INDUSTRIAL AND SYSTEMS ENGINEERING

ISE

Single-Loop Deterministic and Stochastic Interior-Point Algorithms for Nonlinearly Constrained Optimization

Frank E. Curtis, Xin Jiang, and Qi Wang

Department of Industrial and Systems Engineering, Lehigh University

LEHIGH
UNIVERSITY.

COR@L
COMPUTATIONAL OPTIMIZATION
RESEARCH AT LEHIGH

Single-Loop Deterministic and Stochastic Interior-Point Algorithms for Nonlinearly Constrained Optimization

Frank E. Curtis[*1], Xin Jiang[†1], and Qi Wang[‡1]

[1]Department of Industrial and Systems Engineering, Lehigh University

August 28, 2024

Abstract

An interior-point algorithm framework is proposed, analyzed, and tested for solving nonlinearly constrained continuous optimization problems. The main setting of interest is when the objective and constraint functions may be nonlinear and/or nonconvex, and when constraint values and derivatives are tractable to compute, but objective function values and derivatives can only be estimated. The algorithm is intended primarily for a setting that is similar for stochastic-gradient methods for unconstrained optimization, namely, the setting when stochastic-gradient estimates are available and employed in place of gradients of the objective, and when no objective function values (nor estimates of them) are employed. This is achieved by the interior-point framework having a single-loop structure rather than the nested-loop structure that is typical of contemporary interior-point methods. For completeness, convergence guarantees for the framework are provided both for deterministic and stochastic settings. Numerical experiments show that the algorithm yields good performance on a large set of test problems.

1 Introduction

The interior-point methodology is one of the most popular and effective derivative-based algorithm methodologies for solving inequality-constrained continuous optimization problems. Indeed, while there has been a long history of development of other types of algorithms—e.g., of the augmented-Lagrangian and sequential-quadratic-optimization varieties [16]—interior-point methods are arguably the most effective class of approaches for solving many constrained continuous optimization problems. In the setting of nonconvex optimization, which is the setting of interest in this paper, the effectiveness of interior-point methods is evidenced by the fact that the most popular software packages for solving such problems are based on interior-point methods. These include the packages `Ipopt` [19], `Knitro` [5], and `LOQO` [18]. This popularity withstands despite the fact that interior-point methods for solving nonconvex problems do not necessarily achieve the same optimal worst-case complexity properties that can be achieved in convex settings.

Despite its widespread success, little work has been done on extending the interior-point methodology to the stochastic regime. Here, we are focused on the setting in which constraint values and derivatives are tractable to compute, but objective values and derivatives are not. Furthermore, like in the setting of stochastic-gradient-based methods for solving unconstrained problems, our focus is to have an interior-point method that employs stochastic-gradient estimates (in place of exact objective gradients) and no objective function evaluations. Given the success of interior-point methods in the deterministic regime, there is ample

---

[*]E-mail: frank.e.curtis@lehigh.edu

[†]E-mail: xjiang@lehigh.edu

[‡]E-mail: qiw420@lehigh.edu

motivation to explore their potential theoretical and practical behavior in such a stochastic regime. This is the main motivation for the work presented in this paper.

We contend that there are at least a few significant obstacles to the design and analysis of a stochastic interior-point method, which may be at least part of the reason that there has been little prior work on this topic. Firstly, interior-point methods often have a nested-loop structure wherein an inner loop is designed to solve (approximately) a so-called barrier subproblem for a fixed value of a barrier parameter and the outer loop involves decreasing the barrier parameter. Determining when to terminate the inner loop typically involves evaluating a measure of stationarity. However, in the aforementioned stochastic setting, evaluating such a measure is not tractable since objective-gradient evaluations are not tractable. Secondly, the barrier functions introduced in interior-point methods have corresponding gradient functions that are not Lipschitz continuous nor bounded. This presents major difficulties since for many stochastic-gradient-based methods, Lipschitz continuity and boundedness of the gradients are necessary features for ensuring convergence guarantees. Thirdly, interior-point methods regularly involve globalization mechanisms such as line searches or trust-region step-acceptance mechanisms that require exact function evaluations, but these are not tractable to perform in the aforementioned stochastic setting that is of interest here.

1.1 Contributions and Limitations

To provide a significant advancement in the design and analysis of stochastic-gradient-based interior-point methods, in this paper we propose and analyze a single-loop interior-point algorithm framework that offers rigorous convergence guarantees in both deterministic and stochastic settings. Let us emphasize upfront that we do not expect our method to be competitive with state-of-the-art interior-point methods in the deterministic regime, and especially not with those that employ second-order derivatives of the objective. Nevertheless, we present an algorithm and analyze its convergence properties for that setting for completeness and since it helps to elucidate the essential features of an algorithm that are necessary for achieving convergence guarantees in the stochastic regime.

Our method is an extension of the approach proposed in [7], which was proposed and analyzed for the bound-constrained setting. We go beyond the bound-constrained setting and propose a framework that is applicable for problems with nonlinear inequality constraints, at least as long as a strictly feasible initial point for the algorithm can be provided. For the sake of generality, we present the framework in a manner such that linear equality constraints can also be present, although we emphasize that our convergence guarantees in neither the deterministic nor stochastic regime allow for the presence of nonlinear equality constraints. Along with our concluding remarks, we discuss the challenges that would be faced when trying to extend our approach to settings with nonlinear equality constraints.

Most of the assumptions that are required for our convergence guarantees are standard in the literature on interior-point algorithms, specifically for that on so-called feasible interior-point methods that maintain feasibility (with respect to all of the constraints) at all algorithm iterates. However, we require one additional assumption that is not standard. This assumption essentially requires that, when an algorithm iterate is close to a constraint boundary, a direction of sufficient descent can be computed that moves sufficiently away from the boundary. It is, in essence, an assumption that combines two aspects: (a) a type of nondegeneracy assumption and (b) an assumption that the barrier parameter is initialized to be sufficiently large compared to a neighborhood parameter that is introduced in our algorithm. We demonstrate these aspects through some illustrative examples. Since, to the best of our knowledge, there are no other interior-point methods in the literature that possess convergence guarantees in a stochastic setting that is comparable to ours, we contend that our algorithm and analysis together constitute a notable contribution to the literature despite our method being a feasible interior-point method and our need to introduce a nonstandard assumption. Future research may reveal techniques that can offer convergence guarantees for a stochastic-gradient-based interior-point method in other settings.

We also provide the results of numerical experiments showing that our algorithms performs well when employed to solve challenging test problems.

1.2 Notation

We use $\mathbb{R}$ to denote the set of real numbers and, given $a \in \mathbb{R}$, we use $\mathbb{R}_{\geq a}$ (resp., $\mathbb{R}_{>a}$) to denote the set of real numbers that are greater than or equal to a (resp., strictly greater than a). Superscripts are used to indicate the dimension of a vector or matrix whose elements are restricted to such a set; e.g., the set of n-dimensional vectors is denoted as $\mathbb{R}^n$. The set of positive integers is denoted as $\mathbb{N} := \{1, 2, \dots\}$ and, given $n \in \mathbb{N}$, the set of positive integers up to n is denoted as $[n] := \{1, \dots, n\}$. The set of n-by-n-dimensional symmetric positive semidefinite (resp., definite) matrices is denoted as $\mathbb{S}^n_{\succeq 0} \subset \mathbb{R}^{n \times n}$ (resp., $\mathbb{S}^n_{\succ 0} \subset \mathbb{R}^{n \times n}$).

A real-valued sequence is always introduced as $\{v_k\} \subset \mathcal{R}$, where $\mathcal{R}$ is a real vector (sub)space and the subset notation indicates that $v_k \in \mathcal{R}$ for each index $k \in \mathbb{N}$ (although elements in the sequence may repeat). Here, subscripts are used to indicate index number in a sequence, although in some situations a subscript is used to indicate an element number of a vector or vector-valued function. When indicating the ith element of a vector v_k, the notation $[v_k]_i$ is used. In all cases, the meaning of a subscript is clear from the context. The notation $\{v_k\} \to v$ indicates that the sequence $\{v_k\}$ converges to v. The notation $\{v_k\} \searrow 0$ indicates that $\{v_k\}$ is a sequence of positive real numbers that vanishes monotonically.

Given a vector $v \in \mathbb{R}^n$, we use $\mathrm{diag}(v)$ to denote the diagonal matrix for which, for all $i \in [n]$, the (i, i) element is $v_i \in \mathbb{R}$. Given two vectors $z \in \mathbb{R}^m$ and $c \in \mathbb{R}^m$, their Hadamard (i.e., component-wise) product is denoted as $z \circ c \in \mathbb{R}^m$. We use $\mathbb{1}$ to denote a vector whose elements are all equal to one; the length of such a vector is determined by the context in which it appears. Given $A \in \mathbb{R}^{l \times n}$, the null space of A is denoted as $\mathrm{Null}(A)$. Given $(A, B) \in \mathbb{S}^n_{\succeq 0} \times \mathbb{S}^n_{\succeq 0}$, we use $A \succeq B$ (resp., $A \preceq B$) to indicate that $A - B$ (resp., $B - A$) is positive semidefinite. Given $v \in \mathbb{R}^p$ and $M \in \mathbb{S}^p_{\succ 0}$, the M-norm of v is written as $\|v\|_M = \sqrt{v^T M v}$. Finally, given a set $\mathcal{P}$, its interior is denoted as $\mathrm{int}(\mathcal{P})$ and its polar cone is denoted as

$$\mathcal{P}^\circ := \{y \in \mathbb{R}^n : p^T y \leq 0 \text{ for all } p \in \mathcal{P}\}.$$

1.3 Outline

In §2, we present our problem of interest and preliminary commentary. In §3, we present our proposed framework and prove basic facts about its behavior. Section 4 contains our assumptions and convergence analyses for the deterministic and stochastic settings. As previously mentioned, our analyses rely on an assumption that is not standard; we discuss this assumption in detail in §5, highlighting certain settings of interest in which it indeed holds. Details about a practical implementation of our algorithm and resulting numerical experiments are presented in §6. Concluding remarks are provided in §7.

2 Problem Formulation and Preliminaries

Our proposed algorithm framework (Algorithm 1 on page 7) is designed to solve nonlinearly constrained optimization problems of the form

$$\min_{x \in \mathbb{R}^n} \ f(x) \text{ subject to } Ax = b \text{ and } c(x) \leq 0, \tag{1}$$

where $A \in \mathbb{R}^{l \times n}$ (with $l < n$), $b \in \mathbb{R}^l$, and the functions f and c satisfy the following assumption with respect to the feasible region of problem (1), namely,

$$\mathcal{F} := \mathcal{E} \cap \mathcal{C}_{\leq 0}, \text{ where } \begin{cases} \mathcal{E} := \{x \in \mathbb{R}^n : Ax = b\}; \\ \mathcal{C}_{\leq 0} := \{x \in \mathbb{R}^n : c(x) \leq 0\}. \end{cases}$$

Assumption 2.1. *There exists an open set $\overline{\mathcal{F}} \supset \mathcal{F}$ over which the objective function $f : \overline{\mathcal{F}} \to \mathbb{R}$ is continuously differentiable and bounded below by $f_{\inf} \in \mathbb{R}$ and the constraint function $c : \overline{\mathcal{F}} \to \mathbb{R}^m$ is continuously differentiable. In addition, the matrix $A \in \mathbb{R}^{l \times n}$ has full row rank. Finally, there exists $x \in \mathcal{F}$ such that $c(x) < 0$ (i.e., where the inequality holds strictly, component-wise).*

4

The elements of Assumption 2.1 pertaining to f and c are mostly common for the literature on smooth nonlinearly constrained optimization. The only exception is the assumption that there exists $x \in \mathcal{F}$ such that $c(x) < 0$. This aspect of the assumption precludes the presence of nonlinear equality constraints (through two-sided inequalities). It is needed in our setting since our algorithm requires a feasible initial point that is strictly feasible with respect to the nonaffine constraints.

Assumption 2.1 does not require that the feasible region $\mathcal{F}$ is bounded, although our convergence analysis in §4 requires that the iterates remain within a set such that c remains component-wise bounded below, and that the functions f, c, $\nabla f : \overline{\mathcal{F}} \to \mathbb{R}^n$, and $\nabla c_i : \overline{\mathcal{F}} \to \mathbb{R}^n$ for all $i \in [m]$ are Lipschitz continuous; see Assumption 4.1 on page 7. Our analysis could be extended to the setting in which the matrix A does not have full row rank as long as the linear system $Ax = b$ is consistent. Our algorithm framework requires that the initial iterate is contained in $\mathcal{E}$ and that each step lies in the null space of A, which inductively implies that each iterate lies in $\mathcal{E}$. These requirements remain reasonable when A has linearly dependent rows as long as a projection operator onto the null space of A is available. Merely for ease of notation, we assume A has full row rank.

In fact, our algorithm framework ensures that all iterates remain in the set

$$\mathcal{F}_{<0} := \mathcal{E} \cap \mathcal{C}_{<0}, \text{ where } \mathcal{C}_{<0} := \{x \in \mathbb{R}^n : c(x) < 0\}. \tag{2}$$

Observe that this set is guaranteed to be nonempty under Assumption 2.1. In each iteration, our framework involves the computation of a search direction that, for a given barrier parameter $\mu \in \mathbb{R}_{>0}$, aims to reduce the value of the objective augmented with the logarithmic barrier function defined with respect to the inequality-constraint functions of (1), namely, the function $\phi(\cdot, \mu) : \mathcal{C}_{<0} \to \mathbb{R}$ defined by

$$\phi(x, \mu) = f(x) - \mu \sum_{i \in [m]} \log(-c_i(x)).$$

Going forward, we refer to this as the *barrier-augmented objective function*. Observe that if x^μ is a local solution of the problem to minimize $\phi(\cdot, \mu)$ over $\mathcal{F}_{<0} = \mathcal{E} \cap \mathcal{C}_{<0}$, then there exists a Lagrange multiplier vector $y^\mu \in \mathbb{R}^l$ such that, along with $z_i^\mu := -\mu(c_i(x^\mu))^{-1}$ for all $i \in [m]$, one finds that

$$\nabla f(x^\mu) + A^T y^\mu + \nabla c(x^\mu) z^\mu = 0, \ Ax^\mu = b, \ c(x^\mu) < 0, \ z^\mu > 0, \text{ and } -z^\mu \circ c(x^\mu) = \mu \mathbb{1}.$$

Any tuple (x^μ, y^μ, z^μ) satisfying this system is referred to as a stationary point for the problem to minimize $\phi(\cdot, \mu)$ over $\mathcal{F}_{<0}$. This system should be compared to the Karush-Kuhn-Tucker (KKT) conditions for problem (1); in particular, under a constraint qualification—such as the Mangasarian-Fromovitz constraint qualification (MFCQ) [13]—if x is a local solution of problem (1), then there exists $y \in \mathbb{R}^l$ and $z \in \mathbb{R}^m$ such that the following system of KKT conditions is satisfied:

$$\nabla f(x) + A^T y + \nabla c(x) z = 0, \ Ax = b, \ c(x) \leq 0, \ z \geq 0, \text{ and } -z \circ c(x) = 0. \tag{3}$$

Any tuple (x, y, z) satisfying this system is referred to as a stationary point (or KKT point) for problem (1). As is generally the case for an interior-point method, our algorithm framework aims to find such a KKT point by taking steps to reduce the barrier-augmented objective function for a diminishing sequence of barrier parameters—a procedure that can be motivated by the fact that the stationarity conditions for the minimization of $\phi(\cdot, \mu)$ over $\mathcal{F}_{<0}$ tend toward those of (1).

3 Algorithm

Our algorithm framework is stated as Algorithm 1 on page 7. The framework presumes that it is given an initial neighborhood parameter $\theta_0 \in \mathbb{R}_{>0}$ and an initial iterate $x_1 \in \mathcal{E} \cap \mathcal{N}(\theta_0)$, where for all $\theta \in \mathbb{R}_{>0}$ we define

$$\mathcal{N}(\theta) := \{x \in \mathbb{R}^n : c(x) \leq -\theta \mathbb{1}\} \subseteq \mathcal{C}_{<0}.$$

Observe that the existence of such θ_0 and x_1 is guaranteed under Assumption 2.1. Moreover, the steps of Algorithm 1 ensure that, for a prescribed, monotonically decreasing sequence $\{\theta_k\} \subset \mathbb{R}_{>0}$ with $\theta_1 < \theta_0$, one finds $x_k \in \mathcal{E} \cap \mathcal{N}(\theta_{k-1})$ for all $k \in \mathbb{N}$. Hence, to describe the main steps of the algorithm for each $k \in \mathbb{N}$, we may presume that for all $k \in \mathbb{N}$ the algorithm has an iterate $x_k \in \mathcal{E} \cap \mathcal{N}(\theta_{k-1})$.

Following [7], Algorithm 1 also employs a prescribed, monotonically decreasing sequence of barrier parameters $\{\mu_k\} \subset \mathbb{R}_{>0}$ such that $\{\mu_k/\theta_{k-1}\}$ is a constant sequence. Other inputs required by Algorithm 1 in the deterministic setting are $\eta \in (\theta_0/\mu_1, 1)$, $\underline{\eta} \in (\theta_0, \infty)$, $\underline{\zeta} \in \mathbb{R}_{>0}$, $\overline{\zeta} \in \mathbb{R}_{>0}$ with $\underline{\zeta} \leq \overline{\zeta}$, and $\zeta \in (0, 1)$, each of which influences the search direction computation. Given these inputs along with $k \in \mathbb{N}$ and $x_k \in \mathcal{E} \cap \mathcal{N}(\theta_{k-1})$, the computation proceeds as follows. Firstly, a set of indices of constraints that are *nearly active* with respect to x_k is identified as

$$\mathcal{A}_k := \{i \in [m] : -\eta\mu_k < c_i(x_k)\}.$$

Observe that since $x_k \in \mathcal{N}(\theta_{k-1})$ it follows that $c_i(x_k) \leq -\theta_{k-1}$ for all $i \in [m]$, and, since the facts that $\eta \in (\theta_0/\mu_1, 1)$ and $\{\mu_k/\theta_{k-1}\}$ is a constant sequence together ensure $-\eta\mu_k < -\theta_{k-1}$, it follows that $-\eta\mu_k < c_i(x_k) \leq -\theta_{k-1}$ for all $i \in \mathcal{A}_k$. Secondly, a vector $q_k \in \mathbb{R}^n$ is computed, the value of which is distinct between the deterministic and stochastic settings. In particular, the framework employs

$$q_k := \begin{cases} \nabla f(x_k) - \mu_k \nabla c(x_k) \operatorname{diag}(c(x_k))^{-1}\mathbb{1} & \text{(deterministic)} \\ g_k - \mu_k \nabla c(x_k) \operatorname{diag}(c(x_k))^{-1}\mathbb{1} & \text{(stochastic)} \end{cases} \tag{4}$$

where in the stochastic setting the vector g_k is a stochastic estimate of $\nabla f(x_k)$. Third, with $P := I - A^T(AA^T)^{-1}A$ denoting the orthogonal projection matrix onto the null space of A, a search direction $d_k \in \mathbb{R}^n$ is computed such that

$$Ad_k = 0, \tag{5a}$$

$$\underline{\zeta}\|Pq_k\|_2 \leq \|d_k\|_2, \tag{5b}$$

$$\overline{\zeta}\|Pq_k\|_2 \geq \|d_k\|_2, \tag{5c}$$

$$-(Pq_k)^T d_k \geq \zeta\|Pq_k\|_2\|d_k\|_2, \text{ and} \tag{5d}$$

$$\nabla c_i(x_k)^T d_k \leq -\tfrac{1}{2}\underline{\eta}\|d_k\|_2 \text{ for all } i \in \mathcal{A}_k. \tag{5e}$$

In the deterministic setting, all that is required of the search direction d_k is that it satisfies (5). For the stochastic setting, we introduce in our analysis a particular strategy for computing the search direction that implies that (5) holds along with other useful properties for our analysis for that setting.

Let us now discuss (5) in detail. Firstly, condition (5a) requires that the search direction satisfies $d_k \in \operatorname{Null}(A)$, i.e., it lies in the null space of A, which, since $x_k \in \mathcal{E}$, ensures that the subsequent iterate will also lie in $\mathcal{E}$. Secondly, conditions (5b) and (5c) require that the norm of the search direction d_k is proportional to the norm of $Pq_k = (I - A^T(AA^T)^{-1}A)q_k$, i.e., the norm of the projection of q_k onto $\operatorname{Null}(A)$. Thus, conditions (5b) and (5c) require that the norm of the search direction is proportional to that of a projected gradient (estimate), which is a common type of requirement for a search direction in an algorithm for minimizing a smooth objective function over an affine set. Thirdly, condition (5d) requires that d_k makes an angle with Pq_k that is sufficiently obtuse. Again, this is a common type of requirement; e.g., in the deterministic setting, it ensures that any nonzero d_k is a direction of sufficient descent with respect to the barrier-augmented objective function. Together, conditions (5a)–(5d) define a nonempty set and computing a search direction to satisfy these conditions is straightforward; e.g., one can compute the projection of $-q_k$ onto $\operatorname{Null}(A)$ and scale the resulting direction, if needed, to satisfy (5b)–(5c). However, it remains finally to consider condition (5e). Unfortunately, this is a requirement that is not always satisfiable along with the other conditions, namely, (5a)–(5d). That said, we contend that all of (5) can be satisfied as long as (a) the constraints are not degenerate in some sense and (b) the initial barrier parameter μ_1 is sufficiently large relative to θ_0. Denoting the conical hull of nearly active constraint gradients as

$$\mathcal{P}_k := \left\{ \sum_{i \in \mathcal{A}_k} \nabla c_i(x_k)\sigma_i : \sigma \in \mathbb{R}^{|\mathcal{A}_k|}_{\geq 0} \right\},$$

condition (5e) requires that d_k lies sufficiently within the interior of the polar cone of $\mathcal{P}_k$, i.e., int$(\mathcal{P}_k^\circ)$. Hence, to satisfy (5), there needs to exist a direction sufficiently within this interior that also satisfies (5a)–(5d). For our purposes now, let us simply introduce Assumption 3.1 below, which implicitly means that Algorithm 1 is being applied to a problem such that a search direction satisfying (5) always exists. We defer until §5 a more detailed discussion Assumption 3.1, where in particular we provide (a) example problems for which the assumption can be shown to hold or cannot be shown to hold, (b) further explanation about why the assumption can be viewed as a type of nondegeneracy assumption and an assumption about μ_1/θ_0 being sufficient large (which is reminiscent of a requirement in Corollary 3.8 and Lemma 3.15 in [7] for the bound-constrained setting), and (c) a procedure for computing d_k to satisfy (5) when such a direction exists.

Assumption 3.1. *For all $k \in \mathbb{N}$ generated in a run of Algorithm 1, there exists a search direction $d_k \in \mathbb{R}^n$ satisfying the conditions (5).*

Upon computation of the search direction, Algorithm 1 chooses a step size $\alpha_k \in \mathbb{R}_{>0}$. For simplicity, our analyses for the deterministic and stochastic settings consider conservative rules for selecting the step sizes that each depend on the barrier- and neighborhood-parameter sequences, $\{\mu_k\}$ and $\{\theta_k\}$. We conjecture that it would be possible to generalize our algorithm and analyses to allow more flexibility in the step-size selection rules, as is done in [7]. However, for our purposes here, we present conservative strategies that are sufficient for proving convergence.

The algorithm's last major step in the kth iteration is to compute a positive real number $\gamma_k \in \mathbb{R}_{>0}$ such that the line segment $[x_k, x_k + \gamma_k \alpha_k d_k]$ is contained in the neighborhood $\mathcal{N}(\theta_k)$. The subsequent iterate is then set as $x_{k+1} \leftarrow x_k + \gamma_k \alpha_k d_k$. As mentioned at the beginning of this section, from the facts that the kth iterate has $x_k \in \mathcal{E} \cap \mathcal{N}(\theta_{k-1})$, the kth search direction satisfies (5a), and the choice of γ_k ensures that $x_k + \gamma_k \alpha_k d_k \in \mathcal{N}(\theta_k)$, it follows that $x_{k+1} \in \mathcal{E} \cap \mathcal{N}(\theta_k)$. For the sake of formality, we state the following lemma that has been proved.

Lemma 3.1. *For all $k \in \mathbb{N}$ generated in any run of Algorithm 1, it follows that the iterate satisfies $x_k \in \mathcal{E} \cap \mathcal{N}(\theta_{k-1}) \subseteq \mathcal{F}_{<0}$ and the search direction satifies $d_k \in \text{Null}(A)$.*

Algorithm 1 Single-Loop Interior-Point (SLIP) Method

Require: initial neighborhood parameter $\theta_0 \in \mathbb{R}_{>0}$, initial iterate $x_1 \in \mathcal{E} \cap \mathcal{N}(\theta_0)$, barrier-parameter sequence $\{\mu_k\} \searrow 0$, neighborhood-parameter sequence $\{\theta_k\} \searrow 0$ with $\theta_1 < \theta_0$ such that $\{\mu_k/\theta_{k-1}\}$ is a constant sequence, maximum neighborhood-parameter sequence $\{\gamma_{k,\max}\} \subset (0,1]$, $\eta \in (\theta_0/\mu_1, 1)$, $\underline{\eta} \in (\theta_0, \infty)$, $\underline{\zeta} \in \mathbb{R}_{>0}$, $\overline{\zeta} \in \mathbb{R}_{>0}$ with $\underline{\zeta} \leq \overline{\zeta}$, and $\zeta \in (0,1)$
1: **for all** $k \in \mathbb{N}$ **do**
2:      Compute $q_k \in \mathbb{R}^n$ by (4)
3:      Compute $d_k \in \mathbb{R}^n$ to satisfy (5)
4:      Choose $\alpha_k \in \mathbb{R}_{>0}$
5:      Compute $\gamma_k \in (0, \gamma_{k,\max}]$ such that $[x_k, x_k + \gamma_k \alpha_k d_k] \subseteq \mathcal{N}(\theta_k)$
6:      Set $x_{k+1} \leftarrow x_k + \gamma_k \alpha_k d_k$
7: **end for**

---

4 Analysis

We prove theoretical convergence guarantees for Algorithm 1 in both deterministic and stochastic settings. For both settings, we make the following additional, standard type of assumption beyond Assumptions 2.1 and 3.1.

Assumption 4.1. *Let $\mathcal{X}$ be an open convex set containing the sequence of line segments between iterates, namely, $\{[x_k, x_k + \alpha_k d_k]\}$, generated in any run of Algorithm 1. There exist constants $\kappa_{\nabla f} \in \mathbb{R}_{>0}$, $L_{\nabla f} \in \mathbb{R}_{>0}$, $\{\kappa_{c_i}\}_{i \in [m]} \subset \mathbb{R}_{>0}$, $\{L_{c_i}\}_{i \in [m]} \subset \mathbb{R}_{>0}$, $\{\kappa_{\nabla c_i}\}_{i \in [m]} \subset \mathbb{R}_{>0}$, and $\{L_{\nabla c_i}\}_{i \in [m]} \subset \mathbb{R}_{>0}$ such that:*

- f is Lipschitz with constant $\kappa_{\nabla f}$ over $\mathcal{X}$, so (under Assumption 2.1) it follows that ∇f is bounded in ℓ_2 (Euclidean) norm by $\kappa_{\nabla f}$ over $\mathcal{X}$.

- ∇f is Lipschitz with constant $L_{\nabla f}$ with respect to $\|\cdot\|_2$ over $\mathcal{X}$.

- For all $i \in [m]$, c_i is bounded in absolute value by κ_{c_i} over $\mathcal{X}$.

- For all $i \in [m]$, c_i is Lipschitz with constant L_{c_i} over $\mathcal{X}$, so (under Assumption 2.1) it follows that ∇c_i is bounded in ℓ_2 (Euclidean) norm by $\kappa_{\nabla c_i} := L_{c_i}$ over $\mathcal{X}$.

- For all $i \in [m]$, ∇c_i is Lipschitz with constant $L_{\nabla c_i}$ with respect to $\|\cdot\|_2$ over $\mathcal{X}$.

Under Assumption 4.1, with $\{\kappa_{c_i}\}_{i\in[m]} \subset \mathbb{R}_{>0}$ defined in the assumption, we introduce $\mathcal{C}_{[-\kappa_c,0)} := \{x \in \mathbb{R}^n : c_i(x) \in [-\kappa_{c_i}, 0) \text{ for all } i \in [m]\}$ along with the shifted barrier-augmented objective function $\tilde{\phi} : \mathcal{C}_{[-\kappa_c,0)} \times \mathbb{R}_{>0} \to \mathbb{R}$ defined by

$$\tilde{\phi}(x,\mu) = f(x) - \mu \sum_{i\in[m]} \log\left(-\frac{c_i(x)}{\kappa_{c_i}}\right).$$

Important properties of the shifted barrier-augmented objective function are that (a) it is bounded below by $f_{\inf} := \inf_{x\in\mathcal{F}_{<0}} f(x) \in \mathbb{R}$ whose existence follows under Assumption 2.1 and (b) it shares the same gradient function with respect to its first argument as that of the (unshifted) barrier-augmented objective function; i.e., for all $(x,\mu) \in \mathcal{C}_{[-\kappa_c,0)} \times \mathbb{R}_{>0}$, one finds that

$$\nabla_x \phi(x,\mu) = \nabla_x \tilde{\phi}(x,\mu), \tag{6}$$

where ∇_x denotes the gradient operator with respect to a function's first argument. These follow since for any $i \in [m]$ and $\delta \in (0, \kappa_{c_i}]$ one has $-\log(\delta/\kappa_{c_i}) \geq 0$ and $-\log(\delta/\kappa_{c_i}) = -\log(\delta) + \log(\kappa_{c_i})$, so $(x,\mu) \in \mathcal{C}_{[-\kappa_{c_i},0)} \times \mathbb{R}_{>0}$ implies

$$\phi(x,\mu) + \mu \sum_{i\in[m]} \log(\kappa_{c_i}) = \tilde{\phi}(x,\mu) \geq f_{\inf}. \tag{7}$$

In addition, the shifted barrier-augmented objective function is monotonically increasing in its second argument in the sense that for any $x \in \mathcal{C}_{[-\kappa_c,0)}$ one finds that $\tilde{\phi}(x,\mu) < \tilde{\phi}(x,\bar{\mu})$ for any $(\mu,\bar{\mu}) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ with $\mu < \bar{\mu}$. These properties make the function $\tilde{\phi}$ useful for our analysis. Indeed, even though the algorithm is not aware of the constants $\{\kappa_{c_i}\}_{i\in[m]}$ in Assumption 4.1, we shall show that it follows from the former property above that it computes steps that lead to (expected) decrease in the shifted barrier-augmented objective function $\tilde{\phi}$. Also, with the latter property above, decreases in the barrier parameter also lead to decreases in $\tilde{\phi}$.

Our first result is useful for both the deterministic and stochastic settings. It shows that, for any $\theta \in \mathbb{R}_{>0}$, the gradient of the shifted barrier-augmented objective function with respect to its first argument is Lipschitz over line segments in $\mathcal{N}(\theta)$ with a Lipschitz constant that depends on θ. This is a critical property, especially since this function is not globally Lipschitz over $\mathcal{F}_{<0} \supset \mathcal{N}(\theta)$. The proof follows standard techniques for such a result; we provide it for completeness.

Lemma 4.1. *For any* $(\mu,\theta,\bar{\theta}) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ *with* $\bar{\theta} \in (0,\theta]$, $x \in \mathcal{N}(\theta)$, $\bar{x} \in \mathcal{N}(\bar{\theta})$, *and* $\gamma \in [0,1]$, *one finds with*

$$L_{\mu,\theta,\bar{\theta}} := L_{\nabla f} + \frac{\mu}{\theta\bar{\theta}} \sum_{i=1}^m (L_{c_i}\kappa_{\nabla c_i} + \kappa_{c_i}L_{\nabla c_i}) \in \mathbb{R}_{>0}$$

that one has both

$$\|\nabla_x\tilde{\phi}(x + \gamma(\bar{x}-x),\mu) - \nabla_x\tilde{\phi}(x,\mu)\|_2 \leq \gamma L_{\mu,\theta,\bar{\theta}}\|\bar{x}-x\|_2$$
$$\text{and } \tilde{\phi}(\bar{x},\mu) \leq \tilde{\phi}(x,\mu) + \nabla_x\tilde{\phi}(x,\mu)^T(\bar{x}-x) + \tfrac{1}{2}L_{\mu,\theta,\bar{\theta}}\|\bar{x}-x\|_2^2$$

Proof. For arbitrary $(\mu, \theta, \bar{\theta}, x, \bar{x}, \gamma)$ satisfying the conditions of the lemma, it follows from Assumption 2.1, Assumption 4.1, (6), and the triangle inequality that

$$
\begin{aligned}
&\|\nabla_x \tilde{\phi}(x + \gamma(\bar{x} - x), \mu) - \nabla_x \tilde{\phi}(x, \mu)\|_2 \\
&= \left\| \nabla f(x + \gamma(\bar{x} - x)) - \nabla f(x) - \mu \left(\sum_{i=1}^m \frac{\nabla c_i(x + \gamma(\bar{x} - x))}{c_i(x + \gamma(\bar{x} - x))} - \sum_{i=1}^m \frac{\nabla c_i(x)}{c_i(x)} \right) \right\|_2 \\
&\leq \gamma L_{\nabla f} \|\bar{x} - x\|_2 + \mu \sum_{i=1}^m \left\| \frac{c_i(x + \gamma(\bar{x} - x))\nabla c_i(x) - c_i(x)\nabla c_i(x + \gamma(\bar{x} - x))}{c_i(x)c_i(x + \gamma(\bar{x} - x))} \right\|_2 \\
&\leq \gamma L_{\nabla f} \|\bar{x} - x\|_2 + \frac{\mu}{\theta\bar{\theta}} \sum_{i=1}^m \| c_i(x + \gamma(\bar{x} - x))\nabla c_i(x) - c_i(x)\nabla c_i(x) \\
&\qquad\qquad\qquad\qquad\qquad + c_i(x)\nabla c_i(x) - c_i(x)\nabla c_i(x + \gamma(\bar{x} - x)) \|_2 \\
&\leq \gamma L_{\nabla f} \|\bar{x} - x\|_2 + \frac{\mu}{\theta\bar{\theta}} \sum_{i=1}^m (|c_i(x + \gamma(\bar{x} - x)) - c_i(x)|\|\nabla c_i(x)\|_2 \\
&\qquad\qquad\qquad\qquad\qquad + |c_i(x)|\|\nabla c_i(x) - \nabla c_i(x + \gamma(\bar{x} - x))\|_2) \\
&\leq \gamma L_{\nabla f} \|\bar{x} - x\|_2 + \frac{\mu}{\theta\bar{\theta}} \sum_{i=1}^m (\gamma L_{c_i} \kappa_{\nabla c_i} \|\bar{x} - x\|_2 + \gamma \kappa_{c_i} L_{\nabla c_i} \|\bar{x} - x\|_2),
\end{aligned}
$$

from which the first desired conclusion follows. Hence, for arbitrary $(\mu, \theta, \bar{\theta}, x, \bar{x})$ satisfying the conditions of the lemma, it follows from the Fundamental Theorem of Calculus and the Cauchy–Schwarz inequality that

$$
\begin{aligned}
&\tilde{\phi}(\bar{x}, \mu) - \tilde{\phi}(x, \mu) \\
&= \int_0^1 \frac{\partial \tilde{\phi}(x + \gamma(\bar{x} - x), \mu)}{\partial \gamma} \mathrm{d}\gamma = \int_0^1 \nabla_x \tilde{\phi}(x + \gamma(\bar{x} - x), \mu)^T (\bar{x} - x) \mathrm{d}\gamma \\
&= \nabla_x \tilde{\phi}(x, \mu)^T (\bar{x} - x) + \int_0^1 (\nabla_x \tilde{\phi}(x + \gamma(\bar{x} - x), \mu) - \nabla_x \tilde{\phi}(x, \mu))^T (\bar{x} - x) \mathrm{d}\gamma \\
&\leq \nabla_x \tilde{\phi}(x, \mu)^T (\bar{x} - x) + \|\bar{x} - x\|_2 \int_0^1 \|\nabla_x \tilde{\phi}(x + \gamma(\bar{x} - x), \mu) - \nabla_x \tilde{\phi}(x, \mu)\|_2 \mathrm{d}\gamma \\
&\leq \nabla_x \tilde{\phi}(x, \mu)^T (\bar{x} - x) + L_{\mu, \theta, \bar{\theta}} \|\bar{x} - x\|_2 \int_0^1 \gamma \mathrm{d}\gamma,
\end{aligned}
$$

so the desired conclusion follows from the fact that $\int_0^1 \gamma \mathrm{d}\gamma = \frac{1}{2}$. □ □

4.1 Deterministic Setting

In this subsection, we prove convergence guarantees for Algorithm 1 in the deterministic setting, i.e., when $q_k = \nabla f(x_k) - \mu_k \nabla c(x_k) \operatorname{diag}(c(x_k))^{-1} \mathbb{1} = \nabla_x \tilde{\phi}(x_k, \mu_k)$ (see (4) and (6)) for all $k \in \mathbb{N}$. For ease of exposition, we refer to this instance of the method as Algorithm 1(D). In particular, we prove that under a certain rule for choosing the step-size parameters, the algorithm is well defined, generates a sequence of iterates over which a measure of stationarity vanishes, and under a constraint qualification can produce a sequence of Lagrange multipliers such that convergence to a KKT point is guaranteed. This algorithm for the deterministic setting is not expected to be competitive with state-of-the-art (second-order) interior-point methods for solving problems in many real-world settings. That said, our analysis for the deterministic setting provides a useful precursor for our subsequent analysis for the stochastic setting for which state-of-the-art methods are not applicable (since they do not possess known convergence guarantees in the stochastic setting that we consider in our analysis in §4.2).

The class of step-size parameters that we consider for our analysis in this subsection is that satisfying the following parameter rule. We state the rule, then summarize the reasons that the rule has been designed in this manner.

Parameter 1. *Given $t_\alpha \in (-\infty, 0]$, for all $k \in \mathbb{N}$, Algorithm 1(D) employs*

$$\alpha_k \leftarrow k^{t_\alpha} \left(\frac{\zeta\zeta}{\overline{\zeta}^2 L_k} \right),$$

where $L_k := L_{\nabla f} + \frac{\mu_k}{\theta_k \theta_{k-1}} \sum_{i=1}^{m} (L_{c_i} \kappa_{\nabla c_i} + \kappa_{c_i} L_{\nabla c_i}) \in \mathbb{R}_{>0}$. In addition, for all $k \in \mathbb{N}$, Algorithm 1(D) employs $\gamma_{k,\max} \leftarrow 1$ and $\gamma_k \leftarrow \min_{i \in [m]} \gamma_{k,i}$, where, for all $i \in [m]$,

$$\gamma_{k,i} \leftarrow \min\left\{ 1, \frac{-\nabla c_i(x_k)^T d_k + \sqrt{(\nabla c_i(x_k)^T d_k)^2 + 2L_{\nabla c_i} \|d_k\|_2^2(-c_i(x_k) - \theta_k)}}{L_{\nabla c_i} \alpha_k \|d_k\|_2^2} \right\};$$

hence, $\gamma_{k,i} \in (0,1]$ for all $k \in \mathbb{N}$ and $i \in [m]$, so $\gamma_k \in (0,1]$ for all $k \in \mathbb{N}$.

The fundamental aspects of Parameter Rule 1 can be understood as follows. First, the choice of α_k ensures that this step size is chosen sufficiently small such that it ensures sufficient decrease in the barrier-augmented objective function with each step. Intuitively, this can be seen in the fact that the step size is proportional to the reciprocal of a Lipschitz-type constant for the gradient of the barrier function (see Lemma 4.1), as is common in basic descent methods for gradient-based optimization algorithms. For additional flexibility and since it is required in the stochastic setting, we introduce the factor k^{t_α} such that the step size may diminish faster, namely, when a choice $t_\alpha < 0$ is made. (That said, our analysis shows that $t_\alpha = 0$ is a valid choice in the deterministic setting.) Finally, the rule for the choice of γ_k is shown in our analysis to guarantee that $[x_k, x_k + \gamma_k \alpha_k d_k] \subseteq \mathcal{N}(\theta_k)$ for all $k \in \mathbb{N}$, which is necessary for our analysis to employ Lemma 4.1.

For our next lemma, we prove this critical property of γ_k for all $k \in \mathbb{N}$.

Lemma 4.2. *Suppose that Assumptions 2.1, 3.1, and 4.1 hold and that Algorithm 1(D) is run with Parameter Rule 1. Then, for all $k \in \mathbb{N}$. the choice of $\gamma_k \in (0,1]$ in Parameter Rule 1 ensures that $[x_k, x_k + \gamma_k \alpha_k d_k] \subseteq \mathcal{N}(\theta_k)$, i.e., the condition in Step 5 of Algorithm 1(D) holds. Consequently, under Parameter Rule 1, Algorithm 1(D) is well defined in the sense that it will generate an infinite sequence of iterates.*

Proof. Consider arbitrary $k \in \mathbb{N}$ and $i \in [m]$. We prove that the choice of $\gamma_{k,i} \in (0,1]$ in Parameter Rule 1 ensures that $c_i(x_k + \gamma\alpha_k d_k) \leq -\theta_k$ for all $\gamma \in [0, \gamma_{k,i}]$, after which the desired conclusion follows directly from the fact that $\gamma_k \leftarrow \min_{i \in [m]} \gamma_{k,i}$ for all $k \in \mathbb{N}$. Under Assumptions 2.1 and 4.1, one has for all $\gamma \in [0,1]$ that

$$c_i(x_k + \gamma\alpha_k d_k) \leq c_i(x_k) + \gamma\alpha_k \nabla c_i(x_k)^T d_k + \tfrac{1}{2} L_{\nabla c_i} \gamma^2 \alpha_k^2 \|d_k\|_2^2.$$

Therefore, $c_i(x_k + \gamma\alpha_k d_k) \leq -\theta_k$ is ensured as long as $\gamma \in [0,1]$ yields

$$c_i(x_k) + \overline{\gamma}\alpha_k \nabla c_i(x_k)^T d_k + \tfrac{1}{2} L_{\nabla c_i} \overline{\gamma}^2 \alpha_k^2 \|d_k\|_2^2 \leq -\theta_k \quad \text{for all} \ \ \overline{\gamma} \in [0, \gamma]. \tag{8}$$

Recalling that Lemma 3.1 ensures $x_k \in \mathcal{N}(\theta_{k-1})$, which means that $c_i(x_k) \leq -\theta_{k-1} < -\theta_k$, it follows that (8) holds for all $\gamma \in [0,1]$ when $d_k = 0$, and otherwise (when $d_k \neq 0$) one finds that the left-hand side of (8) is a strongly convex quadratic function of $\overline{\gamma}$. The second term in the minimum in the definition of $\gamma_{k,i}$ in Parameter Rule 1 is the unique positive root of this quadratic function, where we remark that $-c_i(x_k) - \theta_k > 0$ since, as previously mentioned, $c_i(x_k) < -\theta_k$. ☐ ☐

We now go beyond Lemma 4.2 and prove a lower bound for γ_k that will be critical for our ultimate convergence guarantee in this subsection.

10

Lemma 4.3. *Suppose that Assumptions 2.1, 3.1, and 4.1 hold and that Algorithm 1(D) is run with Parameter Rule 1. In addition, define*

$$\beta := \overline{\zeta}\left(\kappa_{\nabla f} + \frac{\mu_1}{\theta_0}\sum_{j\in[m]}\kappa_{\nabla c_j}\right) \in \mathbb{R}_{>0},$$

and for all $i \in [m]$ define

$$\gamma_{k,i,\min} := \min\left\{1, \frac{-\kappa_{\nabla c_i} + \sqrt{\kappa_{\nabla c_i}^2 + 2L_{\nabla c_i}(\eta\mu_k - \theta_k)}}{\alpha_k\beta L_{\nabla c_i}}, \frac{\underline{\eta} - \theta_k}{\alpha_k\beta L_{\nabla c_i}}\right\} \in (0,1].$$

Then, $\gamma_k \geq \gamma_{k,\min} := \min_{i\in[m]}\gamma_{k,i,\min}$ for all $k \in \mathbb{N}$.

Proof. Consider arbitrary $k \in \mathbb{N}$ and $i \in [m]$. Our aim is to prove that $\gamma_{k,i} \geq \gamma_{k,i,\min}$, from which the desired conclusion follows due to the fact that $\gamma_k \leftarrow \min_{i\in[m]}\gamma_{k,i}$ under Parameter Rule 1. Toward this end, observe that if Parameter Rule 1 yields $\gamma_{k,i} = 1$, then, clearly, $\gamma_{k,i} \geq \gamma_{k,i,\min} \in (0,1]$ and there is nothing left to prove. Therefore, we may proceed under the assumption that Parameter Rule 1 yields $\gamma_{k,i} \in (0,1)$. It follows under this condition that

$$\gamma_{k,i} = \frac{\frac{-\nabla c_i(x_k)^T d_k}{\|d_k\|_2} + \sqrt{\left(\frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}\right)^2 + 2L_{\nabla c_i}(-c_i(x_k) - \theta_k)}}{\alpha_k\|d_k\|_2 L_{\nabla c_i}}. \tag{9}$$

On the other hand, by Assumptions 2.1, 3.1, and 4.1, the fact that the matrix norm $\|\cdot\|_2$ is submultiplicative, the triangle inequality, (5c), the fact that $x_k \in \mathcal{N}(\theta_{k-1})$, the fact that $\{\mu_k/\theta_{k-1}\}$ is a constant sequence, and the fact that for the orthogonal projection matrix P one has $\|P\|_2 \leq 1$, one finds that

$$\begin{aligned}\|d_k\|_2 &\leq \overline{\zeta}\|P(\nabla f(x_k) - \mu_k\nabla c(x_k)\operatorname{diag}(c(x_k))^{-1}\mathbb{1})\|_2 \\ &\leq \overline{\zeta}\left(\|\nabla f(x_k)\|_2 + \mu_k\sum_{j\in[m]}\left\|c_j(x_k)^{-1}\nabla c_j(x_k)\right\|_2\right) \leq \beta.\end{aligned} \tag{10}$$

Let us now proceed by considering two cases.

1. Suppose $c_i(x_k) \leq -\eta\mu_k$. (The ith constraint is not nearly active.) Observe that Assumption 4.1 and the Cauchy-Schwarz inequality together imply that $\nabla c_i(x_k)^T d_k \leq \kappa_{\nabla c_i}\|d_k\|_2$. At the same time, observe that the numerator of the right-hand side of (9) can be viewed as the value of the function $h : (-\infty, \kappa_{\nabla c_i}] \to \mathbb{R}$ defined by $h(a) = -a + \sqrt{a^2 + b}$ at $a = \frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}$ for $b \in \mathbb{R}_{>0}$, where positivity of b follows since $x_k \in \mathcal{N}(\theta_{k-1})$ implies $-c_i(x_k) - \theta_k \geq \theta_{k-1} - \theta_k > 0$. One finds that h is a monotonically decreasing function of a over $(-\infty, \kappa_{\nabla c_i}]$. Hence, from (9), (10), and the condition of this case, one has

$$\begin{aligned}\gamma_{k,i} &\geq \frac{-\kappa_{\nabla c_i} + \sqrt{\kappa_{\nabla c_i}^2 + 2L_{\nabla c_i}(-c_i(x_k) - \theta_k)}}{\alpha_k\beta L_{\nabla c_i}} \\ &\geq \frac{-\kappa_{\nabla c_i} + \sqrt{\kappa_{\nabla c_i}^2 + 2L_{\nabla c_i}(\eta\mu_k - \theta_k)}}{\alpha_k\beta L_{\nabla c_i}}.\end{aligned} \tag{11}$$

2. Suppose $c_i(x_k) > -\eta\mu_k$. (The ith constraint is nearly active.) Under Assumption 3.1, it follows from (5e) that $\nabla c_i(x_k)^T d_k \leq -\frac{1}{2}\underline{\eta}\|d_k\|_2$. Consequently, by $\theta_k > 0$, the fact that $x_k \in \mathcal{N}(\theta_{k-1})$ implies $-c_i(x_k) - \theta_k > 0$, and the fact that the conditions of the lemma ensure $\underline{\eta} - \theta_k > 0$, it follows that

$$-\frac{\underline{\eta}}{2} \geq \frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}$$

11

$$\implies \qquad L_{\nabla c_i}\left(\frac{-c_i(x_k)-\theta_k}{\underline{\eta}-\theta_k}\right) - \frac{\underline{\eta}-\theta_k}{2} \geq \frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}$$

$$\implies \qquad 2L_{\nabla c_i}(-c_i(x_k)-\theta_k) - (\underline{\eta}-\theta_k)^2 \geq 2(\underline{\eta}-\theta_k)\frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}$$

$$\implies \qquad \left(\frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}\right)^2 + 2L_{\nabla c_i}(-c_i(x_k)-\theta_k) \geq \left(\frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2} + (\underline{\eta}-\theta_k)\right)^2$$

$$\implies \qquad \sqrt{\left(\frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2}\right)^2 + 2L_{\nabla c_i}(-c_i(x_k)-\theta_k)} \geq \frac{\nabla c_i(x_k)^T d_k}{\|d_k\|_2} + (\underline{\eta}-\theta_k),$$

which with (9) and (10) implies

$$\gamma_{k,i} \geq \frac{\underline{\eta}-\theta_k}{\alpha_k \beta L_{\nabla c_i}}.$$

Combining the results from the two cases, one can conclude that $\gamma_{k,i} \geq \gamma_{k,i,\min}$ for all $i \in [m]$, which, as previously mentioned, completes the proof. $\qquad\square\qquad\qquad\square$

For our next lemma, we prove the aforementioned sufficient decrease condition that is guaranteed by the choices in Parameter Rule 1. In particular, one finds in the lemma that the amount of decrease is at least proportional to a squared norm of the *projected* gradient of the barrier-augmented objective function, which makes sense since, under Lemma 3.1, $x_k \in \mathcal{E}$ and $d_k \in \text{Null}(A)$ for all $k \in \mathbb{N}$.

Lemma 4.4. *Suppose that Assumptions 2.1, 3.1, and 4.1 hold and that Algorithm 1(D) is run with Parameter Rule 1. Then, for all $k \in \mathbb{N}$, one finds*

$$\tilde{\phi}(x_{k+1},\mu_{k+1}) - \tilde{\phi}(x_k,\mu_k) \leq -\tfrac{1}{2}\underline{\zeta}\zeta\gamma_k\alpha_k\|P\nabla_x\phi(x_k,\mu_k)\|_2^2.$$

Proof. Consider arbitrary $k \in \mathbb{N}$. By Lemmas 4.1–4.2, (6), (5), $d_k = Pd_k$, and $P = P^T$, it follows with $L_k \in \mathbb{R}_{>0}$ defined as in Parameter Rule 1 that

$$\begin{aligned}
\tilde{\phi}(x_{k+1},\mu_k) - \tilde{\phi}(x_k,\mu_k) &\leq \nabla_x\tilde{\phi}(x_k,\mu_k)^T(x_{k+1}-x_k) + \tfrac{1}{2}L_k\|x_{k+1}-x_k\|_2^2 \\
&= \gamma_k\alpha_k q_k^T d_k + \tfrac{1}{2}\gamma_k^2\alpha_k^2 L_k\|d_k\|_2^2 \\
&= \gamma_k\alpha_k(Pq_k)^T d_k + \tfrac{1}{2}\gamma_k^2\alpha_k^2 L_k\|d_k\|_2^2 \\
&\leq -\gamma_k\alpha_k\underline{\zeta}\|Pq_k\|_2\|d_k\|_2 + \tfrac{1}{2}\gamma_k^2\alpha_k^2 L_k\|d_k\|_2^2 \\
&\leq -\gamma_k\alpha_k(\underline{\zeta}\zeta - \tfrac{1}{2}\overline{\zeta}^2\gamma_k\alpha_k L_k)\|Pq_k\|_2^2. \qquad (12)
\end{aligned}$$

Due to Parameter Rule 1, one finds that $\gamma_k \in (0,1]$, $k^{t_\alpha} \in (0,1]$, and

$$\underline{\zeta}\zeta - \tfrac{1}{2}\overline{\zeta}^2\gamma_k\alpha_k L_k \geq \underline{\zeta}\zeta - \tfrac{1}{2}\overline{\zeta}^2\alpha_k L_k \geq \underline{\zeta}\zeta - \tfrac{1}{2}k^{t_\alpha}\underline{\zeta}\zeta \geq \tfrac{1}{2}\underline{\zeta}\zeta. \qquad (13)$$

Combining (12) and (13), and since $\mu_{k+1} \leq \mu_k$ implies $\tilde{\phi}(x_{k+1},\mu_{k+1}) \leq \tilde{\phi}(x_{k+1},\mu_k)$ (see the discussion following (7)), the desired conclusion follows. $\qquad\square\qquad\qquad\square$

We are now prepared to prove our guarantee for the deterministic setting. For the sake of notational clarity, the following theorem introduces the parameters t_μ and t_θ that dictate the rates of decrease of $\{\mu_k\}$ and $\{\theta_k\}$, respectively. However, the theorem requires $t_\mu = t_\theta$, and indeed one can see in the details of the proof that both $t_\mu \leq t_\theta$ and $t_\mu \geq t_\theta$ are required to prove the theorem.

Theorem 4.1. *Suppose that Assumptions 2.1, 3.1, and 4.1 hold and that Algorithm 1(D) is run with Parameter Rule 1. Suppose, in addition, that for some $(t_\mu, t_\theta, t_\alpha) \in (-\infty,0) \times (-\infty,0) \times (-\infty,0]$ with*

$t_\mu = t_\theta$ and $t_\mu + t_\alpha \in [-1, 0)$ and for some $\mu_1 \in \mathbb{R}_{>0}$ and $\theta_0 \in \mathbb{R}_{>0}$ the algorithm employs $\mu_k = \mu_1 k^{t_\mu}$ and $\theta_{k-1} = \theta_0 k^{t_\theta}$ for all $k \in \mathbb{N}$. Then,

$$\sum_{k=1}^{\infty} \gamma_k \alpha_k = \infty \quad and \quad \liminf_{k \to \infty} \|P \nabla_x \phi(x_k, \mu_k)\|_2^2 = 0. \tag{14}$$

If, in addition, there exists $\mathcal{K} \subseteq \mathbb{N}$ with $|\mathcal{K}| = \infty$ such that

- $\{P \nabla_x \phi(x_k, \mu_k)\}_{k \in \mathcal{K}} \to 0$,

- $\{x_k\}_{k \in \mathcal{K}} \to \bar{x}$ for some $\bar{x} \in \mathcal{F}$, and

- at $\bar{x}$ the linear independence constraint qualification (LICQ) holds with respect to problem (1) in the sense that with $\bar{\mathcal{A}} := \{i \in [m] : c_i(\bar{x}) = 0\}$ the columns of A^T combined with the vectors in $\{\nabla c_i(\bar{x})\}_{i \in \bar{\mathcal{A}}}$ form a linearly independent set,

then $\bar{x}$ is a KKT point for problem (1) in the sense that there exists a pair of Lagrange multipliers $(\bar{y}, \bar{z}) \in \mathbb{R}^l \times \mathbb{R}^m$ such that $(\bar{x}, \bar{y}, \bar{z})$ satisfies (3).

Proof. It follows from (7) that $\tilde{\phi}$ is bounded below by $f_{\inf}$ over $\mathcal{X} \times \mathbb{R}_{\geq 0}$. Then, by summing the expression in Lemma 4.4 over $k \in \mathbb{N}$ and (6), one finds that

$$\infty > \tilde{\phi}(x_1, \mu_1) - f_{\inf} \geq \sum_{k=1}^{\infty} (\tilde{\phi}(x_k, \mu_k) - \tilde{\phi}(x_{k+1}, \mu_{k+1}))$$

$$\geq \sum_{k=1}^{\infty} \tfrac{1}{2} \underline{\zeta} \zeta \gamma_k \alpha_k \|P \nabla_x \phi(x_k, \mu_k)\|_2^2.$$

To complete the proof of (14), let us now show that $\{\gamma_k \alpha_k\}$ is unsummable. To begin, first observe that the lower bound on $\{\gamma_k\}$ stated in Lemma 4.3 holds, i.e., there exist $\kappa \in \mathbb{R}_{>0}$, $\underline{L} \in \mathbb{R}_{>0}$, and $L \in \mathbb{R}_{>0}$ such that, for all $k \in \mathbb{N}$, one has

$$\gamma_{k,\min} \geq \min \left\{ 1, \frac{-\kappa + \sqrt{\kappa^2 + 2\underline{L}(\eta \mu_k - \theta_k)}}{\alpha_k \beta L}, \frac{\eta - \theta_k}{\alpha_k \beta L} \right\}.$$

Consequently, it holds for all $k \in \mathbb{N}$ that

$$\gamma_k \alpha_k \geq \gamma_{k,\min} \alpha_k \geq \min \left\{ \alpha_k, \frac{-\kappa + \sqrt{\kappa^2 + 2\underline{L}(\eta \mu_k - \theta_k)}}{\beta L}, \frac{\eta - \theta_k}{\beta L} \right\}$$

$$=: \min\{\alpha_k, \rho_k, \upsilon_k\}. \tag{15}$$

Let us now consider the sequences $\{\alpha_k\}$, $\{\rho_k\}$, and $\{\upsilon_k\}$ in turn. Our aim is to show that each of these sequences is unsummable, which will complete the proof of (14). With respect to the step-size sequence $\{\alpha_k\}$, one finds by Parameter Rule 1, $\upsilon := \sum_{i=1}^{m} (L_{c_i} \kappa_{\nabla c_i} + \kappa_{c_i} L_{\nabla c_i})$, and the conditions of the theorem that

$$\alpha_k = \frac{k^{t_\alpha} \underline{\zeta} \zeta \bar{\zeta}^{-2}}{L_{\nabla f} + \mu_1 \theta_0^{-2} k^{t_\mu} k^{-t_\theta} (k+1)^{-k_\theta} \upsilon}. \tag{16}$$

Observe that $t_\mu \in (-\infty, 0)$, $t_\theta \in (-\infty, 0)$, and $t_\mu = t_\theta$ show that $k^{t_\mu} k^{-t_\theta} (k+1)^{-t_\theta} \leq k^{t_\mu - t_\theta} (2k)^{-t_\theta} = 2^{-t_\theta} k^{t_\mu - 2t_\theta} = 2^{-t_\mu} k^{-t_\mu}$. With (16) and $t_\mu \in (-\infty, 0)$, this shows

$$\alpha_k \geq \frac{k^{t_\alpha} \underline{\zeta} \zeta \bar{\zeta}^{-2}}{L_{\nabla f} + \mu_1 \theta_0^{-2} 2^{-t_\mu} \upsilon k^{-t_\mu}} = \frac{k^{t_\mu + t_\alpha} \underline{\zeta} \zeta \bar{\zeta}^{-2}}{L_{\nabla f} k^{t_\mu} + \mu_1 \theta_0^{-2} 2^{-t_\mu} \upsilon} \geq \frac{k^{t_\mu + t_\alpha} \underline{\zeta} \zeta \bar{\zeta}^{-2}}{L_{\nabla f} + \mu_1 \theta_0^{-2} 2^{-t_\mu} \upsilon},$$

which since $t_\mu + t_\alpha \in [-1, 0)$ implies that $\{\alpha_k\}$ is unsummable, as desired. Now, with respect to $\{\rho_k\}$, one finds under the conditions of the theorem that

$$\rho_k = \frac{-\kappa + \sqrt{\kappa^2 + 2\underline{L}(\eta\mu_1 k^{t_\mu} - \theta_0(k+1)^{t_\theta})}}{\beta L}. \tag{17}$$

Observe that $t_\mu \in (-\infty, 0)$ and $t_\mu = t_\theta$ show that

$$\eta\mu_1 k^{t_\mu} - \theta_0(k+1)^{t_\theta} \geq \eta\mu_1 k^{t_\mu} - \theta_0 k^{t_\theta} = (\eta\mu_1 - \theta_0)k^{t_\mu}.$$

In fact, $t_\mu \in [-1, 0)$, so the bound above along with (17) shows that $\{\rho_k\}$ is unsummable, as desired. Finally, $\{\upsilon_k\}$ can be seen to be unsummable since $\upsilon_k = \underline{\eta} - \theta_k > \underline{\eta} - \theta_0$ for all $k \in \mathbb{N}$. Having reached the desired conclusion that $\{\alpha_k\}$, $\{\rho_k\}$, and $\{\upsilon_k\}$ are unsummable, it follows through (15) that $\{\gamma_k\alpha_k\}$ is unsummable, which, as previously mentioned, completes the proof of (14).

Now suppose that there exists an infinite-cardinality set $\mathcal{K} \subseteq \mathbb{N}$ as described in the theorem. By construction of the algorithm, it follows that $Ax_k = b$ and $c(x_k) < 0$ for all $k \in \mathbb{N}$. Now, for all $k \in \mathcal{K}$, define the Lagrange multiplier estimates

$$z_k := -\mu_k \operatorname{diag}(c(x_k))^{-1}\mathbb{1} \quad \text{and} \quad y_k := -(AA^T)^{-1}A(\nabla f(x_k) + \nabla c(x_k)z_k). \tag{18}$$

Since $\mu_k > 0$ and $c(x_k) < 0$ for all $k \in \mathbb{N}$, it follows that $z_k \geq 0$ for all $k \in \mathbb{N}$. In addition, the fact that $\{P\nabla\phi(x_k, \mu_k)\}_{k \in \mathcal{K}} \to 0$ shows that

$$\begin{aligned}
&\{\|\nabla f(x_k) + A^T y_k + \nabla c(x_k)z_k\|_2\}_{k \in \mathcal{K}} \\
&= \{\|(\nabla f(x_k) + \nabla c(x_k)z_k) - A^T(AA^T)^{-1}A(\nabla f(x_k) + \nabla c(x_k)z_k)\|_2\}_{k \in \mathcal{K}} \\
&= \{\|P\nabla\phi(x_k, \mu_k)\|_2\}_{k \in \mathcal{K}} \to 0.
\end{aligned} \tag{19}$$

Consider $\bar{\mathcal{A}} = \{i \in [m] : c_i(\bar{x}) = 0\}$. Since $c(x_k) \circ z_k = -\mu_k \mathbb{1}$ for all $k \in \mathbb{N}$ and $\{\mu_k\} \searrow 0$, it follows that $\{[z_k]_i\}_{k \in \mathcal{K}} \to 0$ for all $i \notin \bar{\mathcal{A}}$. Combining this with (19) and using the fact that $\{\nabla c_i\}_{k \in \mathcal{K}}$ is bounded for all $i \in [m]$ under Assumption 4.1,

$$\left\{\left\|\nabla f(x_k) + A^T y_k + \sum_{i \in \bar{\mathcal{A}}} \nabla c_i(x_k)[z_k]_i\right\|_2\right\}_{k \in \mathcal{K}} \to 0.$$

Under the LICQ, it follows from this limit that $\{(y_k, z_k)\}_{k \in \mathcal{K}}$ converges to some pair $(\bar{y}, \bar{z})$ such that $(\bar{x}, \bar{y}, \bar{z})$ satisfies (3), as desired. $\qquad\square\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

4.2 Stochastic Setting

We now prove convergence guarantees for Algorithm 1 in a stochastic setting, i.e., when $q_k = g_k - \mu_k \nabla c(x_k)\operatorname{diag}(c(x_k))^{-1}\mathbb{1}$ (see (4)) for all $k \in \mathbb{N}$ where g_k is a stochastic estimate of $\nabla f(x_k)$. Formally, let us consider the probability space $(\Omega, \mathcal{G}, \mathbb{P})$, where Ω captures all outcomes of a run of Algorithm 1. As mentioned shortly, we make assumptions that guarantee that each iteration is well defined, which implies that each such outcome corresponds to an infinite sequence of iterates. In this manner, each realization of a run of the algorithm can be associated with $\omega \in \Omega$, an infinite-dimensional tuple whose kth element determines the stochastic gradient-of-the-objective estimate in iteration $k \in \mathbb{N}$. The stochastic process defined by the algorithm can thus be expressed as

$$\{(X_k(\omega), G_k(\omega), Q_k(\omega), D_k(\omega), \mathcal{A}_k(\omega), \Gamma_k(\omega))\},$$

where, for all $k \in \mathbb{N}$, the random variables are the iterate $X_k(\omega)$, stochastic objective-gradient estimator $G_k(\omega)$, stochastic gradient estimator

$$Q_k(\omega) := G_k(\omega) - \mu_k \nabla c(X_k(\omega))\operatorname{diag}(c(X_k(\omega)))^{-1}\mathbb{1}, \tag{20}$$

14

direction $D_k(\omega)$, step size $\mathcal{A}_k(\omega)$, and neighborhood-enforcement parameter $\Gamma_k(\omega)$. Observe that the statement of Algorithm 1 on page 7 instantiates a particular realization of a run, expressed as $\{(x_k, g_k, q_k, d_k, \alpha_k, \gamma_k)\}$.

Given the initial conditions of the algorithm (including for simplicity that $X_1(\omega) = x_1$ for all $\omega \in \Omega$), let $\mathcal{G}_1$ denote the sub-σ-algebra of $\mathcal{G}$ corresponding to the initial conditions and, for all $k \in \mathbb{N} \setminus \{1\}$, let $\mathcal{G}_k$ denote the sub-σ-algebra of $\mathcal{G}$ corresponding to the initial conditions and $\{G_1(\omega), \ldots, G_{k-1}(\omega)\}$. In this manner, the sequence $\{\mathcal{G}_k\}$ is a filtration. For our analysis, with respect to this filtration, we make the following assumption. For the sake of brevity, here and for the remainder of our analysis, we drop from our notation the dependence of the stochastic process on an element ω of Ω, since this dependence remains obvious.

Assumption 4.2. *For all $k \in \mathbb{N}$, it holds that $\mathbb{E}[G_k | \mathcal{G}_k] = \nabla f(X_k)$. In addition, there exists $\sigma \in \mathbb{R}_{\geq 0}$ such that, for all $k \in \mathbb{N}$, one has $\|P(G_k - \nabla f(X_k))\|_2 \leq \sigma$.*

Assumption 4.2 amounts to the stochastic-gradient estimators being unbiased with bounded error. This is consistent with the stochastic setting in [7]. Looser assumptions are possible for stochastic-gradient-based methods in the unconstrained setting, but for the context of stochastic-gradient-based interior-point methods we know of no analysis that can avoid a bounded-error assumption.

We also carry forward our prior assumptions, namely, Assumptions 2.1, 3.1, and 4.1. However, we strengthen Assumption 3.1 somewhat to impose additional structure on the manner in which the search direction is defined. In particular, for all $k \in \mathbb{N}$, we define the search direction D_k through the linear system

$$\begin{bmatrix} H_k & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} D_k \\ Y_k \end{bmatrix} = - \begin{bmatrix} Q_k \\ 0 \end{bmatrix}, \tag{21}$$

where the sequence $\{H_k\}$ satisfies the following assumption.

Assumption 4.3. *For all $k \in \mathbb{N}$, the matrix $H_k \in \mathbb{S}_{\succeq 0}^n$ is $\mathcal{G}_k$-measurable and D_k satisfies (21). In addition, there exist constants $(\underline{\lambda}, \overline{\lambda}) \in (0, 1] \times \mathbb{R}_{>0}$ with $\underline{\lambda} \leq \overline{\lambda}$ such that, over any run of Algorithm 1, one finds $u^T H_k u / \|u\|^2 \in [\underline{\lambda}, \overline{\lambda}]$ for all $u \in \text{Null}(A)$.*

All combined, Assumptions 3.1 and 4.3 amount to the assumption that D_k is computed by (21) with a choice of H_k such that (5e) holds. Here, we are remarking on the fact that, under Assumption 4.3, the search direction satisfies the null-space and angle conditions imposed by the combination of (5a)–(5d). Indeed, one finds

$$D_k = -Z(Z^T H_k Z)^{-1} Z^T Q_k, \tag{22}$$

where $Z \in \mathbb{R}^{n \times (n-l)}$ is an orthogonal matrix whose columns form an orthonormal basis for $\text{Null}(A)$, through which one can in turn show that $AD_k = 0$ along with

$$\|D_k\|_2 \in [\overline{\lambda}^{-1} \|PQ_k\|_2, \underline{\lambda}^{-1} \|PQ_k\|_2] \text{ and } -(PQ_k)^T D_k \geq \underline{\lambda}\overline{\lambda}^{-1} \|PQ_k\|_2 \|D_k\|_2. \tag{23}$$

Hence, we employ $(\underline{\lambda}, \overline{\lambda})$ and (23) in place of $(\underline{\zeta}, \overline{\zeta}, \zeta)$ and (5b)–(5d) in this section. For our analysis, it is worthwhile to emphasize that the tuple of parameters $(\eta, \bar{\eta}, \underline{\lambda}, \overline{\lambda})$ is presumed to be determined uniquely for any given instance of problem (1). Therefore, the parameters in Assumption 3.1 (i.e., in (5)) and (as explicitly stated) in Assumption 4.3 are presumed to be uniform over all possible runs of the algorithm. Similarly, for Assumption 4.1 in this section, the set $\mathcal{X}$ and stated constants are assumed to be uniform over all possible realizations of $\{X_k\}$.

Our convergence guarantees for this stochastic setting rely on Parameter Rule 2, stated below, that we employ for choosing step-size- and neighborhood-enforcement parameter sequences, namely, $\{\alpha_k\}$ and $\{(\gamma_{k,\min}, \gamma_k, \gamma_{k,\max})\}$. Like in the deterministic setting, in this stochastic setting, the step sizes depend on a parameter $t_\alpha \in (-\infty, 0)$, with one difference being that the choice $t_\alpha = 0$ is not allowed for the stochastic setting. However, unlike in the deterministic setting, in this stochastic setting we prescribe a particular dependence of the neighborhood-enforcement values $\{(\gamma_{k,\min}, \gamma_k, \gamma_{k,\max})\}$ on the barrier-parameter sequence $\{\mu_k\}$ and neighborhood-parameter sequence $\{\theta_k\}$, which in turn are determined by parameters $(t_\mu, t_\theta) \in (-\infty, 0) \times (-\infty, 0)$. Whereas in the deterministic setting our ultimate requirements for the tuple

$(t_\mu, t_\theta, t_\alpha)$ (specified between Parameter Rule 1 and Theorem 4.1) were that $t_\mu = t_\theta < 0$, $t_\alpha \leq 0$, and $t_\mu + t_\alpha \in [-1, 0)$, the requirements for these parameters is stricter in this stochastic setting. For the sake of simplicity, rather than introduce separate parameters t_μ and t_θ for $\{\mu_k\}$ and $\{\theta_k\}$, respectively, in Parameter Rule 2 we introduce a single parameter t. This is reasonable since, even in the deterministic setting, our analysis requires $t_\mu = t_\theta$. It is worthwhile to emphasize upfront that the restrictions of the rule are satisfiable; e.g., one may consider $t = -0.99$ and $t_\alpha = -0.01$ as one acceptable setting.

Parameter 2. *Suppose the following are given:*

- $(t, t_\alpha) \in (-\infty, 0)^2$ *such that* $t + t_\alpha \in [-1, 0)$, $2t + t_\alpha < -1$, *and* $t + 2t_\alpha < -1$;

- $\mu_1 \in \mathbb{R}_{>0}$ *and* $\theta_0 \in \mathbb{R}_{>0}$; *and*

- $\gamma_{\text{buff}} \in \mathbb{R}_{\geq 0}$ *and* $\{\gamma_{k,\text{buff}}\} \subset \mathbb{R}_{\geq 0}$ *such that* $\gamma_{k,\text{buff}} \leq \gamma_{\text{buff}} k^t$ *for all* $k \in \mathbb{N}$.

Algorithm 1 employs, for all $k \in \mathbb{N}$, the following:

- $\mu_k = \mu_1 k^t$ *and* $\theta_{k-1} = \theta_0 k^t$;

- $\alpha_k \leftarrow k^{t_\alpha} \left(\dfrac{\underline{\lambda}^2}{\overline{\lambda} L_k} \right)$, *where* $L_k \in \mathbb{R}_{>0}$ *is defined as in Parameter Rule 1; and*

- $\gamma_k \leftarrow \min_{i \in [m]} \gamma_{k,i}$ *where*

$$\gamma_{k,i} \leftarrow \min\{\tilde{\gamma}_{k,i}, \gamma_{k,\max}\} \quad \text{for all } i \in [m],$$

$$\tilde{\gamma}_{k,i} \leftarrow \frac{-\nabla c_i(x_k)^T d_k + \sqrt{(\nabla c_i(x_k)^T d_k)^2 + 2L_{\nabla c_i} \|d_k\|_2^2 (-c_i(x_k) - \theta_k)}}{L_{\nabla c_i} \alpha_k \|d_k\|_2^2}$$

$$\text{for all } i \in [m],$$

$$\beta^\sigma \leftarrow \underline{\lambda}^{-1} \left(\kappa_{\nabla f} + \sigma + \frac{\mu_1}{\theta_0} \sum_{j \in [m]} \kappa_{\nabla c_j} \right),$$

$$\gamma_{k,i,\min} \leftarrow \min \left\{ 1, \frac{-\kappa_{\nabla c_i} + \sqrt{\kappa_{\nabla c_i}^2 + 2L_{\nabla c_i}(\eta \mu_k - \theta_k)}}{\alpha_k \beta^\sigma L_{\nabla c_i}}, \frac{\eta - \theta_k}{\alpha_k \beta^\sigma L_{\nabla c_i}} \right\}$$

$$\text{for all } i \in [m],$$

$$\gamma_{k,\min} \leftarrow \min_{i \in [m]} \gamma_{k,i,\min}, \quad \text{and}$$

$$\gamma_{k,\max} \leftarrow \min\{1, \gamma_{k,\min} + \gamma_{k,\text{buff}}\}.$$

The strategy for selecting $\{\alpha_k\}$ in Parameter Rule 2 is consistent with that in Parameter Rule 1. Observe that since $\{\mu_k\}$ and $\{\theta_k\}$ are set in advance of any run of the algorithm, it follows under Parameter Rule 2 that $\{L_k\}$ and so $\{\alpha_k\}$ are determined in advance of a run of the algorithm as well. Thus, our analysis can consider $\{\mathcal{A}_k\} = \{\alpha_k\}$, where $\{\alpha_k\}$ is a prescribed sequence that is uniform over all runs of Algorithm 1 employed for any given instance of problem (1). The choice of $\{(\gamma_{k,\min}, \gamma_k, \gamma_{k,\max})\}$ in Parameter Rule 2, on the other hand, is more stringent than that in Parameter Rule 1. Observe that Parameter Rule 1 amounts to choosing $\gamma_k \leftarrow \min_{i \in [m]} \gamma_{k,i}$ for all $k \in \mathbb{N}$, where $\gamma_{k,i} \leftarrow \min\{\tilde{\gamma}_{k,i}, 1\}$ for all $(k, i) \in \mathbb{N} \times [m]$. The choice in Parameter Rule 2 is similar, but with $\gamma_{k,i} \leftarrow \min\{\tilde{\gamma}_{k,i}, \gamma_{k,\max}\}$ for all $(k, i) \in \mathbb{N} \times [m]$, where for any given $k \in \mathbb{N}$ the upper limit $\gamma_{k,\max} \in (0, 1]$ might be less than 1. The particular strategy for setting $\gamma_{k,\max}$ for all $k \in \mathbb{N}$ in Parameter Rule 2 involves first setting a lower value $\gamma_{k,\min} \in \mathbb{R}_{>0}$ for each $k \in \mathbb{N}$ *that is defined only by values that are set in advance of a run of the algorithm*. This fact is critical, since it ensures that $\{(\gamma_{k,\min}, \gamma_{k,\max})\}$ is prescribed and uniform over all runs of Algorithm 1 employed for any given instance of problem (1). For each $(k, i) \in \mathbb{N} \times [m]$, the values $\tilde{\gamma}_{k,i}$ and γ_k may differ between runs, but importantly the sequence $\{(\gamma_{k,\min}, \gamma_{k,\max})\}$ does not.

Many of our prior results from the deterministic setting carry over here to our present analysis for the stochastic setting. Firstly, Lemma 3.1 holds, as does Lemma 4.1 since it has been proved irrespective of any particular algorithm. Secondly, following the arguments in Lemmas 4.2 and 4.3, it follows that Parameter Rule 2 ensures that $\gamma_{k,\min} \leq \gamma_k \leq \gamma_{k,\max}$ for all $k \in \mathbb{N}$ in any given run of the algorithm. We formalize this in Lemma 4.5 below, the proof of which is omitted since it would follow the same lines of arguments as in the proofs of Lemmas 4.2 and 4.3. The only significant difference that would be in the proof is that the bound $\|P\nabla f(x_k)\|_2 \leq \kappa_{\nabla f}$ needs to be replaced by $\|PG_k\|_2 \leq \kappa_{\nabla f} + \sigma$ for all $k \in \mathbb{N}$, which follows under Assumption 4.2. This is the reason that β^σ in Parameter Rule 2 involves an additional σ term, as opposed to β in Lemma 4.3, which does not. This affects the denominators in the expression for $\gamma_{k,i,\min}$ for all $(k,i) \in \mathbb{N} \times [m]$. Otherwise, $\gamma_{k,i,\min}$ is consistent between Lemma 4.3 and Parameter Rule 2.

Lemma 4.5. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, for all $k \in \mathbb{N}$, the choice of $\Gamma_k \in (0, \gamma_{k,\max}]$ in Parameter Rule 2 ensures that $[X_k, X_k + \Gamma_k \alpha_k D_k] \subseteq \mathcal{N}(\theta_k)$, i.e., the condition in Step 5 of Algorithm 1 holds. Consequently, under Parameter Rule 2, Algorithm 1 is well defined in the sense that it will generate an infinite sequence of iterates. Moreover, Parameter Rule 2 ensures that $\|D_k\|_2 \leq \beta^\sigma$ and $\Gamma_k \in [\gamma_{k,\min}, \gamma_{k,\max}]$ for all $k \in \mathbb{N}$.*

Now let us turn to results for the stochastic setting that are distinct from those for the deterministic setting. Our first main goal is to prove an upper bound on the expected decrease in the shifted barrier-augmented function that occurs with each iteration. Toward this end, let us begin with the following preliminary bound.

Lemma 4.6. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, for all $k \in \mathbb{N}$, one finds that*

$$
\begin{aligned}
&\tilde{\phi}(X_{k+1}, \mu_{k+1}) - \tilde{\phi}(X_k, \mu_k) \\
&\leq -\Gamma_k \alpha_k \overline{\lambda}^{-1} \|P\nabla_x \tilde{\phi}(X_k, \mu_k)\|_2^2 + \tfrac{1}{2}\Gamma_k^2 \alpha_k^2 L_k \underline{\lambda}^{-2} \|PQ_k\|_2^2 \\
&\quad + \Gamma_k \alpha_k \nabla_x \tilde{\phi}(X_k, \mu_k)^T Z(Z^T H_k Z)^{-1} Z^T (\nabla_x \tilde{\phi}(X_k, \mu_k) - Q_k).
\end{aligned}
$$

Proof. Consider arbitrary $k \in \mathbb{N}$. By Lemmas 4.1 and 4.5, (6), (5), (22), and the fact that $\|Z^T \nabla_x \tilde{\phi}(x_k, \mu_k)\|_2 = \|P\nabla_x \tilde{\phi}(x_k, \mu_k)\|_2$, it follows with $L_k \in \mathbb{R}_{>0}$ defined as in Parameter Rule 2 (and 1) that

$$
\begin{aligned}
&\tilde{\phi}(X_{k+1}, \mu_k) - \tilde{\phi}(X_k, \mu_k) \\
&\leq \nabla_x \tilde{\phi}(X_k, \mu_k)^T (X_{k+1} - X_k) + \tfrac{1}{2} L_k \|X_{k+1} - X_k\|_2^2 \\
&= \Gamma_k \alpha_k \nabla_x \tilde{\phi}(X_k, \mu_k)^T D_k + \tfrac{1}{2}\Gamma_k^2 \alpha_k^2 L_k \|D_k\|_2^2 \\
&= -\Gamma_k \alpha_k \nabla_x \tilde{\phi}(X_k, \mu_k)^T Z(Z^T H_k Z)^{-1} Z^T Q_k + \tfrac{1}{2}\Gamma_k^2 \alpha_k^2 L_k \|Z(Z^T H_k Z)^{-1} Z^T Q_k\|_2^2 \\
&= -\Gamma_k \alpha_k \|Z^T \nabla_x \tilde{\phi}(X_k, \mu_k)\|_{(ZH_kZ)^{-1}}^2 + \tfrac{1}{2}\Gamma_k^2 \alpha_k^2 L_k \|(Z^T H_k Z)^{-1} Z^T Q_k\|_2^2 \\
&\quad + \Gamma_k \alpha_k \nabla_x \tilde{\phi}(X_k, \mu_k)^T Z(Z^T H_k Z)^{-1} Z^T (\nabla_x \tilde{\phi}(X_k, \mu_k) - Q_k) \\
&\leq -\Gamma_k \alpha_k \overline{\lambda}^{-1} \|P\nabla_x \tilde{\phi}(X_k, \mu_k)\|_2^2 + \tfrac{1}{2}\Gamma_k^2 \alpha_k^2 L_k \underline{\lambda}^{-2} \|PQ_k\|_2^2 \\
&\quad + \Gamma_k \alpha_k \nabla_x \tilde{\phi}(X_k, \mu_k)^T Z(Z^T H_k Z)^{-1} Z^T (\nabla_x \tilde{\phi}(X_k, \mu_k) - Q_k).
\end{aligned}
$$

Finally, since $\mu_{k+1} \leq \mu_k$ implies $\tilde{\phi}(X_{k+1}, \mu_{k+1}) \leq \tilde{\phi}(X_{k+1}, \mu_k)$ (see the discussion following (7)), the desired conclusion follows. $\qquad\square$ $\hfill\square$

Lemma 4.6 shows that the change in the shifted barrier-augmented function with each iteration is bounded above by the sum of a negative term and two terms that may be considered noise terms. The goal of our next two lemmas is to bound these noise terms. We start with the second term on the right-hand side in Lemma 4.6.

Lemma 4.7. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, for all $k \in \mathbb{N}$, one finds that*

$$\tfrac{1}{2}\Gamma_k^2\alpha_k^2 L_k\underline{\lambda}^{-2}\|PQ_k\|_2^2 \leq \tfrac{3}{4}\Gamma_k^2\alpha_k^2 L_k\underline{\lambda}^{-2}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \tfrac{3}{2}\alpha_k^2 L_k\underline{\lambda}^{-2}\sigma^2.$$

Proof. Consider arbitrary $k \in \mathbb{N}$. Observe that, for any $(a,b) \in \mathbb{R}^n \times \mathbb{R}^n$,

$$0 \leq \|P(\tfrac{1}{2}a - b)\|_2^2 = \tfrac{1}{4}\|Pa\|_2^2 - a^T P^T Pb + \|Pb\|_2^2,$$

from which it follows that

$$\|P(a+b)\|_2^2 = \|Pa\|_2^2 + 2a^T P^T Pb + \|Pb\|_2^2 \leq \tfrac{3}{2}\|Pa\|_2^2 + 3\|Pb\|_2^2.$$

Consequently, with (6) it holds that

$$\begin{aligned}
\tfrac{1}{2}\|PQ_k\|_2^2 &= \tfrac{1}{2}\|P(\nabla_x\tilde{\phi}(X_k,\mu_k) + Q_k - \nabla_x\tilde{\phi}(X_k,\mu_k))\|_2^2 \\
&\leq \tfrac{3}{4}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \tfrac{3}{2}\|P(Q_k - \nabla_x\tilde{\phi}(X_k,\mu_k))\|_2^2 \\
&\leq \tfrac{3}{4}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \tfrac{3}{2}\sigma^2.
\end{aligned}$$

Therefore, since $\Gamma_k \leq 1$ under Parameter Rule 2, the proof is complete. $\square$ $\square$

The next lemma provides an upper bound on a conditional expectation of the last term on the right-hand side of the inequality in Lemma 4.6.

Lemma 4.8. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, for all $k \in \mathbb{N}$, one finds that*

$$\mathbb{E}[\Gamma_k\alpha_k\nabla_x\tilde{\phi}(X_k,\mu_k)^T Z(Z^T H_k Z)^{-1}Z^T(\nabla_x\tilde{\phi}(X_k,\mu_k) - Q_k)|\mathcal{G}_k] \leq \gamma_{k,\mathrm{buff}}\alpha_k B^\sigma,$$

where

$$B^\sigma := \underline{\lambda}^{-1}\left(\kappa_{\nabla f} + \frac{\mu_1}{\theta_0}\sum_{j\in[m]}\kappa_{\nabla c_j}\right)\sigma.$$

Proof. Consider arbitrary $k \in \mathbb{N}$. The inner product in the expected value of interest may be nonnegative or nonpositive. Hence, let us invoke the Law of Total Expectation in order to handle each case separately. Let $\mathcal{I}_k$ be the event that $I_k \geq 0$, where $I_k := \nabla_x\tilde{\phi}(X_k,\mu_k)^T Z(Z^T H_k Z)^{-1}Z^T(\nabla_x\tilde{\phi}(X_k,\mu_k) - Q_k)$, and let $\mathcal{I}_k^c$ be the complementary event that $I_k < 0$. By Parameter Rule 2, the fact that $\mathbb{E}[I_k|\mathcal{G}_k] = 0$ under Assumption 4.2, and Lemma 4.5, it follows that

$$\begin{aligned}
&\mathbb{E}[\Gamma_k\alpha_k I_k|\mathcal{G}_k] \\
&= \mathbb{E}[\Gamma_k\alpha_k I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] + \mathbb{E}[\Gamma_k\alpha_k I_k|\mathcal{G}_k \wedge \mathcal{I}_k^c]\mathbb{P}[\mathcal{I}_k^c|\mathcal{G}_k] \\
&\leq \gamma_{k,\max}\alpha_k\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] + \gamma_{k,\min}\alpha_k\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k^c]\mathbb{P}[\mathcal{I}_k^c|\mathcal{G}_k] \\
&\leq \gamma_{k,\min}\alpha_k(\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] + \mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k^c]\mathbb{P}[\mathcal{I}_k^c|\mathcal{G}_k]) \\
&\quad + \gamma_{k,\mathrm{buff}}\alpha_k\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] \\
&= \gamma_{k,\mathrm{buff}}\alpha_k\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k]. \qquad (24)
\end{aligned}$$

Furthermore, by Assumptions 2.1–4.3, the Cauchy–Schwarz inequality, the triangle inequality, $\|P\|_2 \leq 1$, and the fact that $\|Z^T q\|_2 = \|Pq\|_2$ for any $q \in \mathbb{R}^n$ one has

$$\begin{aligned}
&\mathbb{E}[I_k|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] \\
&= \mathbb{E}[\nabla_x\tilde{\phi}(X_k,\mu_k)^T Z(Z^T H_k Z)^{-1}Z^T(\nabla_x\tilde{\phi}(X_k,\mu_k) - Q_k)|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k] \\
&\leq \underline{\lambda}^{-1}\mathbb{E}[\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2\|P(\nabla_x\tilde{\phi}(X_k,\mu_k) - Q_k)\|_2|\mathcal{G}_k \wedge \mathcal{I}_k]\mathbb{P}[\mathcal{I}_k|\mathcal{G}_k]
\end{aligned}$$

$$\leq \underline{\lambda}^{-1}\left(\kappa_{\nabla f} + \frac{\mu_1}{\theta_0}\sum_{j\in[m]}\kappa_{\nabla c_j}\right)\sigma.$$

Combining this inequality with (24) yields the desired conclusion. $\qquad\square$ $\qquad\square$

Combining the preliminary bound in Lemma 4.6 with the previous two lemmas, we are now prepared to prove an upper bound on the expected decrease in the shifted barrier-augmented function that occurs with each iteration.

Lemma 4.9. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, for all $k\in\mathbb{N}$, one finds that*

$$\mathbb{E}[\tilde{\phi}(X_{k+1},\mu_{k+1})|\mathcal{G}_k] - \tilde{\phi}(X_k,\mu_k)$$
$$\leq -\tfrac{1}{4}\gamma_{k,\min}\alpha_k\overline{\lambda}^{-1}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \tfrac{3}{2}\alpha_k^2 L_k\underline{\lambda}^{-2}\sigma^2 + \gamma_{k,\mathrm{buff}}\alpha_k B^\sigma,$$

where B^σ is defined as in Lemma 4.8.

Proof. Consider arbitrary $k\in\mathbb{N}$. One finds from Lemmas 4.6, 4.7, and 4.8 that

$$\mathbb{E}[\tilde{\phi}(X_{k+1},\mu_{k+1})|\mathcal{G}_k] - \tilde{\phi}(X_k,\mu_k)$$
$$\leq -\Gamma_k\alpha_k(\overline{\lambda} - \tfrac{3}{4}\Gamma_k\alpha_k L_k\underline{\lambda}^{-2})\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \tfrac{3}{2}\alpha_k^2 L_k\underline{\lambda}^{-2}\sigma^2 + \gamma_{k,\mathrm{buff}}\alpha_k B^\sigma.$$

Then, under Parameter Rule 2, one has $\Gamma_k \leq 1$ and $k^{t_\alpha} \leq 1$, so

$$\alpha_k = k^{t_\alpha}\underline{\lambda}^2\overline{\lambda}^{-1}L_k^{-1} \implies \overline{\lambda} - \tfrac{3}{4}\Gamma_k\alpha_k L_k\underline{\lambda}^{-2} = \overline{\lambda}^{-1} - \tfrac{3}{4}\overline{\lambda}^{-1}\Gamma_k k^{t_\alpha} \geq \tfrac{1}{4}\overline{\lambda}^{-1},$$

which when combined with the prior conclusion completes the proof. $\qquad\square$ $\qquad\square$

Using this sufficient decrease result and looking further into the selections in Parameter Rule 2, we obtain the following result.

Lemma 4.10. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then, $(\nu,\varsigma)\in\mathbb{R}_{>0}\times\mathbb{R}_{>0}$ exists such that, for all $k\in\mathbb{N}$,*

$$\mathbb{E}[\tilde{\phi}(X_{k+1},\mu_{k+1})|\mathcal{G}_k] - \tilde{\phi}(X_k,\mu_k)$$
$$\leq -\nu k^{t+t_\alpha}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2 + \varsigma k^{\max\{2t+t_\alpha, t+2t_\alpha\}}.$$

Proof. We prove the result by building on Lemma 4.9. By Parameter Rule 2, there exist $\kappa\in\mathbb{R}_{>0}$, $\underline{L}\in\mathbb{R}_{>0}$, and $L\in\mathbb{R}_{>0}$ such that for all $k\in\mathbb{N}$ one finds

$$\gamma_{k,\min}\alpha_k = \min\left\{\alpha_k, \frac{-\kappa + \sqrt{\kappa^2 + 2\underline{L}(\eta\mu_k - \theta_k)}}{\beta^\sigma L}, \frac{\eta - \theta_k}{\beta^\sigma L}\right\}.$$

Since this is the same type of lower bound as in (15), the same argument in the proof of Theorem 4.1 shows that there exists $\nu\in\mathbb{R}_{>0}$ such that $\tfrac{1}{4}\gamma_{k,\min}\alpha_k\overline{\lambda}^{-1} \geq \nu k^{t+t_\alpha}$ for all $k\in\mathbb{N}$. Hence, by Lemma 4.9, all that remains is to prove that there exists $\varsigma\in\mathbb{R}_{>0}$ such that for all $k\in\mathbb{N}$ one finds that

$$\tfrac{3}{2}\alpha_k^2 L_k\underline{\lambda}^{-2}\sigma^2 + \gamma_{k,\mathrm{buff}}\alpha_k B^\sigma \leq \varsigma k^{\max\{2t+t_\alpha, t+2t_\alpha\}}. \tag{25}$$

Starting with the first term in (25), one finds that

$$\tfrac{3}{2}\alpha_k^2 L_k\underline{\lambda}^{-2}\sigma^2 = \tfrac{3}{2}\alpha_k(\alpha_k L_k\underline{\lambda}^{-2})\sigma^2 = \tfrac{3}{2}\underline{\lambda}^2\overline{\lambda}^{-2}\sigma^2\frac{k^{2t_\alpha}}{L_k},$$

where with $\upsilon := \sum_{i=1}^{m} (L_{c_i}\kappa_{\nabla c_i} + \kappa_{c_i}L_{\nabla c_i})$ one finds that

$$\frac{k^{2t_\alpha}}{L_k} = \frac{k^{2t_\alpha}}{L_{\nabla f} + \mu_1 k^t \theta_0^{-2} k^{-t}(k+1)^{-t}\upsilon} \leq \frac{k^{2t_\alpha}}{L_{\nabla f} + \mu_1\theta_0^{-2}\upsilon k^{-t}} \leq \frac{k^{t+2t_\alpha}}{\mu_1\theta_0^{-2}\upsilon}.$$

As for the second term in (25), one similarly finds that

$$\gamma_{k,\text{buff}}\alpha_k B^\sigma \leq \gamma_{\text{buff}}k^t k^{t_\alpha}\left(\frac{\underline{\lambda}^2 B^\sigma}{\overline{\lambda}^2 L_k}\right) \leq \left(\frac{\gamma_{\text{buff}}\underline{\lambda}^2 B^\sigma}{\mu_1\theta_0^{-2}\upsilon\overline{\lambda}^2}\right)k^{2t+t_\alpha}.$$

Combining these bounds yields (25) for some $\varsigma \in \mathbb{R}_{>0}$. $\qquad\square$ $\qquad\square$

Our final theorem, presented next, now follows similarly to Theorem 3.16 in [7]. We provide a complete proof for the sake of completeness.

Theorem 4.2. *Suppose that Assumptions 2.1, 3.1, 4.1, 4.2, and 4.3 hold and that Algorithm 1 is run with Parameter Rule 2. Then,*
$$\liminf_{k\to\infty} \|P\nabla_x\phi(X_k,\mu_k)\|_2^2 = 0 \quad \text{almost surely.}$$

Consequently, if in a given run there exists $\mathcal{K} \subseteq \mathbb{N}$ with $|\mathcal{K}| = \infty$ such that

- $\{P\nabla_x\phi(x_k,\mu_k)\}_{k\in\mathcal{K}} \to 0$,

- $\{x_k\}_{k\in\mathcal{K}} \to \overline{x}$ *for some $\overline{x} \in \mathcal{F}$, and*

- *at $\overline{x}$ the linear independence constraint qualification (LICQ) holds with respect to problem (1) in the sense that with $\overline{\mathcal{A}} := \{i \in [m] : c_i(\overline{x}) = 0\}$ the columns of A^T combined with the vectors in $\{\nabla c_i(\overline{x})\}_{i\in\overline{\mathcal{A}}}$ form a linearly independent set,*

then $\overline{x}$ is a KKT point for problem (1) in the sense that there exists a pair of Lagrange multipliers $(\overline{y},\overline{z}) \in \mathbb{R}^l \times \mathbb{R}^m$ such that $(\overline{x},\overline{y},\overline{z})$ satisfies (3).

Proof. It follows from Lemma 4.10 and the Law of Total Expectation that there exists $(\nu,\varsigma) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ such that, for all $k \in \mathbb{N}$, one finds that

$$\mathbb{E}[\tilde{\phi}(X_{k+1},\mu_{k+1})] - \mathbb{E}[\tilde{\phi}(X_k,\mu_k)]$$
$$\leq -\nu k^{t+t_\alpha}\mathbb{E}[\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2] + \varsigma k^{\max\{2t+t_\alpha,t+2t_\alpha\}}.$$

Consider arbitrary $K \in \mathbb{N}$. Summing over $k \in [K]$ yields

$$f_{\inf} - \mathbb{E}[\tilde{\phi}(x_1,\mu_1)] \leq \mathbb{E}[\tilde{\phi}(X_{K+1},\mu_{K+1})] - \mathbb{E}[\tilde{\phi}(x_1,\mu_1)]$$
$$\leq -\nu\sum_{k=1}^{K} k^{t+t_\alpha}\mathbb{E}[\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2] + \varsigma\sum_{k=1}^{K} k^{\max\{2t+t_\alpha,t+2t_\alpha\}}.$$

After rearrangement, this yields

$$\sum_{k=1}^{K} k^{t+t_\alpha}\mathbb{E}[\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2] \leq \frac{1}{\nu}(\mathbb{E}[\tilde{\phi}(x_1,\mu_1)] - f_{\inf}) + \frac{\varsigma}{\nu}\sum_{k=1}^{K} k^{\max\{2t+t_\alpha,t+2t_\alpha\}}.$$

Since $2t + t_\alpha \in (-\infty,-1)$ and $t + 2t_\alpha \in (-\infty,-1)$, the right-hand side of this inequality converges to a finite limit as $K \to \infty$. Furthermore, by $\sum_{k=1}^{\infty} k^{t+t_\alpha} = \infty$, nonnegativity of $\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2$, and Fatou's lemma, one almost surely has

$$0 = \liminf_{k\to\infty}\mathbb{E}[\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2] \geq \mathbb{E}\left[\liminf_{k\to\infty}\|P\nabla_x\tilde{\phi}(X_k,\mu_k)\|_2^2\right] = 0.$$

Now consider $\Phi := \liminf_{k\to\infty} \|P\nabla_x\tilde{\phi}(X_k, \mu_k)\|_2^2$, the expectation of which has been shown above to be zero. Nonnegativity of $\|P\nabla_x\tilde{\phi}(X_k, \mu_k)\|_2^2$ for all $k \in \mathbb{N}$ and the Law of Total Expectation imply that $0 = \mathbb{E}[\Phi] \geq \mathbb{P}[\Phi > 0]\mathbb{E}[\Phi|\Phi > 0]$, so $0 = \mathbb{P}[\Phi > 0] = \mathbb{P}[\liminf_{k\to\infty} \|P\nabla_x\tilde{\phi}(X_k, \mu_k)\|_2^2 > 0]$. This is the first desired conclusion of the theorem. The second desired conclusion, on the other hand, follows by using the same argument as in the proof of Theorem 4.1. $\qquad\square$ $\qquad\square$

5 Discussion of Assumptions 3.1 and 4.3

Our aim in this section is to discuss Assumptions 3.1 and 4.3. In particular, first for the deterministic setting, we provide (a) example problems for which Assumption 3.1 can be shown to hold or cannot be shown to hold, (b) further explanation about why Assumption 3.1 can be viewed as a combination of a nondegeneracy assumption and an assumption about μ_1/θ_0 being sufficiently large, and (c) a procedure for computing d_k to satisfy (5) when such a direction exists. We follow this with a discussion about the combination of Assumptions 3.1 and 4.3 for the stochastic setting. We emphasize upfront that, generally speaking, *a straightforward procedure for computing d_k to satisfy the conditions of our convergence analysis (for either the deterministic or stochastic settings) may not be available in practice.* For example, for one thing, determining a sufficiently large value for the ratio μ_1/θ_0 may rely on problem-specific constants that are not known in advance of a run of the algorithm. That being said, in this section we provide guidance for how to compute d_k in practice such that, along with parameter tuning (as is common for stochastic-gradient-based methods in practice), one obtains good practical performance.

Throughout this section, for the sake of clarity, we drop the subscript notation that indicates the iteration index of an algorithm—rather, in this section, a subscript only refers to an index of a component of a vector. In particular, in this section, we denote $x \equiv x_k$ (which in turn means, e.g., that x_1 refers to the first component of the vector x), $\mu \equiv \mu_k$, $\theta \equiv \theta_{k-1}$, $q \equiv q_k$, and $d \equiv d_k$.

Let us begin with some example problems for the deterministic setting. Firstly, recall that under Assumption 2.1 a strictly feasible point must exist. This means that one can rule out problems with, e.g., the affine constraint $x_1 = 0$ and the inequality constraint $x_1 \leq 0$, or any problems with such a combination of constraints that implies the lack of a strictly feasible point. Secondly, observe that the existence or not of a direction satisfying (5) depends on the feasible region reduced to $\text{Null}(A)$. This means that one can distinguish cases of whether (5) holds or not by presuming that one has already restricted attention to this reduced space and consider only the geometry of the feasible region with respect to inequality constraints where the ambient space of the variables is this reduced space.

Let us now present an example for which Assumption 3.1 can be shown to hold. In particular, the following example shows that there exists a value of the tuple $(\mu, \theta, \eta, \underline{\eta}, \zeta, \overline{\zeta}, \zeta)$ under the ranges specified by Algorithm 1 such that at all $x \in \mathcal{F}_{<0}$ there exists d satisfying (5). The main aspect of the example is that μ must be sufficiently large relative to θ. We remark that the case $a \in \mathbb{R}_{>0}$ that is considered in the example is the more challenging case to consider. Indeed, if $a \in \mathbb{R}_{\leq 0}$, then, using similar arguments, the parameter requirements are less restrictive than in the following example. We expound on this claim further after the example.

Example 1. *To start, let $(\mu, \theta, \eta, \underline{\eta}, \zeta, \overline{\zeta}, \zeta)$ be arbitrary positive constants under the ranges specified by Algorithm 1. (A further restriction on the relationship between μ and θ is developed during the discussion of this example, when it is needed.) Consider a two-dimensional problem where $f(x) = v^T x$ for some $v \in \mathbb{R}^2$, $c_1(x) = -x_1$, $c_2(x) = ax_1 - x_2$ for some $a \in \mathbb{R}_{>0}$, and $g = \nabla f(x) = v$ (see (4)), so*

$$q = v + \mu\left(\frac{1}{-c_1(x)}\right)\nabla c_1(x) + \mu\left(\frac{1}{-c_2(x)}\right)\nabla c_2(x). \tag{26}$$

Defining $\mathcal{A}(x) := \{i \in \{1, 2\} : -\eta\mu < c_i(x) \leq -\theta\}$, the conditions (5) reduce to

$$\underline{\zeta}\|q\|_2 \leq \|d\|_2 \leq \overline{\zeta}\|q\|_2, \tag{27a}$$

$$-q^T d \geq \zeta\|q\|_2\|d\|_2, \tag{27b}$$

$$\text{and} \quad \nabla c_i(x)^T d \le -\tfrac{1}{2}\underline{\eta}\|d\|_2 \quad \text{for all} \ \ i \in \mathcal{A}(x). \tag{27c}$$

Since the gradient vectors $\nabla f(x)$, $\nabla c_1(x)$, and $\nabla c_2(x)$ are constant over $x \in \mathcal{F}_{<0}$ and the latter two gradient vectors are nonzero, it follows from (26) *that for any $x \in \mathcal{F}_{<0}$ one finds that $q/\|q\|_2 \to q_{\lim}/\|q_{\lim}\|_2$ as $\mu \to \infty$, where $q_{\lim} = \nabla c_1(x)/(-c_1(x)) + \nabla c_2(x)/(-c_2(x))$. At the same time, for any pair of constants $(\chi_1, \chi_2) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ and with the search direction defined by*

$$d = -v - \mu\left(\frac{\chi_1}{-c_1(x)}\right)\nabla c_1(x) - \mu\left(\frac{\chi_2}{-c_2(x)}\right)\nabla c_2(x), \tag{28}$$

one has for any $x \in \mathcal{F}_{<0}$ that $d/\|d\|_2 \to d_{\lim}/\|d_{\lim}\|_2$ as $\mu \to \infty$, where the limiting direction is $d_{\lim} = (\chi_1/c_1(x))\nabla c_1(x) + (\chi_2/c_2(x))\nabla c_2(x)$.

Now suppose that the constants (μ, θ, η) and point $x \in \mathcal{F}_{<0}$ are such that both constraints are nearly active at x, i.e., $\mathcal{A}(x) = \{1, 2\}$. Since one has $\chi_3 := (-\nabla c_1(x)^T \nabla c_2(x))/(\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2) = a/\sqrt{1+a^2} \in (0,1)$, it follows that

$$\nabla c_1(x)^T d_{\lim} = \chi_1 \underbrace{\frac{\|\nabla c_1(x)\|_2^2}{c_1(x)}}_{<0} + \chi_2\chi_3 \underbrace{\frac{\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2}{-c_2(x)}}_{>0}$$

$$\text{and} \quad \nabla c_2(x)^T d_{\lim} = \chi_1\chi_3 \underbrace{\frac{\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2}{-c_1(x)}}_{>0} + \chi_2 \underbrace{\frac{\|\nabla c_2(x)\|_2^2}{c_2(x)}}_{<0}.$$

Now suppose $\chi_1 = c_1(x)\|\nabla c_2(x)\|_2/(c_2(x)\|\nabla c_1(x)\|_2)$ and $\chi_2 = 1$. (The effect of these choices is to normalize the contributions of the two terms in the definition of $d_{\lim}$.) These choices and the fact that $\eta = \psi\theta/\mu$ for some $\psi \in \mathbb{R}_{>1}$ implies that

$$\nabla c_1(x)^T d_{\lim} < -\left(\frac{1-\chi_3}{\psi\theta}\right)\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2$$

$$\text{and} \quad \nabla c_2(x)^T d_{\lim} < -\left(\frac{1-\chi_3}{\psi\theta}\right)\|\nabla c_2(x)\|_2^2.$$

On the other hand, one has that $\|d_{\lim}\|_2 \le 2\|\nabla c_2(x)\|_2/\theta$. Thus, one may conclude that there exists $\chi_4 \in \mathbb{R}_{>0}$ such that $\nabla c_1(x)^T d_{\lim} \le -\chi_4\|d_{\lim}\|_2$ and $\nabla c_2(x)^T d_{\lim} \le -\chi_4\|d_{\lim}\|_2$ for all $x \in \mathcal{F}_{<0}$ such that both constraints are nearly active.

On the other hand, with $d_{\lim}$ defined as in the previous paragraph (i.e., with $\chi_1 = c_1(x)\|\nabla c_2(x)\|_2/(c_2(x)\|\nabla c_1(x)\|_2)$ and $\chi_2 = 1$), it follows that

$$-q_{\lim}^T d_{\lim}$$
$$= \left(\frac{1}{c_1(x)}\nabla c_1(x) + \frac{1}{c_2(x)}\nabla c_2(x)\right)^T \left(\frac{\chi_1}{c_1(x)}\nabla c_1(x) + \frac{\chi_2}{c_2(x)}\nabla c_2(x)\right)$$
$$= \frac{1}{c_1(x)c_2(x)}\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2 + \frac{1}{c_2(x)^2}\|\nabla c_2(x)\|_2^2$$
$$\quad + \left(\frac{1}{c_1(x)c_2(x)} + \frac{\|\nabla c_2(x)\|_2}{c_2(x)^2\|\nabla c_1(x)\|_2}\right)\nabla c_1(x)^T\nabla c_2(x)$$
$$= \frac{1-\chi_3}{c_1(x)c_2(x)}\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2 + \frac{1-\chi_3}{c_2(x)^2}\|\nabla c_2(x)\|_2^2$$
$$\ge \frac{1-\chi_3}{\theta^2}(\|\nabla c_1(x)\|_2\|\nabla c_2(x)\|_2 + \|\nabla c_2(x)\|_2^2).$$

On the other hand, $\|d_{\lim}\|_2 \le 2\|\nabla c_2(x)\|_2/\theta$ and $\|q_{\lim}\|_2 \le 2\|\nabla c_2(x)\|_2/\theta$. Thus, one may conclude that there exists $\chi_5 \in \mathbb{R}_{>0}$ such that $-q_{\lim}^T d_{\lim} \ge \chi_5\|q_{\lim}\|_2\|d_{\lim}\|_2$ for all $x \in \mathcal{F}_{<0}$ such that both constraints are nearly active.

The arguments in the preceding two paragraphs involve the limiting directions $d_{\lim}$ and $q_{\lim}$, where it was shown that there exist positive constants χ_4 and χ_5—that are uniform with respect to $x \in \mathcal{F}_{<0}$—such that conditions of the form (27b)–(27c) hold for these limit directions. All that remains is to observe that, since these are the limiting directions for any given $x \in \mathcal{F}_{<0}$ as $\mu \to \infty$, such conditions and (27a) hold for (d, q) for some positive constants as long as μ is sufficiently large relative to θ. All together from our observations, it follows that with μ sufficiently large relative to θ, $\eta \in (\theta/\mu, 1)$, $x \in \mathcal{F}_{<0}$ such that both constraints are nearly active, $\underline{\eta}$ sufficiently small, ζ sufficiently small, $\overline{\zeta}$ sufficiently large, and ζ sufficiently small, one finds d in (28) with the aforementioned choices of χ_1 and χ_2 satisfies (5).

Finally, observe that for any $x \in \mathcal{F}_{<0}$ such that both constraints are not nearly active, the situation is simpler than above. In such a setting, one does not need condition (27c) to hold for both constraints; it only needs to hold for whichever constraint, if any, is nearly active. This offers much more flexibility in the choice of d. Overall, since the constraint gradients are constant over $x \in \mathcal{F}_{<0}$, one can again derive the existence of uniform parameter choices, including that μ is sufficiently large relative to θ, to ensure that d satisfying (5) exists at all such x. $\qquad\square$

The analysis in Example 1 can be extended to any problem with polyhedral constraints (for arbitrary $n \in \mathbb{N}$ and $m \in \mathbb{N}$) as long as at any strictly feasible point in $\mathcal{F}_{<0}$ the interior of the polar cone of the nearly active constraint gradients is nonempty. Intuitively, this can be understood as follows. With μ sufficiently large relative to θ, the gradient of the barrier term corresponding to nearly active constraints *pushes* the vector $-q$ (through more emphasis on the barrier term) to point with the interior of the polar cone of the nearly active constraint gradients. This means that, as long as a direction d is chosen appropriately to compensate for the different magnitudes of the norms of the constraint gradients, the desired conditions in (5) hold. Admittedly, *the parameter choices that ensure that such a direction d always exists are not straightforward to determine in advance of a run of the algorithm.* This means that, in practice, the algorithm parameters require tuning and/or adaptive choices to be made, as we discuss further shortly. To illustrate the situation that we have described in Example 1, please see Figure 1, on the left.
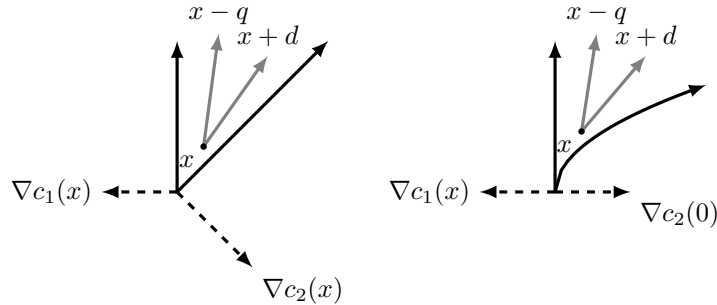

Figure 1: On the left, an illustration of Example 1 for which Assumption 3.1 holds as long as the barrier parameter μ is sufficiently large relative to θ (recall that μ_k/θ_{k-1} is constant in the algorithm) amongst other parameter choices. Since μ is sufficiently large relative to θ, it follows that a direction d pointing into the interior of the polar cone of nearly active constraint gradients is also one of sufficient decrease for the barrier-augmented objective function. On the right, an illustration of Example 2 for which Assumption 3.1 fails to hold since, as x approaches the origin from within the feasible region, there is no minimum value for the ratio μ/θ such that a direction into the interior of the polar cone of nearly active constraint gradients satisfies (5).

Now let us briefly describe an example for which we are unable to show that Assumption 3.1 holds with uniform parameter choices; see Figure 1, on the right.

Example 2. *Consider a two-dimensional problem where $f(x) = v^T x$ for some $v \in \mathbb{R}^2$, $c_1(x) = -x_1$, $c_2(x) = x_1 - x_2^2$, and $g = \nabla f(x) = v$ (see (4)). Defining, as before, $\mathcal{A}(x) := \{i \in \{1, 2\} : -\eta\mu < c_i(x) \leq -\theta\}$, (5) reduces to (27). The situation is similar to that in Example 1, except that $\nabla c_2(x) = [1 \;\; -2x_2]^T$ is not*

constant over x. Attempting to follow a similar analysis as in Example 1, one finds that the arguments break down since $\chi_3 = -\nabla c_1(x)^T \nabla c_2(x)/(\|\nabla c_1(x)\|_2 \|\nabla c_2(x)\|_2) \to 1$ as $x \to 0$ from within the interior of the feasible region. □

Intuitively, Example 2 has the property that the constraint gradients point in opposite directions as $x \to 0$ from within the interior of the feasible region. This means that, if the algorithm were to compute $x \to 0$ (say, since 0 is the optimal solution), the barrier terms for the two constraints would push against each other, in the limit. Consequently, we are not able to show that there exists a uniform ratio μ/θ such that a direction d satisfying (5) always exists over the feasible region.

One sees in Figure 1 our claim that Assumption 3.1 is a type of nondegeneracy assumption. When, at any strictly feasible point, the polar cone of the nearly active constraint gradients is sufficiently *wide*, the situation is favorable, like on the left in Figure 1. On the other hand, if the polar cone collapses as $x \to \bar{x}$ for some $\bar{x}$, like in the situation on the right in Figure 1, then the situation is not favorable since the barrier terms for different constraints push against each other.

We close this section by observing that a procedure for computing a direction to satisfy (5), if one exists, has been suggested in Example 1. In particular, at any iterate x one can compute d by (a) determining the set of nearly active constraints, (b) computing weights for the nearly active constraints, as in (28), such that the contribution from each nearly active constraint gradient is normalized in some sense in order to satisfy (5). Concretely, one can consider the feasibility problem

$$\text{find } \{\chi_i\}_{i \in [m]} \in \mathbb{R}^m_{>0} \text{ such that } d = g - \mu \sum_{i \in [m]} \frac{\chi_i}{c_i(x)} \nabla c_i(x) \text{ satisfies } (5).$$

However, in practice, we suspect such a procedure not to be worth the required computational effort. Instead, we suggest that, in practice, a reasonable choice is to initialize μ and θ to positive values, and in each iteration compute d by solving a linear system of the form (21), as we specify in Assumption 4.3 for the stochastic setting. For example, H_k can simply be an identity matrix, or it might be chosen as an identity matrix plus (an approximation of) the Hessian of the barrier function, which may involve second-order derivatives of $\{c_i\}_{i \in [m]}$, but does not require derivatives (of any order) of the objective function f. In any case, if d fails to satisfy (5) (specifically, (5e)) for some choices of the algorithm parameters, then the algorithm might "reset" the barrier parameter to a larger value. This may be allowed to occur iteratively, at least up to some upper limit for practical purposes. In the worst case, the algorithm may need to compute smaller values of γ than as stated in our theoretical results. However, at least this is a practical approach that worked well in our experiments.

6   Numerical Experiments

As previously stated, we do not claim that the deterministic version of our algorithm would be competitive with state-of-the-art derivative-based interior-point methods, especially not with such methods that allow infeasible iterates and/or employ second-order derivatives. On the other hand, for the stochastic setting, we know of no other stochastic interior-point method like ours that can solve generally constrained, smooth optimization problems, meaning that there is no such method in the literature against which our algorithm can be compared on a level playing field. All of this being said, it is illustrative to demonstrate the practical performance of our method through numerical experiments, both on a well-established and diverse set of test problems and on a couple of relatively straightforward test problems that represent the settings for which the algorithm has been designed. In this section, we present the results of such numerical experiments. For our experiments, each run of the algorithm was conducted on a compute node with an 16 AMD Opteron Processor 6128 with 32GB of memory.

First, we conducted an experiment with a large test set of problems in order to demonstrate the broad applicability of our algorithm. For such a demonstration, in these experiments we considered only the deterministic version of our algorithm. (Our second and third sets of experiments consider the stochastic version.) We generated our test set from the CUTEst collection [12], specifically using the PyCUTEst

interface [11]. This collection includes 1315 unconstrained or constrained smooth optimization problems. To generate the subset of problems that we employed for our experiments, we proceeded as follows. Firstly, we removed all problems with nonlinear equality constraints, with no inequality constraints, or that were too large to be loaded into memory. This resulted in a set of 396 problems of the form (1). Second, since our algorithm requires a strictly feasible initial point, we ran a so-called Phase I algorithm (implemented in Python) starting from the initial point provided by PyCUTEst; see Algorithm 2 in Appendix A. Of the 396 problems on which the Phase I algorithm was run, 10 encountered function evaluation errors through the interface and 59 resulted in a failure to find a strictly feasible point; these were all removed. Our test set is the remainder of these problems—327 in total—for which a strictly feasible initial point could be found. These points were stored as the initial points for this first set of our experiments.

We implemented Algorithm 1 in Python. We refer to the software as `SLIP`. For each test problem, for which x_1 was determined by Algorithm 2, the input parameters for the algorithm were set as follows. The initial neighborhood parameter was set as $\theta_0 \leftarrow -0.9 \max\{c(x_1)\} > 0$ and the initial barrier parameter was set as $\mu_1 \leftarrow \max\{10^{-1}, 2\theta_0\}$. The corresponding sequences of neighborhood and barrier parameters were then prescribed such that $\theta_{k-1} \leftarrow \theta_0 k^{-t}$ and $\mu_k \leftarrow \mu_1 k^{-t}$ with $t \leftarrow 0.7$ for all $k \in \mathbb{N}$. Estimates of $\kappa_{\nabla f}$, $L_{\nabla f}$, $\{\kappa_{c_i}\}_{i\in[m]}$, $\{L_{c_i}\}_{i\in[m]}$, $\{\kappa_{\nabla c_i}\}_{i\in[m]}$, and $\{L_{\nabla c_i}\}_{i\in[m]}$ were determined by randomly generating n points from a normal distribution centered at x_1, then computing constraint function and derivative values at these points to compute bound and Lipschitz-constant estimates. These values were employed to set the step sizes $\{\alpha_k\}$ as stated in Parameter Rule 1 with $t_\alpha \leftarrow 0$. (This ensures $t + t_\alpha \in [-1, 0)$, which means that the condition on these exponents in Theorems 4.1 was satisfied.) Finally, `SLIP` employed the parameter values $\eta \leftarrow (\theta_0/\mu_1 + 1)/2$, $\underline{\eta} \leftarrow \theta_0 + 10^{-8}$, $\underline{\zeta} \leftarrow 1$, $\overline{\zeta} \leftarrow 1$, and $\zeta \leftarrow 1$. Rather than attempt to ensure that (5) holds for all $k \in \mathbb{N}$, `SLIP` follows the strategy described at the end of §5 to reset μ_1 whenever a search direction is computed that does not satisfy (5). In particular, μ_1 was multiplied by a factor of 2 in such cases, although we imposed an overall limit of 10^4 beyond which μ_1 was not increased. One additional feature that we included in the implementation is that we allowed the algorithm to explore values of γ_k greater than 1, since we believe this would be a useful feature in any practical implementation of the method. Specifically, if $x_k + \alpha_k d_k \notin \mathcal{N}(\theta_k)$, then the algorithm would iteratively *reduce* γ_k from 1 until $x_k + \gamma_k \alpha_k d_k \in \mathcal{N}(\theta_k)$ was found. On the other hand, if $x_k + \alpha_k d_k \in \mathcal{N}(\theta_k)$, then the algorithm would iteratively *increase* γ_k from 1 as long as $x_k + \gamma_k \alpha_k d_k \in \mathcal{N}(\theta_k)$ and the barrier term in the barrier-augmented objective function did not increase.

We ran `SLIP` with an iteration budget of $K := 2 \times 10^4$ in order to determine the progress that the algorithm could make within a budget. (To resemble the stochastic setting, our implementation of `SLIP` does not employ second-order derivatives of the objective function, so one should not expect the algorithm to converge at a fast rate like other interior-point methods that employ second-order derivatives.) Let $x_K \in \mathbb{R}^n$ be the final iterate generated by a run of the algorithm on a test problem. Since our algorithm is a feasible method, one can compare $f(x_1)$ and $f(x_K)$ for each run in order to determine if the algorithm has made progress. We confirmed that, indeed, in all of our runs of `SLIP` over our entire test set, we found that $f(x_K) < f(x_1)$. However, without knowledge of $f_{\inf} := \inf_{x \in \mathcal{F}} f(x)$ for each problem in our test set, it is not possible to compare $f(x_K)$ with $f_{\inf}$. Instead, we considered the relative stationarity measure (recall Theorem 4.1):

$$\text{relative stationarity} := \frac{\|P\nabla_x \phi(x_K, \mu_K)\|_2}{\min\{\|P\nabla_x \phi(x_1, \mu_1)\|_2, \|P\nabla_x \phi(x_1, \mu_K)\|_2\}}. \tag{29}$$

Here, the min in the denominator respects the fact that different choices of the barrier parameter correspond to different multiplier estimates at the initial point (see (18)), which in turn give different stationarity measures at the initial point. Our use of the min in the denominator in this definition of relative stationarity means that we are determining the better of the stationarity measures at the initial point to compare against our stationarity measure at the final iterate.

Figure 2 shows a histogram of relative stationarity values over our test set of problems from the CUTEst collection. The results show that, generally speaking, the relative stationarity values were quite small, meaning that the final iterates in most runs appeared to be much closer to stationarity than the initial iterates, thus indicating that within the iteration budget the algorithm made substantial progress toward

stationarity. We conjecture that the problems on which the relative stationarity value was higher may be ones that are very nonlinear and/or have constraints that are degenerate near the iterates generated by the algorithm.
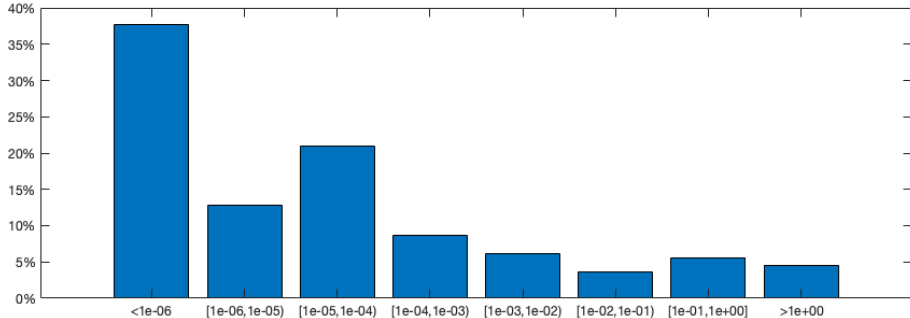

Figure 2: Histogram of relative stationarity values for experiments with a deterministic version of `SLIP` over problems from the CUTEst collection with a strictly feasible initial point.

As a second experiment, we tested stochastic runs of `SLIP` on a randomly generated instance of a second-order conic optimization problem, namely,

$$\min_{x \in \mathbb{R}^n} \ c^T x \quad \text{s.t.} \quad Ax = b \ \text{ and } \ \|x_{1:n-1}\|_2 \le x_n. \tag{30}$$

Specifically, we randomly generated A and b such that a strictly feasible point existed, and randomly generated c such that the problem had a finite optimal value. We chose a problem of this type since it is nonlinear, but convex, so for our first demonstration of stochastic runs of our algorithm we could avoid challenging situations that arise due to nonconvexity, such as the algorithm potentially converging to different minimizers. (By contrast, our third experiment considers a nonconvex problem.) First, as a benchmark, we ran deterministic `SLIP` with the same parameter choices as our experiments with the CUTEst collection, except for two differences: (a) we set $t_\alpha = -0.151$ so that $t + t_\alpha \in [-1, 0)$ and $t + 2t_\alpha < -1$, as required in Parameter Rule 2, and (b) our procedure for computing γ_k for all $k \in \mathbb{N}$ *did not* observe barrier-augmented objective function values, since this would require objective function values that are not tractable to compute in a stochastic setting. Instead, we allowed γ_k to increase above 1, but only up to a limit of 10, when doing so maintained all constraint values less than or equal to $-\theta_k$.

Second, we ran stochastic `SLIP` with 10 different seeds for the random number generator. These runs used the same values for x_1, θ_0, initial μ_1, η, and $\underline{\eta}$ as the deterministic algorithm along with $\gamma_{k,\max} = 10$ for all $k \in \mathbb{N}$. It also employed the same Lipschitz constant estimates and employed the same procedured for "resetting" μ_1 (up to a limit of 10^4) as the deterministic algorithm. Thus, the main difference between the deterministic and stochastic runs of the algorithm were that the latter employed stochastic objective gradient estimates of the form $c + \Delta C$, where for all $k \in \mathbb{N}$ in each run the vector ΔC was drawn randomly from a standard normal distribution. By design of the algorithm, the final iterates generated by the deterministic and all stochastic runs of the algorithm were feasible with respect to all of the constraints. Table 1 shows the final objective values for each of the 10 stochastic runs of `SLIP`. For reference, $f(x_1) = 1.80561\mathrm{e}{+03}$ for all runs, deterministic `SLIP` achieved $f(x_K) = 6.31100\mathrm{e}{+02}$, and the relative stationarity measure achieved by deterministic `SLIP` (see (29)) was $1.65377\mathrm{e}{-02}$. Thus, overall, one can observe that the deterministic and stochastic runs of `SLIP` yielded final iterates that were relatively very close to stationarity.

As a third experiment, we tested stochastic runs of `SLIP` to train a binary classifier subject to a norm constraint for the `mushrooms` data set from the LIBSVM collection [6]. The model that we trained was an artificial neural network with a single hidden layer with 512 nodes, tanh activation, and cross-entropy loss. This objective function is nonconvex. We trained the model subject to the constraint that the neural network weights had squared ℓ_2-norm less than or equal to 100. This feasible region is convex. Like for

Table 1: Final objective function values over 10 runs of stochastic `SLIP` to solve (30).

run	1	2	3	4	5
$f(x_K)$	6.31103e+02	6.31102e+02	6.31100e+02	6.31101e+02	6.31100e+02
run	6	7	8	9	10
$f(x_K)$	6.31101e+02	6.31102e+02	6.31101e+02	6.31101e+02	6.31103e+02

our second experiment, we chose a convex feasible region to ensure that it would be reasonable to expect multiple stochastic runs of the algorithm to converge to a point of similar solution quality in terms of the final objective function values (our measure of comparison), despite the nonconvexity of the objective function. The experimental setup was the same as for our second experiment, except that, rather than artificial noise in the stochastic-gradient estimates, we employed mini-batch gradient estimates with a mini-batch size of 256. (The `mushrooms` dataset has 8124 data points, each with 112 features.)

Table 6 shows the final objective values for each of the 10 stochastic runs of `SLIP`. For reference, $f(x_1) = $ 6.93137e-01 for all runs, deterministic `SLIP` achieved $f(x_K) = $ 7.77697e-03, and the relative stationarity measure achieved by deterministic `SLIP` (see (29)) was 7.06457610e-02. Thus, one can observe that the deterministic and stochastic runs of `SLIP` yielded final iterates of comparable quality.

run	1	2	3	4	5
$f(x_K)$	7.77711e-03	7.76221e-03	7.79686e-03	7.82982e-03	7.78385e-03
run	6	7	8	9	10
$f(x_K)$	7.79379e-03	7.80981e-03	7.80194e-03	7.75375e-03	7.77848e-03

7 Conclusion

We have proposed, analyzed, and tested a single-loop interior-point framework for solving constrained optimization problems. Of particular interest is a stochastic-gradient-based version of the framework, which can be employed for solving problems with affine equality constraints and (potentially nonconvex) nonlinear inequality constraints, at least when a strictly feasible initial point can be provided (or at least computed through a so-called Phase I algorithm, such as the method that we present in Appendix A). We have shown that the algorithm possesses convergence guarantees in both the deterministic and stochastic settings. We have also shown that a deterministic version of the algorithm performs reliably on a large test set of problems, and that a stochastic version of the algorithm yields good results both in the setting of artificial noise and in the context of a mini-batch stochastic-gradient-based algorithm for the training an artificial neural network for binary classification subject to a norm constraint.

A few significant open questions remain in the study of stochastic-gradient-based interior-point methods for solving nonlinearly constrained optimization problems. For example, our theoretical convergence guarantees requires that the ratio μ_1/θ_0 is sufficiently large and that each computed search direction satisfies the conditions in Assumptions 3.1 and/or 4.3, but, as we have explained throughout the paper, a computationally effective strategy for computing search directions that adhere to our theoretical guarantees is not straightforward to design. A related open question is whether it is possible to design a single-loop *infeasible* interior-point framework that possesses theoretical convergence guarantees that are at least on par with those that we offer for the algorithm proposed in this paper. The main challenge in the design of such an approach is that our theoretical guarantees, which employ a prescribed sequence $\{\mu_k\} \searrow 0$, rely heavily on the fact that the iterates generated by our algorithm are feasible in every iteration. There exist stochastic-gradient-based infeasible Newton-type methods for solving equality-constrained optimization problems; see, e.g., [3, 1, 2, 4, 9, 8, 10, 14, 15, 17]. Hence, one might expect it to be possible to employ such methods for solving the equality-constrained subproblems that arise in an infeasible interior-point method. However, even if one were to introduce slack variables for which the barrier functions are introduced, it remains un-

clear how to merge such an approach with our strategy for ensuring feasibility at all iterates. Concretely, suppose that inequality constraints $c(x) \leq 0$ are reformulated with slack variables as $c(x) + s = 0$ along with $s \geq 0$, where the latter inequalities are handled with a barrier function. Attempting to follow the strategy for bound-constrained optimization in [7], one is confronted with the challenge that the search direction in the slack variables, call it d_k^s, depends on the search direction in the original variables, say d_k^x, where it may be possible that $c(x_k) \not\leq 0$. It is difficult to follow a strategy such as ours to ensure, e.g., that there exists γ *uniformly bounded below* such that $s_k + \gamma \alpha_k d_k^s$ is sufficiently positive, say to stay within a prescribed neighborhood. The difficulties of designing such an approach led us to propose the feasible method that has been presented in this paper, although it is possible that such an approach, or one based on alternative strategies, could be designed with convergence guarantees.

A   Appendix: Algorithm for Finding a Strictly Feasible Point

Our algorithm for finding a strictly feasible initial point for our numerical experiments with problems from the CUTEst collection [12] is presented as Algorithm 2 in this appendix. For simplicity, for our discussion here we state user-defined parameters in terms of the specific values that we used in our implementation rather than introduce a generic parameter range.

The algorithm can be understood as an infeasible interior-point method, where line searches on a merit function are performed as a step-acceptance mechanism. Rather than attempt to solve an optimization problem to high accuracy, the aim of the algorithm is merely to find a strictly feasible point, i.e., a point in $\mathcal{F}_{<0}$ (recall (2)) satisfying affine equality constraints and strictly satisfying (potentially nonlinear) inequality constraints. Hence, the barrier parameter is fixed at 1 and the algorithm terminates as soon as a strictly feasible point is found. A strictly feasible point is defined as follows. As opposed to problem (1), which is stated in terms of only one-sided inequality constraints, our definition here of a strictly feasible point distinguishes between one- and two-sided inequalities. With respect to any one-sided inequality, say, $\varphi(x) \leq \varphi_u$, strict feasibility of a point x is defined as $\varphi(x) \leq \varphi_u - 10^{-4}$. With respect to any two-sided inequality, say, $\varphi_l \leq \varphi(x) \leq \varphi_u$, strict feasibility of a point x is defined as

$$\varphi_l + 10^{-4} \min\{\varphi_u - \varphi_l, 1\} \leq \varphi(x) \leq \varphi_u - 10^{-4} \min\{\varphi_u - \varphi_l, 1\}$$

Supposing again that the inequality constraints are stated as in problem (1), the algorithm can be understood as an iterative method toward solving

$$\min_{(x,s) \in \mathbb{R}^n \times \mathbb{R}^m} \quad -\sum_{i \in [m]} \log(s_i) \quad \text{s.t.} \begin{cases} Ax = b \\ c(x) + s = 0 \end{cases} \tag{31}$$

the first-order optimality conditions for which are

$$-S^{-1}e + z = 0, \quad A^T y + \nabla c(x)z = 0, \quad c(x) + s = 0, \quad \text{and} \quad Ax = b.$$

Given an initial point $x_0 \in \mathbb{R}^n$, the algorithm commences by attempting to compute $x_1 \in \mathbb{R}^n$ with $Ax_1 = b$ by solving a least-squares problem. (If this yields $\|Ax_1 - b\|_2 \leq 10^{-6}$, then the algorithm continues; otherwise, the run is a failure.) Then, up to an iteration limit, the algorithm follows a standard infeasible interior-point approach with line searches on a merit function; in particular, the merit function $\phi : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_{>0} \to \mathbb{R}$ is defined by

$$\phi(x, s, \tau) = -\tau \sum_{i=1}^m \log(s_i) + \|c(x) + s\|_1,$$

where $\tau \in \mathbb{R}_{>0}$ is a merit parameter that is updated adaptively. (If the iteration limit is exceeded, then the run is a failure.) Firstly, a Hessian approximation is determined with a modification, if necessary, to ensure that it is sufficiently positive definite. To represent potential use in practice with our proposed Algorithm 1, the Hessian approximation does not employ second-order derivatives of the objective function, although it

does employ second-order derivatives of the constraint functions. Secondly, a search direction is determined by solving a linear system. Thirdly, the merit parameter is updated using a standard technique from the literature; see, e.g., [5]. Specifically, defining the predicted reduction in the merit function corresponding to the tuple (x_k, τ_k) and search direction components (d_k^x, d_k^s) as

$$\Delta q(x_k, s_k, d_k^x, d_k^s, \tau_k) = -\tau_k(-d_k^{s\,T} S_k^{-1} \mathbb{1} + \tfrac{1}{2} d_k^{x\,T} H_k d_k^x + \tfrac{1}{2} d_k^{s\,T} S_k^{-2} d_k^s) + \|c(x_k) + s_k\|_1,$$

the update ensures that τ_k is sufficiently small such that $\Delta q(x_k, s_k, d_k^x, d_k^s, \tau_k)$ is sufficiently large relative to $\|c(x_k) + s_k\|_1$. Fourthly, the largest step size in $(0, 1]$ satisfying a fraction-to-the-boundary rule is determined. Finally, a line search is conducted to compute a step size that satisfies the fraction-to-the-boundary rule and yields sufficient decrease in the merit function.

Algorithm 2 Finding a Strictly Feasible Point

Require: $x_0 \in \mathbb{R}^n$, $\tau_0 = 1$

1: Set $x_1 \leftarrow \operatorname{argmin}_{x \in \mathbb{R}^n} \frac{1}{2}\|x - x_0\|_2^2$ s.t. $Ax = b$
2: Set $s_1 \leftarrow \max\{-c(x_1), 1\}$ and $z_1 \leftarrow s_1^{-1}$ (component-wise)
3: **for** $k \in [10^3]$ **do**
4:     **if** $c(x_k)$ is sufficiently interior, component-wise **then**
5:         terminate and return x_k
6:     **end if**
7:     Set $H_k \leftarrow I + \sum_{i \in [m]} [y_k]_i \nabla^2 c_i(x_k) + \lambda_k I$ for $\lambda_k \geq 0$ such that $H_k \succeq 10^{-4} I$
8:     Solve

$$\begin{bmatrix} H_k & 0 & A^T & \nabla c(x_k) \\ 0 & S_k^{-2} & 0 & I \\ A & 0 & 0 & 0 \\ \nabla c(x_k)^T & I & 0 & 0 \end{bmatrix} \begin{bmatrix} d_k^x \\ d_k^s \\ y_k \\ d_k^z \end{bmatrix} = - \begin{bmatrix} \nabla c(x_k) z_k \\ -S_k^{-1} \mathbb{1} + z_k \\ 0 \\ c(x_k) + s_k \end{bmatrix}$$

9:     Set, with $\Theta_k := -d_k^{s\,T} S_k^{-1} \mathbb{1} + d_k^{x\,T} H_k d_k^x + d_k^{s\,T} S_k^{-2} d_k^s$, first

$$\tau_k^{\text{trial}} \leftarrow \begin{cases} \infty & \text{if } \Theta_k \leq 0 \\ \frac{0.5\|c(x_k) + s_k\|_1}{\Theta_k} & \text{otherwise} \end{cases}$$

$$\text{then } \tau_k \leftarrow \begin{cases} \tau_{k-1} & \text{if } \tau_{k-1} \leq \tau_k^{\text{trial}} \\ (1 - 10^{-6})\tau_k^{\text{trial}} & \text{otherwise} \end{cases}$$

10:     Set $\alpha_k^{\text{ftb}} \leftarrow \min\{\alpha \in (0, 1] : s_k + \alpha d_k^s \geq 0.1 s_k\}$
11:     Set $\alpha_k \leftarrow \alpha^j \alpha_k^{\text{ftb}}$, where j is the minimum value in $\{0\} \cup \mathbb{N}$ such that

$$\phi(x_k + \alpha^j \alpha_k^{\text{ftb}} d_k^x, s_k + \alpha^j \alpha_k^{\text{ftb}} d_k^s, \tau_k) \leq \phi(x_k, s_k, \tau_k) - 10^{-4} \alpha^j \alpha_k^{\text{ftb}} \Delta q(x_k, s_k, d_k^x, d_k^s, \tau_k)$$

12: **end for**

References

[1] Albert S. Berahas, Raghu Bollapragada, and Baoyu Zhou. An adaptive sampling sequential quadratic programming method for equality constrained stochastic optimization. *arXiv 2206.00712*, 2022.

[2] Albert S. Berahas, Frank E. Curtis, Michael J. O'Neill, and Daniel P. Robinson. A Stochastic Sequential Quadratic Optimization Algorithm for Nonlinear Equality Constrained Optimization with Rank-Deficient Jacobians. *Mathematics of Operations Research*, https://doi.org/10.1287/moor.2021.0154, 2023.

[3] Albert S. Berahas, Frank E. Curtis, Daniel P. Robinson, and Baoyu Zhou. Sequential quadratic optimization for nonlinear equality constrained stochastic optimization. *SIAM Journal on Optimization*, 31(2):1352–1379, 2021.

[4] Albert S. Berahas, Jiahao Shi, Zihong Yi, and Baoyu Zhou. Accelerating stochastic sequential quadratic programming for equality constrained optimization using predictive variance reduction. *Computational Optimization and Applications*, 86:79–116, 2022.

[5] Richard H. Byrd, Mary E. Hribar, and Jorge Nocedal. An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4):877–900, 1999.

[6] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[7] Frank E. Curtis, Vyacheslav Kungurtsev, Daniel P. Robinson, and Qi Wang. A Stochastic-Gradient-based Interior-Point Algorithm for Solving Smooth Bound-Constrained Optimization Problems. *arXiv e-prints*, arXiv:2304.14907, April 2023.

[8] Frank E. Curtis, Michael J. O'Neill, and Daniel P. Robinson. Worst-Case Complexity of an SQP Method for Nonlinear Equality Constrained Stochastic Optimization. *Mathematical Programming*, https://doi.org/10.1007/s10107-023-01981-1, 2023.

[9] Frank E. Curtis, Daniel P. Robinson, and Baoyu Zhou. Inexact sequential quadratic optimization for minimizing a stochastic objective function subject to deterministic nonlinear equality constraints. *arXiv 2107.03512 (to appear in INFORMS Journal on Optimization)*, 2021.

[10] Yuchen Fang, Sen Na, Michael W. Mahoney, and Mladen Kolar. Fully stochastic trust-region sequential quadratic programming for equality-constrained optimization problems. *SIAM Journal on Optimization*, 34(2):2007–2037, 2024.

[11] Jaroslav Fowkes, Lindon Roberts, and Árpád Brmen. Pycutest: an open source python package of optimization test problems. *Journal of Open Source Software*, 7(78):4377, 2022.

[12] Nicholas I. M. Gould, Dominique Orban, and Philippe L. Toint. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Comp. Opt. Appl.*, 60(3):545–557, 2015.

[13] Olvi L. Mangasarian and Stan Fromovitz. The Fritz John necessary optimality conditions in the presence of equality and inequality constraints. *Journal of Mathematical Analysis and Applications*, 17:37–47, 1967.

[14] Sen Na, Mihai Anitescu, and Mladen Kolar. An adaptive stochastic sequential quadratic programming with differentiable exact augmented Lagrangians. *Mathematical Programming*, 199:721–791, 2023.

[15] Sen Na and Michael W. Mahoney. Asymptotic convergence rate and statistical inference for stochastic sequential quadratic programming. *arXiv 2205.13687*, 2022.

[16] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer-Verlag New York, 2006.

[17] Songqiang Qiu and Vyacheslav Kungurtsev. A sequential quadratic programming method for optimization with stochastic objective functions, deterministic inequality constraints and robust subproblems. *arXiv 2302.07947*, 2023.

[18] Robert J. Vanderbei and David F. Shanno. An Interior-Point Algorithm for Nonconvex Nonlinear Programming. *Computational Optimization and Applications*, 13:231–252, 1999.

[19] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.