
HS268	249	1259	827	3.075321E+00	0.0E+00	9.6E−05	1.1E−04	1
HS27	16	33	22	4.000000E−02	0.0E+00	0.0E+00	3.1E−01	1
HS28	6	15	9	0.000000E+00	0.0E+00	0.0E+00	9.0E−01	1
HS29	360	1230	2659	-2.262742E+01	8.2E−06	7.7E−05	1.0E+00	1
HS3	518	1019	519	2.490010E−04	0.0E+00	9.9E−05	1.0E−04	1
HS30	7	20	8	1.000061E+00	0.0E+00	3.0E−05	1.0E+00	1
HS31	129	315	599	5.999992E+00	1.3E−06	5.2E−05	1.0E−01	1
HS32	11	47	17	1.000000E+00	0.0E+00	9.2E−17	2.0E−01	1
HS33	291	589	292	-4.000000E+00	0.0E+00	2.6E−07	5.8E−09	1
HS34	11	50	12	-8.340324E−01	4.5E−10	4.2E−06	1.0E+00	1
HS35	18	42	87	1.111111E−01	0.0E+00	1.3E−05	6.5E−01	1
HS35I	18	42	87	1.111111E−01	0.0E+00	1.3E−05	6.5E−01	1
HS35MOD	2	5	3	2.500000E−01	0.0E+00	0.0E+00	1.0E+00	1
HS36	376	1093	377	-3.300000E+03	0.0E+00	2.8E−16	2.0E−08	1
HS37	389	1136	408	-3.456000E+03	0.0E+00	1.9E−05	1.5E−08	1
HS38	44	139	78	1.070841E−02	0.0E+00	4.8E−05	7.8E−03	1
HS39	28	83	46	-1.000044E+00	4.3E−05	8.5E−06	8.1E−01	1
HS3MOD	28	33	29	0.000000E+00	0.0E+00	0.0E+00	5.9E−01	1
HS4	9	23	16	2.666667E+00	0.0E+00	0.0E+00	1.8E−01	1
HS40	41	1826	174	-2.500000E−01	1.6E−09	8.6E−05	1.5E−01	1
HS41	176	636	215	1.925926E+00	0.0E+00	5.8E−05	1.0E−07	1
HS42	20	62	50	1.385786E+01	1.6E−07	5.1E−05	3.1E−01	1
HS43	24	93	48	-4.400000E+01	1.5E−08	5.6E−05	4.5E−01	1
HS44	98	398	99	-1.500000E+01	0.0E+00	4.4E−16	1.0E−04	1
HS44NEW	91	380	92	-1.500000E+01	0.0E+00	0.0E+00	1.1E−04	1
HS45	4	17	5	1.000000E+00	0.0E+00	0.0E+00	1.0E+00	1
HS46	90	470	248	1.987922E−05	4.8E−06	8.8E−05	1.0E+00	1
HS47	41	247	80	2.842654E−05	3.6E−05	8.2E−05	8.1E−01	1
HS48	4	16	6	1.109336E−31	0.0E+00	3.7E−17	1.0E+00	1
HS49	20	116	27	4.978240E−03	0.0E+00	8.1E−05	2.8E−01	1
HS5	8	12	39	-1.913223E+00	0.0E+00	3.8E−05	1.0E+00	1
HS50	10	53	17	1.232595E−32	4.4E−16	7.0E−19	1.8E−01	1
HS51	13	59	23	2.170139E−08	0.0E+00	8.3E−05	7.2E−01	1
HS52	37	213	109	5.326649E+00	1.6E−16	9.0E−05	5.0E−02	1
HS53	22	110	45	4.093023E+00	1.1E−16	6.4E−05	1.2E−01	1
HS54	5	8	37	-1.539517E−01	0.0E+00	1.4E−06	1.0E+00	1
HS55	1	7	2	6.666667E+00	1.1E−16	3.7E−17	1.0E+00	1
HS56	2	5	3	-1.000000E+00	1.2E−15	2.7E−05	1.0E−04	1
HS57	7	7	61	3.064631E−02	0.0E+00	9.6E−05	1.0E+00	1
HS59	9	13	10	3.012922E+01	1.4E−07	9.5E−06	1.0E+00	1
HS6	314	789	1710	1.503772E−13	7.7E−06	8.5E−05	9.0E−01	1
HS60	117	397	834	3.256821E−02	2.0E−09	9.8E−05	1.0E+00	1
HS61	18	60	34	-1.436462E+02	5.3E−05	7.2E−05	4.9E−01	1
HS62	258	745	1395	-2.627251E+04	1.6E−16	8.4E−05	3.1E−05	1
HS63	16	54	46	9.617152E+02	2.7E−06	1.7E−05	5.5E−01	1
HS64	43	101	51	6.299779E+03	3.4E−05	1.8E−06	4.9E−02	1
HS65	23	59	30	9.535289E−01	1.7E−10	4.3E−05	1.0E+00	1
HS66	23	61	84	5.181632E−01	2.4E−07	7.0E−05	5.3E−01	1
HS67	1025	3071	1034	-9.162074E+02	0.0E+00	7.3E−02	1.5E−04	-1
HS68	38	138	389	2.400000E−05	0.0E+00	4.2E−14	1.7E−03	1
HS69	62	281	505	-9.280357E+02	1.1E−10	4.4E−05	2.8E−08	1
HS7	17	32	51	-1.732051E+00	3.0E−09	4.7E−05	9.0E−01	1
HS70	1025	4056	2799	1.866660E−02	0.0E+00	1.7E−02	9.6E−03	-1
HS71	28	529	237	1.701402E+01	3.9E−08	9.1E−05	3.9E−01	1
HS72	99	333	104	7.276793E+02	4.4E−09	7.0E−05	1.7E−05	1
HS73	31	272	33	2.989438E+01	1.5E−07	2.3E−06	3.7E−02	1
HS74	51	200	64	5.126498E+03	1.4E−06	4.6E−05	1.3E−01	1
HS75	1025	6031	1042	5.127004E+03	3.1E−02	4.9E−05	2.4E−03	-1
HS76	23	97	68	-4.681818E+00	0.0E+00	2.6E−05	3.9E−01	1
HS76I	23	97	68	-4.681818E+00	0.0E+00	2.6E−05	3.9E−01	1
HS77	49	245	141	2.415043E−01	1.8E−05	7.2E−05	1.0E+00	1

Continued on next page

Table 3 – continued from previous page

Problem	# iter	# pivot	# f	$f(x^*)$	$v(x^*)$	KKT	ρ_*	Exit
HS78	19	111	73	-2.919700E+00	2.7E−07	4.7E−05	8.1E−01	1
HS79	226	1653	1181	7.877686E−02	2.9E−06	8.7E−05	1.0E+00	1
HS8	5	10	6	-1.000000E+00	6.4E−07	0.0E+00	1.0E+00	1
HS80	136	864	1662	5.394985E−02	2.6E−09	3.7E−05	1.0E+00	1
HS81	69	497	294	5.394986E−02	2.0E−05	9.5E−05	8.1E−01	1
HS83	1025	99668	34494	-2.539096E+04	2.5E+00	1.8E+45	9.1E−49	-1
HS84	1025	100970	1083	-2.325944E+09	3.8E+05	1.7E+45	2.0E−47	-1
HS85	1025	5148	3315	4.374488E+01	9.3E+06	1.0E+00	5.7E−04	-1
HS86	25	146	64	-3.234868E+01	2.2E−16	9.8E−05	5.6E−02	1
HS87	15	52	16	8.997184E+03	1.6E−09	2.3E−07	1.0E−04	1
HS88	59	98	163	1.349683E+00	1.2E−05	8.3E−05	1.0E−04	1
HS89	187	545	778	1.357072E+00	5.4E−06	9.1E−05	1.0E−04	1
HS9	4	10	6	-5.000000E−01	0.0E+00	8.5E−09	1.0E+00	1
HS90	679	3485	3923	1.385570E+00	5.0E−06	9.9E−05	1.0E−04	1
HS91	525	2535	2481	1.357178E+00	5.2E−06	7.9E−05	1.0E−04	1
HS92	421	2645	1628	1.349981E+00	1.2E−05	7.5E−05	1.0E−04	1
HS93	1025	7846	6127	1.353296E+02	1.9E−06	3.2E−03	2.2E−05	-1
HS95	33	206	62	1.561953E−02	0.0E+00	2.7E−17	1.0E−02	1
HS96	30	206	64	1.561953E−02	0.0E+00	1.0E−16	1.0E−02	1
HS97	53	1007	81	4.071246E+00	0.0E+00	7.2E−15	7.0E−04	1
HS98	36	303	102	3.135809E+00	0.0E+00	1.8E−15	1.1E−03	1
HS99	57	348	462	-8.310799E+08	1.0E−11	6.2E−05	3.8E−01	1
HS99EXP	1025	32423	1025	0.000000E+00	5.2E+03	1.2E+00	1.0E+00	-1

REFERENCES

[1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[2] Thomas E Baker and Leon S Lasdon. Successive linear programming at exxon. *Management science*, 31(3):264–274, 1985.

[3] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[4] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[5] Léon Bottou. Stochastic learning. In *Advanced lectures on machine learning*, pages 146–168. Springer, 2004.

[6] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[7] Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer, 2012.

[8] James V Burke, Frank E Curtis, and Hao Wang. A sequential quadratic optimization algorithm with rapid infeasibility detection. *SIAM Journal on Optimization*, 24(2):839–872, 2014.

[9] James V Burke, Frank E Curtis, Hao Wang, and Jiashan Wang. A dynamic penalty parameter updating strategy for matrix-free sequential quadratic optimization. *arXiv preprint arXiv:1803.09224*, 2018.

[10] Richard H Byrd, Frank E Curtis, and Jorge Nocedal. An inexact sqp method for equality constrained optimization. *SIAM Journal on Optimization*, 19(1):351–369, 2008.

[11] Richard H Byrd, Frank E Curtis, and Jorge Nocedal. Infeasibility detection and sqp methods for nonlinear optimization. *SIAM Journal on Optimization*, 20(5):2281–2299, 2010.

[12] Richard H Byrd, Nicholas IM Gould, Jorge Nocedal, and Richard A Waltz. An algorithm for nonlinear optimization using linear programming and equality constrained subproblems. *Mathematical Programming*, 100(1):27–48, 2003.

[13] Richard H Byrd, Jorge Nocedal, and Richard A Waltz. Steering exact penalty methods for nonlinear programming. *Optimization Methods and Software*, 23(2):197–213, 2008.

[14] Frank H Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.

[15] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, 57(11):1413–1457, 2004.

[16] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[17] Roger Fletcher and E. Sainz de la Maza. Nonlinear programming and nonsmooth optimization by successive linear programming. *Mathematal Programmins*, 43:235–256, 1989.

[18] Willi Hock and Klaus Schittkowski. Test examples for nonlinear programming codes. *Journal of Optimization Theory and Applications*, 30(1):127–129, 1980.

[19] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

[20] Anatoli Juditsky, Arkadi Nemirovski, and Claire Tauvel. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems*, 1(1):17–58, 2011.

[21] William Karush. Minima of functions of several variables with inequalities as side conditions. In *Traces and Emergence of Nonlinear Programming*, pages 217–245. Springer, 2014.

[22] Harold W Kuhn and Albert W Tucker. Nonlinear programming. In *Traces and emergence of nonlinear programming*, pages 247–258. Springer, 2014.

[23] LS Lasdon, AD Waren, S Sarkar, and F Palacios. Solving the pooling problem using generalized reduced gradient and successive linear programming algorithms. *ACM Sigmap Bulletin*, (27):9–15, 1979.

[24] Ronny Luss and Marc Teboulle. Conditional gradient algorithmsfor rank-one matrix approximations with a sparsity constraint. *SIAM Review*, 55(1):65–98, 2013.

[25] Mikhail V. Solodov. Constraint qualifications. *Wiley Encyclopedia of Operations Research and Management Science*, 2010.

[26] Constantin Udriste. *Convex functions and optimization methods on Riemannian manifolds*, volume 297. Springer Science & Business Media, 1994.

[27] Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.