A Stochastic Sequential Quadratic Optimization Algorithm for Nonlinear Equality Constrained Optimization with Rank-Deficient Jacobians

Albert S. Berahas

Department of Industrial and Operations Engineering, University of Michigan

Frank E. Curtis, Michael J. O'Neill, and Daniel P. Robinson

Department of Industrial and Systems Engineering, Lehigh University

COR@L Technical Report 21T-013

A Stochastic Sequential Quadratic Optimization Algorithm for Nonlinear Equality Constrained Optimization with Rank-Deficient Jacobians

Albert S. Berahas[*1], Frank E. Curtis[†2], Michael J. O'Neill[‡2], and Daniel P. Robinson[§2]

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

June 24, 2021

Abstract

A sequential quadratic optimization algorithm is proposed for solving smooth nonlinear equality constrained optimization problems in which the objective function is defined by an expectation of a stochastic function. The algorithmic structure of the proposed method is based on a step decomposition strategy that is known in the literature to be widely effective in practice, wherein each search direction is computed as the sum of a normal step (toward linearized feasibility) and a tangential step (toward objective decrease in the null space of the constraint Jacobian). However, the proposed method is unique from others in the literature in that it both allows the use of stochastic objective gradient estimates and possesses convergence guarantees even in the setting in which the constraint Jacobians may be rank deficient. The results of numerical experiments demonstrate that the algorithm offers superior performance when compared to popular alternatives.

1 Introduction

We propose an algorithm for solving equality constrained optimization problems in which the objective function is defined by an expectation of a stochastic function. Formulations of this type arise throughout science and engineering in important applications such as data-fitting problems, where one aims to determine a model that minimizes the discrepancy between values yielded by the model and corresponding known outputs.

Our algorithm is designed for solving such problems when the decision variables are restricted to the solution set of a (potentially nonlinear) set of equations. We are particularly interested in such problems when the constraint Jacobian—i.e., the matrix of first-order derivatives of the constraint function—may be rank deficient in some or even all iterations during the run of an algorithm, since this can be an unavoidable occurrence in practice that would ruin the convergence properties of any algorithm that is not specifically designed for this setting. The structure of our algorithm follows a step decomposition strategy that is common in the constrained optimization literature; in particular, our algorithm has roots in the Byrd-Omojokun approach [17]. However, our algorithm is unique from previously proposed algorithms in that it

[*]E-mail: albertberahas@gmail.com
[†]E-mail: frank.e.curtis@lehigh.edu
[‡]E-mail: moneill@lehigh.edu
[§]E-mail: daniel.p.robinson@lehigh.edu

offers convergence guarantees while allowing for the use of stochastic objective gradient information in each iteration. We prove that our algorithm converges to stationarity (in expectation), both in desirable cases when the constraints are feasible and convergence to the feasible region can be guaranteed (in expectation), and in less desirable cases, such as when the constraints are infeasible and one can only guarantee convergence to an infeasible stationary point. To the best of our knowledge, there exist no other algorithms in the literature that have been designed specifically for this setting, namely, stochastic optimization with equality constraints that may exhibit rank deficiency.

Our algorithm builds upon the method for solving equality constrained stochastic optimization problems proposed in [1]. The method proposed in that article assumes that the singular values of the constraint Jacobians are bounded below by a positive constant throughout the optimization process, which implies that the linear independence constraint qualification (LICQ) holds at all iterates. By contrast, the algorithm proposed in this paper make no such assumption. Handling the potential lack of full-rank Jacobians necessitates a different algorithmic structure and a distinct approach to proving convergence guarantees; e.g., one needs to account for the fact that primal-dual stationarity conditions may not be necessary and/or the constraints may be infeasible.

Similar to the context in [1], our algorithm is intended for the highly stochastic regime in which the stochastic gradient estimates might only be unbiased estimators of the gradients of the objective at the algorithm iterates that satisfy a loose variance condition. Indeed, we show that in *nice* cases—in particular, when the adaptive merit parameter employed in our algorithm eventually settles at a value that is sufficiently small—our algorithm has convergence properties in expectation that match those of the algorithm in [1]. These results parallel those for the stochastic gradient method in the context of unconstrained optimization [2, 21, 22]. However, for cases not considered in [1] when the merit parameter sequence may vanish, we require the stronger assumption that the difference between each stochastic gradient estimate and the corresponding true gradient of the objective eventually is bounded *deterministically* in each iteration. This is appropriate in many ways since in such a scenario the algorithm aims to transition from solving a *stochastic* optimization problem to the *deterministic* one of minimizing constraint violation. Finally, we discuss how in any particular run of the algorithm, the probability is zero that the merit parameter settles at too large of a value, and provide commentary on what it means to assume that the total probability of such an event (over all possible runs of the algorithm) is zero.

Our algorithm has some similarities, but many differences with another recently proposed algorithm, namely, that in [14]. That algorithm is also designed for equality constrained stochastic optimization, but: (*i*) like for the algorithm in [1], for the algorithm in [14] the LICQ is assumed to hold at all algorithm iterates, and (*ii*) the algorithm in [14] employs an adaptive line search that may require the algorithm to compute relatively accurate stochastic gradient estimates throughout the optimization process. Our algorithm, on the other hand, does not require the LICQ to hold and is meant for a more stochastic regime, meaning that it does not require a procedure for refining the stochastic gradient estimate within an iteration. Consequently, the convergence guarantees that can be proved for our method, and the expectations that one should have about the practical performance of our method, are quite distinct from those for the algorithm in [14].

Besides the methods in [1, 14], there have been few proposed algorithms that might be used to solve problem of the form (1). Some methods have been proposed that employ stochastic (proximal) gradient strategies applied to minimizing penalty functions derived from constrained problems [4, 11, 15], but these do not offer convergence guarantees to stationarity with respect to the original constrained problem. On the other hand, stochastic Frank-Wolfe methods have been proposed [10, 12, 13, 19, 20, 24], but these can only be applied in the context of convex feasible regions. Our algorithm, by contrast, is designed for nonlinear equality constrained stochastic optimization.

1.1 Notation

The set of real numbers is denoted as $\mathbb{R}$, the set of real numbers greater than (respectively, greater than or equal to) $r \in \mathbb{R}$ is denoted as $\mathbb{R}_{>r}$ (respectively, $\mathbb{R}_{\geq r}$), the set of n-dimensional real vectors is denoted as $\mathbb{R}^n$, the set of m-by-n-dimensional real matrices is denoted as $\mathbb{R}^{m \times n}$, and the set of n-by-n-dimensional real symmetric matrices is denoted as $\mathbb{S}^n$. Given $J \in \mathbb{R}^{m\times n}$, the range space of J^T is denoted as $\text{Range}(J^T)$ and the null space of J is denoted as $\text{Null}(J)$. (By the Fundamental Theorem of Linear Algebra, for any $J \in \mathbb{R}^{m\times n}$, the spaces $\text{Range}(J^T)$ and $\text{Null}(J)$ are orthogonal and $\text{Range}(J^T) + \text{Null}(J) = \mathbb{R}^n$.) The set of nonnegative integers is denoted as $\mathbb{N} := \{0, 1, 2, \dots\}$. For any $m \in \mathbb{N}$, let $[m]$ denote the set of integers $\{0, 1, \dots, m\}$.

The algorithm that we propose is iterative in the sense that, given a starting point $x_0 \in \mathbb{R}^n$, it generates a sequence of iterates $\{x_k\}$ with $x_k \in \mathbb{R}^n$ for all $k \in \mathbb{N}$. For simplicity of notation, the iteration number is appended as a subscript to other quantities corresponding to each iteration; e.g., with a function $c : \mathbb{R}^n \to \mathbb{R}$, its value at x_k is denoted as $c_k := c(x_k)$ for all $k \in \mathbb{N}$. Given $J_k \in \mathbb{R}^{m\times n}$, we use Z_k to denote a matrix whose columns form an orthonormal basis for $\text{Null}(J_k)$.

1.2 Organization

Our problem of interest and basic assumptions about the problem and the behavior of our algorithm are presented in Section 2. Our algorithm is motivated and presented in Section 3. Convergence guarantees for our algorithm are presented in Section 4. The results of numerical experiments are provided in Section 5 and concluding remarks are provided in Section 6.

2 Problem Statement

Our algorithm is designed for solving (potentially nonlinear and/or nonconvex) equality constrained optimization problems of the form

$$\min_{x\in\mathbb{R}^n} \ f(x) \quad \text{s.t.} \quad c(x) = 0, \quad \text{with} \quad f(x) = \mathbb{E}[F(x, \iota)], \tag{1}$$

where the functions $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^m$ are smooth, ι is a random variable with associated probability space $(\Omega, \mathcal{F}, P)$, $F : \mathbb{R}^n \times \Omega \to \mathbb{R}$, and $\mathbb{E}[\cdot]$ denotes expectation taken with respect to P. We assume that values and first-order derivatives of the constraint functions can be computed, but that the objective and its associated first-order derivatives are intractable to compute, and one must instead employ stochastic estimates. (We formalize our assumptions about such stochastic estimates starting with Assumption 2 on page 6.) Formally, we make the following assumption with respect to (1) and our proposed algorithm, which generates a sequence of iterates $\{x_k\}$.

Assumption 1. *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be an open convex set containing the sequence $\{x_k\}$ generated by any run of the algorithm. The objective function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and bounded over $\mathcal{X}$ and its gradient function $\nabla f : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous with constant $L \in \mathbb{R}_{>0}$ (with respect to $\|\cdot\|_2$) and bounded over $\mathcal{X}$. The constraint function $c : \mathbb{R}^n \to \mathbb{R}^m$ (with $m \le n$) is continuously differentiable and bounded over $\mathcal{X}$ and its Jacobian function $J := \nabla c^T : \mathbb{R}^n \to \mathbb{R}^{m\times n}$ is Lipschitz continuous with constant $\Gamma \in \mathbb{R}_{>0}$ (with respect to $\|\cdot\|_2$) and bounded over $\mathcal{X}$.*

The aspects of Assumption 1 that pertain to the objective function f and constraint function c are typical for the equality constrained optimization literature. Notice that we do not assume that the iterate sequence itself is bounded. Under Assumption 1, it follows that there exist positive real numbers $(f_{\inf}, f_{\sup}, \kappa_{\nabla f}, \kappa_c, \kappa_J) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ such that

$$f_{\inf} \le f_k \le f_{\sup}, \quad \|\nabla f(x_k)\|_2 \le \kappa_{\nabla f}, \quad \|c_k\|_2 \le \kappa_c, \quad \text{and} \quad \|J_k\|_2 \le \kappa_J \quad \text{for all} \quad k \in \mathbb{N}. \tag{2}$$

Given that our proposed algorithm is stochastic, it is admittedly not ideal to have to assume that the objective value, objective gradient, constraint value, and constraint Jacobian are bounded over the set $\mathcal{X}$ containing the iterates. This is a common assumption in the deterministic optimization literature, where it may be justified in the context of an algorithm that is guaranteed to make progress in each iteration, say with respect to a merit function. However, for a stochastic algorithm such as ours, such a claim may be seen as less than ideal since a stochastic algorithm may only be guaranteed to make progress *in expectation* in each iteration, meaning that it is possible for the iterates to drift far from desirable regions of the search space during the optimization process.

Our justification for Assumption 1 is two-fold. First, any reader who is familiar with analyses of stochastic algorithms for unconstrained optimization—in particular, those analyses that do not require that the objective gradient is bounded over a set containing the iterates—should appreciate that additional challenges present themselves in the context of constrained optimization. For example, whereas in unconstrained optimization one naturally considers the objective f as a measure of progress, in (nonconvex) constrained optimization one needs to employ a merit function for measuring progress, and for practical purposes such a function typically needs to involve a parameter (or parameters) that must be adjusted dynamically by the algorithm. One finds that it is the adaptivity of our merit parameter (see (10) later on) that necessitates the aforementioned boundedness assumptions that we use in our analysis. (Certain exact merit functions, such as that employed in [14], might not lead to the same issues as the merit function that we employ. However, we remark that the merit function employed in [14] is not a viable option unless the LICQ holds at all algorithm iterates.) Our second justification is that we know of no other algorithm that offers convergence guarantees that are as comprehensive as ours (in terms of handling feasible, degenerate, and infeasible settings) under an assumption that is at least as loose as Assumption 1.

Let the Lagrangian $\ell : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ corresponding to (1) be given by $\ell(x, y) = f(x) + c(x)^T y$, where $y \in \mathbb{R}^m$ represents a vector of Lagrange multipliers. Under a constraint qualification (such as the LICQ), necessary conditions for first-order stationarity with respect to (1) are given by

$$0 = \begin{bmatrix} \nabla_x \ell(x, y) \\ \nabla_y \ell(x, y) \end{bmatrix} = \begin{bmatrix} \nabla f(x) + J(x)^T y \\ c(x) \end{bmatrix}; \tag{3}$$

see, e.g., [16]. However, under only Assumption 1, it is possible for (1) to be degenerate—in which case (3) might not be necessary at a solution of (1)—or (1) may be infeasible. In the latter case, one aims to design an algorithm that transitions automatically from seeking stationarity with respect to (1) to seeking stationarity with respect to a measure of infeasibility of the constraints. For our purposes, we employ the infeasibility measure $\varphi : \mathbb{R}^n \to \mathbb{R}$ defined by $\varphi(x) = \|c(x)\|_2$. A point $x \in \mathbb{R}^n$ is stationary with respect to φ if and only if either $c(x) = 0$ or both $c(x) \neq 0$ and

$$0 = \nabla \varphi(x) = \frac{J(x)^T c(x)}{\|c(x)\|_2}. \tag{4}$$

3 Algorithm Description

Our algorithm can be characterized as a sequential quadratic optimization (commonly known as SQP) method that employs a step decomposition strategy and chooses stepsizes that attempt to ensure sufficient decrease in a merit function in each iteration. We present our complete algorithm in this section, which builds upon this basic characterization to involve various unique aspects that are designed for handling the combination of (*i*) stochastic gradient estimates and (*ii*) potential rank deficiency of the constraint Jacobians.

In each iteration $k \in \mathbb{N}$, the algorithm first computes the *normal component* of the search direction toward reducing linearized constraint violation. Conditioned on the event that x_k is reached as the kth iterate, the problem defining this computation, namely,

$$\min_{v \in \mathbb{R}^n} \ \tfrac{1}{2}\|c_k + J_k v\|_2^2 \ \ \text{s.t.} \ \ \|v\|_2 \leq \omega \|J_k^T c_k\|_2 \tag{5}$$

where $\omega \in \mathbb{R}_{>0}$ is a user-defined parameter, is deterministic since the constraint function value c_k and constraint Jacobian J_k are available. An exact solution of (5) may be expensive to obtain. Fortunately, however, our algorithm merely requires that the normal component $v_k \in \mathbb{R}^n$ is feasible for problem (5), lies in Range(J_k^T), and satisfies the Cauchy decrease condition

$$\|c_k\|_2 - \|c_k + J_k v_k\|_2 \geq \epsilon_v (\|c_k\|_2 - \|c_k + \alpha_k^C J_k v_k^C\|_2) \tag{6}$$

for some user-defined parameter $\epsilon_v \in (0,1]$. Here, $v_k^C := -J_k^T c_k$ is the steepest descent direction for the objective of problem (5) at $v = 0$ and the stepsize $\alpha_k^C \in \mathbb{R}$ is the unique solution to the problem to minimize $\frac{1}{2}\|c_k + \alpha^C J_k v_k^C\|_2^2$ over $\alpha^C \in \mathbb{R}_{\geq 0}$ subject to $\alpha^C \leq \omega$. Since this allows one to choose $v_k \leftarrow v_k^C$, the normal component can be computed at low computational cost. For a more accurate solution to (5), one can employ a so-called matrix-free iterative algorithm such as the linear conjugate gradient (CG) method with Steihaug stopping conditions [23] or GLTR [6], each of which is guaranteed to yield a solution satisfying the aforementioned conditions no matter how many iterations (greater than or equal to one) are performed.

After the computation of the normal component, our algorithm computes the *tangential component* of the search direction by minimizing a model of the objective function subject to remaining in the null space of the constraint Jacobian. This ensures that the progress toward linearized feasibility offered by the normal component is not undone by the tangential component when the components are added together. The problem defining the computation of the tangential component is

$$\min_{u \in \mathbb{R}^n} \ (g_k + H_k v_k)^T u + \tfrac{1}{2} u^T H_k u \quad \text{s.t.} \quad J_k u = 0, \tag{7}$$

where $g_k \in \mathbb{R}^n$ is a stochastic gradient estimate *at least* satisfying Assumption 2 below and the real symmetric matrix $H_k \in \mathbb{S}^n$ satisfies Assumption 3 below. (Specific additional requirements for $\{g_k\}$ are stated separately for each case in our convergence analysis.)

Assumption 2. *For all $k \in \mathbb{N}$, the stochastic gradient estimate $g_k \in \mathbb{R}^n$ is an unbiased estimator of $\nabla f(x_k)$, i.e., $\mathbb{E}_k[g_k] = \nabla f(x_k)$, where $\mathbb{E}_k[\cdot]$ denotes expectation conditioned on the event that the algorithm has reached x_k as the kth iterate. In addition, there exists a positive real number $M \in \mathbb{R}_{>0}$ such that, for all $k \in \mathbb{N}$, one has $\mathbb{E}_k[\|g_k - \nabla f(x_k)\|_2^2] \leq M$.*

Assumption 3. *The matrix $H_k \in \mathbb{S}^n$ is chosen independently from g_k for all $k \in \mathbb{N}$, the sequence $\{H_k\}$ is bounded in norm by $\kappa_H \in \mathbb{R}_{>0}$, and there exists $\zeta \in \mathbb{R}_{>0}$ such that, for all $k \in \mathbb{N}$, one has $u^T H_k u \geq \zeta \|u\|_2^2$ for all $u \in \mathrm{Null}(J_k)$.*

In our context, one can generate g_k in iteration $k \in \mathbb{N}$ by independently drawing b_k realizations of the random variable ι, denoting the *mini-batch* as $\mathcal{B}_k := \{\iota_{k,1}, \dots, \iota_{k,b_k}\}$, and setting

$$g_k \leftarrow \frac{1}{b_k} \sum_{\iota \in \mathcal{B}_k} \nabla f(x_k, \iota). \tag{8}$$

It is a modest assumption about the function f and the sample sizes $\{b_k\}$ to say that $\{g_k\}$ generated in this manner satisfies Assumption 2. As for Assumption 3, the assumptions that the elements of $\{H_k\}$ are bounded in norm and that H_k is sufficiently positive definite in $\mathrm{Null}(J_k)$ for all $k \in \mathbb{N}$ are typical for the constrained optimization literature. In practice, one may choose H_k to be (an approximation of) the Hessian of the Lagrangian at (x_k, y_k) for some y_k, if such a matrix can be computed with reasonable effort in a manner that guarantees that Assumption 3 holds. A simpler alternative is that H_k can be set to some positive definite diagonal matrix (independent of g_k).

Under Assumption 3, the tangential component u_k solving (7) can be obtained by solving

$$\begin{bmatrix} H_k & J_k^T \\ J_k & 0 \end{bmatrix} \begin{bmatrix} u_k \\ y_k \end{bmatrix} = -\begin{bmatrix} g_k + H_k v_k \\ 0 \end{bmatrix}. \tag{9}$$

Even if the constraint Jacobian J_k does not have full row rank, the linear system (9) is consistent since it represents sufficient optimality conditions (under Assumption 3) of the linearly constrained quadratic optimization problem in (7). Under Assumption 3, the solution component u_k is unique, although the component y_k might not be unique (if J_k does not have full row rank).

Upon computation of the search direction, our algorithm proceeds to determining a positive stepsize. For this purpose, we employ the merit function $\phi : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ defined by

$$\phi(x, \tau) = \tau f(x) + \|c(x)\|_2 \tag{10}$$

where τ is a merit parameter whose value is set dynamically. The function ϕ is a type of exact penalty function that is common in the literature [8, 9, 18]. For setting the merit parameter value in each iteration, we employ a local model of ϕ denoted as $l : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and defined by

$$l(x, \tau, g, d) = \tau(f(x) + g^T d) + \|c(x) + J(x)d\|_2.$$

Given the search direction vectors v_k, u_k, and $d_k \leftarrow v_k + u_k$, the algorithm sets

$$\tau_k^{\text{trial}} \leftarrow \begin{cases} \infty & \text{if } g_k^T d_k + u_k^T H_k u_k \leq 0 \\ \dfrac{(1-\sigma)(\|c_k\|_2 - \|c_k + J_k d_k\|_2)}{g_k^T d_k + u_k^T H_k u_k} & \text{otherwise,} \end{cases} \tag{11}$$

where $\sigma \in (0, 1)$ is user-defined. The merit parameter value is then set as

$$\tau_k \leftarrow \begin{cases} \tau_{k-1} & \text{if } \tau_{k-1} \leq \tau_k^{\text{trial}} \\ \min\{(1-\epsilon_\tau)\tau_{k-1}, \tau_k^{\text{trial}}\} & \text{otherwise,} \end{cases} \tag{12}$$

where $\epsilon_\tau \in (0, 1)$ is user-defined. This rule ensures that $\{\tau_k\}$ is monotonically nonincreasing, $\tau_k \leq \tau_k^{\text{trial}}$ for all $k \in \mathbb{N}$, and, with the reduction function $\Delta l : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ defined by

$$\Delta l(x, \tau, g, d) = l(x, \tau, g, 0) - l(x, \tau, g, d) = -\tau g^T d + \|c(x)\|_2 - \|c(x) + J(x)d\|_2 \tag{13}$$

and Assumption 3, it ensures the following fact that is critical for our analysis:

$$\Delta l(x_k, \tau_k, g_k, d_k) \geq \tfrac{1}{2}\tau_k u_k^T H_k u_k + \sigma(\|c_k\|_2 - \|c_k + J_k v_k\|_2). \tag{14}$$

Similar to the algorithm in [1], our algorithm also adaptively sets other parameters that are used for determining an allowable range for the stepsize in each iteration. For distinguishing between search directions that are dominated by the tangential component and others that are dominated by the normal component, the algorithm adaptively defines sequences $\{\chi_k\}$ and $\{\zeta_k\}$. (These sequences were not present in the algorithm in [1]; they are newly introduced for the needs of our proposed algorithm.) In particular, in iteration $k \in \mathbb{N}$, the algorithm employs the conditions

$$\|u_k\|_2^2 \geq \chi_{k-1}\|v_k\|_2^2 \quad \text{and} \quad \tfrac{1}{2}d_k^T H_k d_k < \tfrac{1}{4}\zeta_{k-1}\|u_k\|_2^2 \tag{15}$$

in order to set

$$(\chi_k, \zeta_k) \leftarrow \begin{cases} ((1+\epsilon_\chi)\chi_{k-1}, (1-\epsilon_\zeta)\zeta_{k-1}) & \text{if (15) holds} \\ (\chi_{k-1}, \zeta_{k-1}) & \text{otherwise,} \end{cases} \tag{16}$$

where $\epsilon_\chi \in \mathbb{R}_{>0}$ and $\epsilon_\zeta \in (0, 1)$ are user-defined. It follows from (16) that $\{\chi_k\}$ is monotonically nondecreasing and $\{\zeta_k\}$ is monotonically nonincreasing. It will be shown in our analysis that $\{\chi_k\}$ is bounded above by a positive real number and $\{\zeta_k\}$ is bounded below by a positive real number, where these bounds are uniform over all runs of the algorithm; i.e., these sequences are bounded *deterministically.* This means that despite the stochasticity of the algorithm iterates, these sequences have $(\chi_k, \zeta_k) = (\chi_{k-1}, \zeta_{k-1})$ for all sufficiently large $k \in \mathbb{N}$ in any run of the algorithm.

Whether $\|u_k\|_2^2 \geq \chi_k\|v_k\|_2^2$ (i.e., the search direction is *tangentially dominated*) or $\|u_k\|_2^2 < \chi_k\|v_k\|_2^2$ (i.e., the search direction is *normally dominated*) influences two aspects of iteration $k \in \mathbb{N}$. First, it influences a value that the algorithm computes to estimate a lower bound for the ratio between the reduction in the model l of the merit function and a quantity involving the squared norm of the search direction. (A similar, but slightly different sequence was employed for the algorithm in [1].) In iteration $k \in \mathbb{N}$ of our algorithm, the estimated lower bound is set adaptively by first setting

$$\xi_k^{\text{trial}} \leftarrow \begin{cases} \dfrac{\Delta l(x_k, \tau_k, g_k, d_k)}{\tau_k\|d_k\|_2^2} & \text{if } \|u_k\|_2^2 \geq \chi_k\|v_k\|_2^2 \\ \dfrac{\Delta l(x_k, \tau_k, g_k, d_k)}{\|d_k\|_2^2} & \text{otherwise,} \end{cases} \tag{17}$$

then setting

$$\xi_k \leftarrow \begin{cases} \xi_{k-1} & \text{if } \xi_{k-1} \leq \xi_k^{\text{trial}} \\ \min\{(1-\epsilon_\xi)\xi_{k-1}, \xi_k^{\text{trial}}\} & \text{otherwise,} \end{cases} \tag{18}$$

for some user-defined $\epsilon_\xi \in (0,1)$. The procedure in (18) ensures that $\{\xi_k\}$ is monotonically nonincreasing and $\xi_k \leq \xi_k^{\text{trial}}$ for all $k \in \mathbb{N}$. It will be shown in our analysis that $\{\xi_k\}$ is bounded away from zero *deterministically*, even though in each iteration it depends on stochastic quantities. To achieve this property, it is critical that the denominator in (17) is different depending on whether the search direction is tangentially or normally dominated; see Lemma 3 later on for details. The second aspect of the algorithm that is affected by whether a search direction is tangentially or normally dominated is a rule for setting the stepsize; this will be seen in (22) later on.

We are now prepared to present the mechanism by which a positive stepsize is selected in each iteration $k \in \mathbb{N}$ of our algorithm. We present a strategy that allows for our convergence analysis in Section 4 to be as straightforward as possible. In Section 5, we remark on extensions of this strategy that are included in our software implementation for which our convergence guarantees also hold (as long as some additional cases are considered in one key lemma).

We motivate our strategy by considering an upper bound for the change in the merit function corresponding to the computed search direction, namely, $d_k \leftarrow v_k + u_k$. In particular, under Assumption 1, in iteration $k \in \mathbb{N}$, one has for any nonnegative stepsize $\alpha \in \mathbb{R}_{\geq 0}$ that

$$\begin{aligned} &\phi(x_k + \alpha d_k, \tau_k) - \phi(x_k, \tau_k) \\ &= \tau_k f(x_k + \alpha d_k) - \tau_k f(x_k) + \|c(x_k + \alpha d_k)\|_2 - \|c_k\|_2 \\ &\leq \alpha \tau_k \nabla f(x_k)^T d_k + \|c_k + \alpha J_k d_k\|_2 - \|c_k\|_2 + \tfrac{1}{2}(\tau_k L + \Gamma)\alpha^2 \|d_k\|_2^2 \\ &\leq \alpha \tau_k \nabla f(x_k)^T d_k + |1-\alpha| \|c_k\|_2 - \|c_k\|_2 + \alpha \|c_k + J_k d_k\|_2 + \tfrac{1}{2}(\tau_k L + \Gamma)\alpha^2 \|d_k\|_2^2. \end{aligned} \tag{19}$$

This upper bound is a convex, piecewise quadratic function in α. In a deterministic algorithm in which the gradient $\nabla f(x_k)$ is available, it is common to require that the stepsize α yields

$$\phi(x_k + \alpha d_k, \tau_k) - \phi(x_k, \tau_k) \leq -\eta \alpha \Delta l(x_k, \tau_k, \nabla f(x_k), d_k), \tag{20}$$

where $\eta \in (0,1)$ is user-defined. However, in our setting, (20) cannot be enforced since our algorithm avoids the evaluation of $\nabla f(x_k)$ and in lieu of it only computes a stochastic gradient g_k. The first main idea of our stepsize strategy is to determine a stepsize such that the upper bound in (19) is less than or equal to the right-hand side of (20) when the true gradient $\nabla f(x_k)$ is replaced by its estimate g_k. Since (14), the orthogonality of $v_k \in \text{Range}(J_k^T)$ and $u_k \in \text{Null}(J_k)$, and the properties of the normal step (which, as shown in Lemma 1 later on, include that the left-hand side of (6) is positive whenever $v_k \neq 0$) ensure that $\Delta l(x_k, \tau_k, g_k, d_k) > 0$ whenever $d_k \neq 0$, it follows that a stepsize satisfying this aforementioned property is given, for any $\beta_k \in (0,1]$, by

$$\alpha_k^{\text{suff}} \leftarrow \min\left\{ \frac{2(1-\eta)\beta_k \Delta l(x_k, \tau_k, g_k, d_k)}{(\tau_k L + \Gamma)\|d_k\|_2^2}, 1 \right\} \in \mathbb{R}_{>0}. \tag{21}$$

The sequence $\{\beta_k\}$ referenced in (21) is chosen with different properties—namely, constant or diminishing—depending on the desired type of convergence guarantee. We discuss details of the possible choices for $\{\beta_k\}$ and the consequences of these choices along with our convergence analysis.

Given that the stepsize α_k^{suff} in (21) has been set based on a stochastic gradient estimate, a safeguard is needed for our convergence guarantees. For this purpose, the second main idea of our stepsize selection strategy is to project the trial stepsize onto an interval that is appropriate depending on whether the search direction is tangentially dominated or normally dominated. In particular, the stepsize is chosen as

$\alpha_k \leftarrow \text{Proj}_k(\alpha_k^{\text{suff}})$ where

$$\text{Proj}_k(\cdot) := \begin{cases} \text{Proj}\left(\cdot \,\middle|\, \left[\frac{2(1-\eta)\beta_k\xi_k\tau_k}{\tau_k L + \Gamma}, \frac{2(1-\eta)\beta_k\xi_k\tau_k}{\tau_k L + \Gamma} + \theta\beta_k^2\right]\right) & \text{if } \|u_k\|_2^2 \geq \chi_k\|v_k\|_2^2 \\ \text{Proj}\left(\cdot \,\middle|\, \left[\frac{2(1-\eta)\beta_k\xi_k}{\tau_k L + \Gamma}, \frac{2(1-\eta)\beta_k\xi_k}{\tau_k L + \Gamma} + \theta\beta_k^2\right]\right) & \text{otherwise.} \end{cases} \tag{22}$$

Here, $\text{Proj}(\cdot|\mathcal{I})$ denotes the projection onto the interval $\mathcal{I} \subset \mathbb{R}$. Motivation for the difference in the interval depending on whether the search direction is tangentially or normally dominated can be seen Lemma 15 later on, where it is critical that the stepsize for a normally dominated search direction does not necessarily vanish if/when the merit parameter vanishes, i.e., $\{\tau_k\} \searrow 0$.

Overall, our stepsize selection mechanism can be understood as follows. First, the algorithm adaptively sets the sequences $\{\chi_k\}$, $\{\zeta_k\}$, and $\{\xi_k\}$ in order to estimate bounds that are needed for the stepsize selection and are known to exist theoretically, but cannot be computed directly. By the manner in which these sequences are set, our analysis shows that they remain constant for sufficiently large $k \in \mathbb{N}$ in any run of the algorithm. With these values, our stepsize selection strategy aims to achieve a reduction in the merit function in expectation, with safeguards since the computed values are based on stochastic quantities. One finds by the definition of the projection interval in (22) that the stepsize *for a tangentially dominated search direction* may decrease to zero if $\{\tau_k\} \searrow 0$; this is needed in cases when the problem is degenerate or infeasible, and the algorithm wants to avoid long steps in the tangential component that may ruin progress toward minimizing constraint violation. Otherwise, *for a normally dominated search direction*, the stepsize would remain bounded away from zero if $\beta_k = \beta \in (0,1]$ for all $k \in \mathbb{N}$; i.e., it can only decrease to zero if $\{\beta_k\}$ is diminishing. If our algorithm did not make this distinction between the projection intervals for tangentially versus normally dominated search directions, then the algorithm would fail to have desirable convergence guarantees even in the deterministic setting. (In particular, our proof in Appendix A of Theorem 1, which is upcoming in Section 4, would break down.)

Our complete algorithm is stated as Algorithm 1 on page 10.

4 Convergence Analysis

In this section, we prove convergence guarantees for Algorithm 1. To understand the results that can be expected given our setting and the type of algorithm that we employ, let us first present a set of guarantees that can be proved if Algorithm 1 were to be run with $g_k = \nabla f(x_k)$ and $\beta_k = \beta$ for all $k \in \mathbb{N}$, where $\beta \in \mathbb{R}_{>0}$ is sufficiently small. For such an algorithm, we prove the following theorem in Appendix A. The theorem is consistent with what can be proved for other deterministic algorithms in our context; e.g., see Theorem 3.3 in [5].

Theorem 1. *Suppose Algorithm 1 is employed to solve problem* (1) *such that Assumption 1 holds,* $g_k = \nabla f(x_k)$ *for all* $k \in \mathbb{N}$, $\{H_k\}$ *satisfies Assumption 3, and* $\beta_k = \beta$ *for all* $k \in \mathbb{N}$ *where*

$$\beta \in (0,1] \quad \textit{and} \quad \frac{2(1-\eta)\beta\xi_{-1}\max\{\tau_{-1},1\}}{\Gamma} \in (0,1]. \tag{23}$$

If there exist $k_J \in \mathbb{N}$ *and* $\sigma_J \in \mathbb{R}_{>0}$ *such that the singular values of* J_k *are bounded below by* σ_J *for all* $k \geq k_J$*, then the merit parameter sequence* $\{\tau_k\}$ *is bounded below by a positive real number and*

$$0 = \lim_{k\to\infty} \left\|\begin{bmatrix} \nabla f(x_k) + J_k^T y_k \\ c_k \end{bmatrix}\right\|_2 = \lim_{k\to\infty} \left\|\begin{bmatrix} Z_k^T \nabla f(x_k) \\ c_k \end{bmatrix}\right\|_2. \tag{24}$$

Otherwise, if such k_J *and* σ_J *do not exist, then it still follows that*

$$0 = \lim_{k\to\infty} \|J_k^T c_k\|_2, \tag{25}$$

Algorithm 1 Stochastic SQP Algorithm

```
Require: $L \in \mathbb{R}_{>0}$, a Lipschitz constant for $\nabla f$; $\Gamma \in \mathbb{R}_{>0}$, a Lipschitz constant for $c$; $\{\beta_k\} \subset (0,1]$; $x_0 \in \mathbb{R}^n$; $\tau_{-1} \in \mathbb{R}_{>0}$; $\chi_{-1} \in \mathbb{R}_{>0}$; $\zeta_{-1} \in \mathbb{R}_{>0}$; $\omega \in \mathbb{R}_{>0}$; $\epsilon_v \in (0,1]$; $\sigma \in (0,1)$; $\epsilon_\tau \in (0,1)$; $\epsilon_\chi \in \mathbb{R}_{>0}$; $\epsilon_\zeta \in (0,1)$; $\epsilon_\xi \in (0,1)$; $\eta \in (0,1)$; $\theta \in \mathbb{R}_{\geq 0}$
for $k \in \mathbb{N}$ do
    if $\|J_k^T c_k\|_2 = 0$ and $\|c_k\|_2 > 0$ then
        terminate and return $x_k$ (infeasible stationary point)
    end if
    Compute a stochastic gradient $g_k$ at least satisfying Assumption 2
    Compute $v_k \in \mathrm{Range}(J_k^T)$ that is feasible for problem (5) and satisfies (6)
    Compute $(u_k, y_k)$ as a solution of (9), and then set $d_k \leftarrow v_k + u_k$
    if $d_k = 0$ then
        Set $\tau_k^{\mathrm{trial}} \leftarrow \infty$ and $\tau_k \leftarrow \tau_{k-1}$
        Set $(\chi_k, \zeta_k) \leftarrow (\chi_{k-1}, \zeta_{k-1})$
        Set $\xi_k^{\mathrm{trial}} \leftarrow \infty$ and $\xi_k \leftarrow \xi_{k-1}$
        Set $\alpha_k^{\mathrm{suff}} \leftarrow 1$ and $\alpha_k \leftarrow 1$
    else
        Set $\tau_k^{\mathrm{trial}}$ by (11) and $\tau_k$ by (12)
        Set $(\chi_k, \zeta_k)$ by (15)–(16)
        Set $\xi_k^{\mathrm{trial}}$ by (17) and $\xi_k$ by (18)
        Set $\alpha_k^{\mathrm{suff}}$ by (21) and $\alpha_k \leftarrow \mathrm{Proj}_k(\alpha_k^{\mathrm{suff}})$ using (22)
    end if
    Set $x_{k+1} \leftarrow x_k + \alpha_k d_k$
end for
```

and if $\{\tau_k\}$ is bounded below by a positive real number, then

$$0 = \lim_{k\to\infty} \|\nabla f(x_k) + J_k^T y_k\|_2 = \lim_{k\to\infty} \|Z_k^T \nabla f(x_k)\|_2. \tag{26}$$

Based on Theorem 1, the following aims—which are all achieved in certain forms in our analyses in Sections 4.1 and 4.2—can be set for Algorithm 1 in the stochastic setting. First, if Algorithm 1 is run and the singular values of the constraint Jacobians happen to remain bounded away from zero beyond some iteration, then (following (24)) one should aim to prove that a primal-dual stationarity measure (recall (3)) vanishes in expectation. This is shown under certain conditions in Corollary 1 (and the subsequent discussion) on page 17. Otherwise, a (sub)sequence of $\{J_k\}$ tends to singularity, in which case (following (25)) one should at least aim to prove that $\{\|J_k^T c_k\|_2\}$ vanishes in expectation, which would mean that a (sub)sequence of iterates converges in expectation to feasibility or at least stationarity with respect to the constraint infeasibility measure φ (recall (4)). Such a conclusion is offered under certain conditions by combining Corollary 1 (see page 17) and Theorem 3 (see page 19). The remaining aim (paralleling (26)) is that one should aim to prove that even if a (sub)sequence of $\{J_k\}$ tends to singularity, if the merit parameter sequence $\{\tau_k\}$ happens to remain bounded below by a positive real number, then $\{\|Z_k^T \nabla f(x_k)\|_2\}$ vanishes in expectation. This can also be seen to occur under certain conditions in Corollary 1 on page 17.

In addition, due to its stochastic nature, there are events that one should consider in which the algorithm may exhibit behavior that cannot be exhibited by the deterministic algorithm. One such event is when the merit parameter eventually remains fixed at a value that is not sufficiently small. We show in Section 4.3 that, under certain conditions in a run of the algorithm, the probability of the occurrence of this event is zero, and discuss what it means to assume that the total probability of this event (over all possible runs) is zero. We complete the picture of the possible behaviors of our algorithm by discussing remaining possible (practically irrelevant) events in Section 4.4.

Let us now commence our analysis of Algorithm 1. If a run terminates finitely at iteration $k \in \mathbb{N}$, then an infeasible stationary point has been found. Hence, without loss of generality throughout the remainder of our analysis and discussions, we assume that the algorithm does not terminate finitely, i.e., an infinite number of iterates are generated. As previously mentioned, for much of our analysis, we merely assume that the stochastic gradient estimates satisfy Assumption 2. This is done to show that many of our results hold under this general setting. However, we will ultimately impose a stronger condition on $\{g_k\}$ that is needed for cases when $\{\tau_k\} \searrow 0$; see Section 4.2.

We build to our main results through a series of lemmas. Our first lemma has appeared for various deterministic algorithms in the literature. It extends easily to our setting since the normal component computation is deterministic conditioned on the event that the algorithm reaches x_k.

Lemma 1. *There exist $\kappa_v \in \mathbb{R}_{>0}$ and $\underline{\omega} \in \mathbb{R}_{>0}$ such that, in any run of the algorithm,*

$$\begin{aligned}&\|c_k\|_2(\|c_k\|_2 - \|c_k + J_k v_k\|_2) \geq \kappa_v \|J_k^T c_k\|_2^2 \\ &\qquad \text{and} \;\; \underline{\omega}\|J_k^T c_k\|_2^2 \leq \|v_k\|_2 \leq \omega \|J_k^T c_k\|_2 \;\; \text{for all} \;\; k \in \mathbb{N} \; \text{with} \; \|c_k\|_2 > 0.\end{aligned}$$

Proof. Proof. The proof follows as for Lemmas 3.5 and 3.6 in [5]. □ □

Our second lemma shows that the procedure for setting $\{\chi_k\}$ and $\{\zeta_k\}$ guarantees that these sequences are constant *deterministically* for sufficiently large $k \in \mathbb{N}$.

Lemma 2. *There exist $(\chi_{\max}, \zeta_{\min}) \in \mathbb{R}_{>0} \times \mathbb{R}_{>0}$ such that, in any run, there exists $k_{\chi,\zeta} \in \mathbb{N}$ such that $(\chi_k, \zeta_k) = (\chi_{k_{\chi,\zeta}}, \zeta_{k_{\chi,\zeta}})$ for all $k \geq k_{\chi,\zeta}$, where $(\chi_{k_{\chi,\zeta}}, \zeta_{k_{\chi,\zeta}}) \in (0, \chi_{\max}] \times [\zeta_{\min}, \infty)$.*

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. If $d_k = 0$, then the algorithm sets $(\chi_k, \zeta_k) = (\chi_{k-1}, \zeta_{k-1})$. Otherwise, under Assumption 3, it follows for any $\chi \in \mathbb{R}_{>0}$ that $\|u_k\|_2^2 \geq \chi \|v_k\|_2^2$ implies

$$\begin{aligned}\tfrac{1}{2} d_k^T H_k d_k &= \tfrac{1}{2} u_k^T H_k u_k + u_k^T H_k v_k + \tfrac{1}{2} v_k^T H_k v_k \\ &\geq \tfrac{1}{2}\zeta \|u_k\|_2^2 - \|u_k\|_2 \|H_k\|_2 \|v_k\|_2 - \tfrac{1}{2}\|H_k\|_2 \|v_k\|_2^2 \geq \left(\frac{\zeta}{2} - \frac{\kappa_H}{\sqrt{\chi}} - \frac{\kappa_H}{2\chi}\right)\|u_k\|_2^2.\end{aligned}$$

Hence, for sufficiently large $\chi \in \mathbb{R}_{>0}$, one finds that $\|u_k\|_2^2 \geq \chi \|v_k\|_2^2$ implies $\frac{1}{2} d_k^T H_k d_k \geq \frac{1}{4}\zeta \|u_k\|_2^2$. The conclusion follows from this fact and the procedure for setting (χ_k, ζ_k) in (15)–(16). □ □

We now prove that the sequence $\{\xi_k\}$ is bounded below *deterministically.*

Lemma 3. *There exists $\xi_{\min} \in \mathbb{R}_{>0}$ such that, in any run of the algorithm, there exists $k_\xi \in \mathbb{N}$ such that $\xi_k = \xi_{k_\xi}$ for all $k \geq k_\xi$, where $\xi_{k_\xi} \in [\xi_{\min}, \infty)$.*

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. If $d_k = 0$, then the algorithm sets $\xi_k = \xi_{k-1}$. If $d_k \neq 0$ and $\|u_k\|_2^2 \geq \chi_k \|v_k\|_2^2$, then it follows from (13)–(14) and (17)–(18) that either $\xi_k = \xi_{k-1}$ or

$$\xi_k \geq (1 - \epsilon_\xi)\xi_k^{\text{trial}} = (1 - \epsilon_\xi)\left(\frac{\Delta l(x_k, \tau_k, g_k, d_k)}{\tau_k \|d_k\|_2^2}\right) \geq (1 - \epsilon_\xi)\frac{\frac{1}{2}\tau_k \zeta \|u_k\|_2^2}{\tau_k (1 + \chi_k^{-1})\|u_k\|_2^2} \geq (1 - \epsilon_\xi)\frac{\frac{1}{2}\zeta}{(1 + \chi_{-1}^{-1})}.$$

If $d_k \neq 0$ and $\|u_k\|_2^2 < \chi_k \|v_k\|_2^2$, then by (13)–(14), (17)–(18), and Lemmas 1–2 either $\xi_k = \xi_{k-1}$ or

$$\xi_k \geq (1 - \epsilon_\xi)\xi_k^{\text{trial}} = (1 - \epsilon_\xi)\left(\frac{\Delta l(x_k, \tau_k, g_k, d_k)}{\|d_k\|_2^2}\right) \geq (1 - \epsilon_\xi)\frac{\sigma \kappa_v \kappa_c^{-1} \|J_k^T c_k\|_2^2}{(\chi_k + 1)\omega^2 \|J_k^T c_k\|_2^2} \geq (1 - \epsilon_\xi)\frac{\sigma \kappa_v \kappa_c^{-1}}{(\chi_{\max} + 1)\omega^2}.$$

Combining these results, the desired conclusion follows. □ □

Our next two lemmas provide useful relationships between deterministic and stochastic quantities conditioned on the event that the algorithm has reached x_k as the kth iterate. The first result is similar to [1, Lemma 3.6], although the proof presented here is different in order to handle potential rank deficiency of the constraint Jacobians. Here and throughout the remainder of our analysis, conditioned on the event that the algorithm reaches x_k as the kth iterate, we denote u_k^{true} as the tangential component of the search direction that would be computed if $\nabla f(x_k)$ were used in place of g_k in (9), and similarly denote $d_k^{\text{true}} := v_k + u_k^{\text{true}}$.

Lemma 4. *For all $k \in \mathbb{N}$ in any run, $\mathbb{E}_k[u_k] = u_k^{\rm true}$ and $\mathbb{E}_k[\|d_k - d_k^{\rm true}\|_2] \leq \zeta^{-1}\sqrt{M}$.*

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. Under Assumption 3, it follows from (9) that there exist w_k and $w_k^{\rm true}$ such that $u_k = Z_k w_k$ and $u_k^{\rm true} = Z_k w_k^{\rm true}$ where $w_k = -(Z_k^T H_k Z_k)^{-1} Z_k^T(g_k + H_k v_k)$ and $w_k^{\rm true} = -(Z_k^T H_k Z_k)^{-1} Z_k^T(\nabla f(x_k) + H_k v_k)$. Since $(Z_k^T H_k Z_k)^{-1} Z_k^T$ and Z_k are linear operators, it follows that $\mathbb{E}_k[w_k] = w_k^{\rm true}$ and hence $\mathbb{E}_k[u_k] = u_k^{\rm true}$, as desired. Then, it follows from consistency and submultiplicity of the spectral norm, orthonormality of Z_k, Jensen's inequality, concavity of the square root operator, and Assumptions 2 and 3 that

$$\begin{aligned}
\mathbb{E}_k[\|d_k - d_k^{\rm true}\|_2] = \mathbb{E}_k[\|u_k - u_k^{\rm true}\|_2] &= \mathbb{E}_k[\|Z_k(w_k - w_k^{\rm true})\|_2] \\
&= \mathbb{E}_k[\|Z_k(Z_k^T H_k Z_k)^{-1} Z_k^T(g_k - \nabla f(x_k))\|_2] \\
&\leq \mathbb{E}_k[\|Z_k(Z_k^T H_k Z_k)^{-1} Z_k^T\|_2 \|g_k - \nabla f(x_k)\|_2] \\
&= \|(Z_k^T H_k Z_k)^{-1}\|_2 \mathbb{E}_k[\|g_k - \nabla f(x_k)\|_2] \\
&\leq \zeta^{-1} \mathbb{E}_k[\|g_k - \nabla f(x_k)\|_2] \\
&\leq \zeta^{-1}\sqrt{\mathbb{E}_k[\|g_k - \nabla f(x_k)\|_2^2]} \leq \zeta^{-1}\sqrt{M},
\end{aligned}$$

which is the final desired conclusion. □ □

Our next result is part of [1, Lemma 3.9]; we provide a proof for completeness.

Lemma 5. *For all $k \in \mathbb{N}$ in any run, $\nabla f(x_k)^T d_k^{\rm true} \geq \mathbb{E}_k[g_k^T d_k] \geq \nabla f(x_k)^T d_k^{\rm true} - \zeta^{-1} M$.*

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. The arguments in the proof of Lemma 4 give

$$\begin{aligned}
g_k^T u_k &= -g_k^T Z_k (Z_k^T H_k Z_k)^{-1} Z_k^T (g_k + H_k v_k) \\
\text{and} \ \ \nabla f(x_k)^T u_k^{\rm true} &= -\nabla f(x_k)^T Z_k (Z_k^T H_k Z_k)^{-1} Z_k^T (\nabla f(x_k) + H_k v_k).
\end{aligned}$$

On the other hand, under Assumptions 2 and 3, it follows that

$$\zeta^{-1} M \geq \mathbb{E}_k[\|Z_k^T(g_k - \nabla f(x_k))\|^2_{(Z_k^T H_k Z_k)^{-1}}] \geq 0,$$

where

$$\begin{aligned}
&\mathbb{E}_k[\|Z_k^T(g_k - \nabla f(x_k))\|^2_{(Z_k^T H_k Z_k)^{-1}}] \\
&= \mathbb{E}_k[\|Z_k^T g_k\|^2_{(Z_k^T H_k Z_k)^{-1}}] - 2\mathbb{E}_k[g_k^T Z_k (Z_k^T H_k Z_k)^{-1} Z_k^T \nabla f(x_k)] + \|Z_k^T \nabla f(x_k)\|^2_{(Z_k^T H_k Z_k)^{-1}} \\
&= \mathbb{E}_k[\|Z_k^T g_k\|^2_{(Z_k^T H_k Z_k)^{-1}}] - \|Z_k^T \nabla f(x_k)\|^2_{(Z_k^T H_k Z_k)^{-1}}.
\end{aligned}$$

Combining the facts above and again using Assumption 2, it follows that

$$\begin{aligned}
\nabla f(x_k)^T d_k^{\rm true} - \mathbb{E}_k[g_k^T d_k] &= \nabla f(x_k)^T v_k + \nabla f(x_k)^T u_k^{\rm true} - \mathbb{E}_k[g_k^T v_k + g_k^T u_k] \\
&= \nabla f(x_k)^T u_k^{\rm true} - \mathbb{E}_k[g_k^T u_k] \\
&= -\nabla f(x_k)^T Z_k (Z_k^T H_k Z_k)^{-1} Z_k^T (\nabla f(x_k) + H_k v_k) \\
&\quad + \mathbb{E}_k[g_k^T Z_k (Z_k^T H_k Z_k)^{-1} Z_k^T (g_k + H_k v_k)] \\
&= -\|Z_k^T \nabla f(x_k)\|^2_{(Z_k^T H_k Z_k)^{-1}} + \mathbb{E}_k[\|Z_k^T g_k\|^2_{(Z_k^T H_k Z_k)^{-1}}] \in [0, \zeta^{-1} M],
\end{aligned}$$

which gives the desired conclusion. □ □

In the subsequent subsections, our analysis turns to offering guarantees conditioned on each of a few possible events that occur in a run of the algorithm, two of which involve the merit parameter sequence eventually remaining constant in a run of the algorithm. Before considering these events, let us first prove under certain circumstances that such behavior of the merit parameter sequence would occur. As seen in Theorem 1, it is worthwhile to consider such an occurrence regardless of the properties of the sequence of constraint Jacobians. That said, one might only be able to prove that it occurs when the constraint Jacobians are eventually bounded away from singularity.

Our first lemma here proves that if the constraint Jacobians are eventually bounded away from singularity, then the normal components of the search directions satisfy a useful upper bound. The proof is essentially the same as that of [5, Lemma 3.15], but we provide it for completeness.

Lemma 6. *If, in a run of the algorithm, there exist $k_J \in \mathbb{N}$ and $\sigma_J \in \mathbb{R}_{>0}$ such that the singular values of J_k are bounded below by σ_J for all $k \geq k_J$, then there exists $\kappa_\omega \in \mathbb{R}_{>0}$ such that*

$$\|v_k\|_2 \leq \kappa_\omega(\|c_k\|_2 - \|c_k + J_k v_k\|_2) \ \ \textit{for all} \ \ k \geq k_J.$$

Proof. Proof. Under the conditions of the lemma, $\|J_k^T c_k\|_2 \geq \sigma_J \|c_k\|_2$ for all $k \geq k_J$. Hence, along with Lemma 1, it follows that $\|c_k\|_2(\|c_k\|_2 - \|c_k + J_k v_k\|_2) \geq \kappa_v \|J_k^T c_k\|_2^2 \geq \kappa_v \sigma_J^2 \|c_k\|_2^2$ for all $k \geq k_J$. Combining this again with Lemma 1, it follows with the Cauchy-Schwarz inequality and (2) that

$$\|v_k\|_2 \leq \omega \|J_k^T\|_2 \|c_k\|_2 \leq \frac{\omega \kappa_J}{\kappa_v \sigma_J^2}(\|c_k\|_2 - \|c_k + J_k v_k\|_2) \ \ \text{for all} \ \ k \geq k_J,$$

from which the desired conclusion follows. □ □

We now prove that if the differences between the stochastic gradient estimates and the true gradients are bounded deterministically, then the sequence of tangential components is bounded. We remark that it is possible for the tangential component sequence to remain bounded even if such a condition on the stochastic gradient estimates does not hold.

Lemma 7. *If, in a run of the algorithm, the sequence $\{\|g_k - \nabla f(x_k)\|_2\}$ is bounded by a positive real number $\kappa_g \in \mathbb{R}_{>0}$, then the sequence $\{\|u_k\|_2\}$ is bounded by a positive real number $\kappa_u \in \mathbb{R}_{>0}$.*

Proof. Proof. Under Assumption 1, the sequence $\{\|\nabla f(x_k)\|_2\}$ is bounded; recall (2). Hence, under the conditions of the lemma, $\{\|g_k\|_2\}$ is bounded. The first block of (9) yields $u_k^T H_k u_k = -u_k^T(g_k + H_k v_k)$, which under Assumption 3 yields $\zeta \|u_k\|_2^2 \leq -u_k^T g_k - u_k^T H_k v_k \leq (\|g_k\|_2 + \|H_k\|_2 \|v_k\|_2)\|u_k\|_2$. Hence, the conclusion follows from these facts, Assumption 1, Assumption 3, and Lemma 1. □ □

By combining the preceding two lemmas, the following lemma indicates certain circumstances under which the sequence of merit parameters will eventually remain constant. We remark that it is possible for the merit parameter sequence to remain constant eventually even if the conditions of the lemma do not hold, which is why our analyses in the subsequent subsections do not presume that these conditions hold. That said, to prove that there exist settings in which the merit parameter is guaranteed to remain constant eventually, we offer the following lemma.

Lemma 8. *If, in a run, there exist $k_J \in \mathbb{N}$ and $\sigma_J \in \mathbb{R}_{>0}$ such that the singular values of J_k are bounded below by σ_J for all $k \geq k_J$ and $\{\|g_k - \nabla f(x_k)\|_2\}$ is bounded by a positive real number $\kappa_g \in \mathbb{R}_{>0}$, then there exist $k_\tau \in \mathbb{N}$ and $\tau_{\min} \in \mathbb{R}_{>0}$ such that $\tau_k = \tau_{\min}$ for all $k \in \mathbb{N}$ with $k \geq k_\tau$.*

Proof. Proof. Observe that the algorithm terminates if $\|J_k^T c_k\|_2 = 0$ while $\|c_k\|_2 > 0$. Let us now show that if $\|c_k\|_2 = 0$, then the algorithm sets $\tau_k \leftarrow \tau_{k-1}$. Indeed, $\|c_k\|_2 = 0$ implies $v_k = 0$ by Lemma 1. If $u_k = 0$ as well, then $d_k = 0$ and the algorithm explicitly sets $\tau_k \leftarrow \tau_{k-1}$. Otherwise, if $v_k = 0$ and $u_k \neq 0$, then (9) yields $g_k^T u_k + u_k^T H_k u_k = g_k^T d_k + u_k^T H_k u_k$, in which case (11)–(12) again yield $\tau_k \leftarrow \tau_{k-1}$. Overall, it follows that $\tau_k < \tau_{k-1}$ if and only if one finds $\|J_k^T c_k\|_2 > 0$, $g_k^T d_k + u_k^T H_k u_k > 0$, and $\tau_{k-1}(g_k^T d_k + u_k^T H_k u_k) > (1 - \sigma)(\|c_k\|_2 - \|c_k + J_k v_k\|_2)$. On the other hand, from the first equation in (9), the Cauchy-Schwarz inequality, (2), and Lemmas 6 and 7, it holds that

$$\begin{aligned} g_k^T d_k + u_k^T H_k u_k = (g_k - H_k u_k)^T v_k &= (g_k - \nabla f(x_k) + \nabla f(x_k) - H_k u_k)^T v_k \\ &\leq (\kappa_g + \kappa_{\nabla f} + \kappa_H \kappa_u)\|v_k\|_2 \\ &\leq (\kappa_g + \kappa_{\nabla f} + \kappa_H \kappa_u)\kappa_\omega(\|c_k\|_2 - \|c_k + J_k v_k\|_2). \end{aligned}$$

Combining these facts, the desired conclusion follows. □ □

4.1 Constant, Sufficiently Small Merit Parameter

Our goal in this subsection is to prove a convergence guarantee for our algorithm in the event $E_{\tau,\mathrm{low}}$, which is defined formally in the assumption below. In the assumption, similar to our notation of u_k^{true} and d_k^{true}, we use $\tau_k^{\mathrm{trial,true}}$ to denote the value of τ_k^{trial} that, conditioned on x_k as the kth iterate, would be computed in iteration $k \in \mathbb{N}$ if the search direction were computed using the true gradient $\nabla f(x_k)$ in place of g_k in (9).

Assumption 4. *In a run of the algorithm, event $E_{\tau,\mathrm{low}}$ occurs, i.e., there exists an iteration number $k_\tau \in \mathbb{N}$ and a merit parameter value $\tau_{\min} \in \mathbb{R}_{>0}$ such that*

$$\tau_k = \tau_{\min} \le \tau_k^{\mathrm{trial,true}}, \quad \chi_k = \chi_{k-1}, \quad \zeta_k = \zeta_{k-1}, \quad \text{and} \quad \xi_k = \xi_{k-1} \quad \text{for all} \quad k \ge k_\tau.$$

In addition, along the lines of Assumption 2, $\{g_k\}_{k \ge k_\tau}$ satisfies $\mathbb{E}_{k,\tau,\mathrm{low}}[g_k] = \nabla f(x_k)$ and $\mathbb{E}_{k,\tau,\mathrm{low}}[\|g_k - \nabla f(x_k)\|_2^2] \le M$, where $\mathbb{E}_{k,\tau,\mathrm{low}}$ denotes expectation with respect to the distribution of ι conditioned on the event that $E_{\tau,\mathrm{low}}$ occurs and the algorithm has reached x_k in iteration $k \in \mathbb{N}$.

Recall from Lemmas 2 and 3 that the sequences $\{\chi_k\}$, $\{\zeta_k\}$, and $\{\xi_k\}$ are guaranteed to be bounded deterministically, and in particular will remain constant for sufficiently large $k \in \mathbb{N}$. Hence, one circumstance in which Assumption 4 may hold is under the conditions of Lemma 8. A critical distinction in Assumption 4 is that the value at which the merit parameter eventually settles is sufficiently small such that $\tau_k \le \tau_k^{\mathrm{trial,true}}$ for all sufficiently large $k \in \mathbb{N}$. This is the key distinction between the event $E_{\tau,\mathrm{low}}$ and some of the events we consider in Sections 4.3 and 4.4.

For the sake of brevity in the rest of this subsection, let us temporarily redefine $\mathbb{E}_k := \mathbb{E}_{k,\tau,\mathrm{low}}$.

Our next lemma provides a key result that drives our analysis for this subsection. It shows that as long as β_k is sufficiently small for all $k \in \mathbb{N}$ (in a manner similar to (23)), the reduction in the merit function in each iteration is at least the reduction in the model of the merit function corresponding to the *true* gradient and its associated search direction minus quantities that can be attributed to the error in the stochastic gradient estimate.

Lemma 9. *Suppose that $\{\beta_k\}$ is chosen such that*

$$\beta_k \in (0,1] \quad \text{and} \quad \frac{2(1-\eta)\beta_k\xi_k \max\{\tau_k, 1\}}{\tau_k L + \Gamma} \in (0,1] \quad \text{for all} \quad k \in \mathbb{N}. \tag{28}$$

Then, for all $k \in \mathbb{N}$ in any such run of the algorithm, it follows that

$$\begin{aligned}&\phi(x_k, \tau_k) - \phi(x_k + \alpha_k d_k, \tau_k) \\ \ge\ &\alpha_k \Delta l(x_k, \tau_k, \nabla f(x_k), d_k^{\mathrm{true}}) - (1-\eta)\alpha_k\beta_k\Delta l(x_k, \tau_k, g_k, d_k) - \alpha_k\tau_k\nabla f(x_k)^T(d_k - d_k^{\mathrm{true}}).\end{aligned}$$

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. From (21)–(22) and the supposition about $\{\beta_k\}$, one finds $\alpha_k \in (0,1]$. Hence, with (19) and $J_k d_k = J_k d_k^{\mathrm{true}}$ (since $J_k u_k = J_k u_k^{\mathrm{true}} = 0$ by (9)), one has

$$\begin{aligned}&\phi(x_k, \tau_k) - \phi(x_k + \alpha_k d_k, \tau_k) \\ \ge\ &-\alpha_k(\tau_k \nabla f(x_k)^T d_k - \|c_k\|_2 + \|c_k + J_k d_k\|_2) - \tfrac{1}{2}(\tau_k L + \Gamma)\alpha_k^2\|d_k\|_2^2 \\ =\ &-\alpha_k(\tau_k\nabla f(x_k)^T d_k^{\mathrm{true}} - \|c_k\|_2 + \|c_k + J_k d_k^{\mathrm{true}}\|_2) - \tfrac{1}{2}(\tau_k L + \Gamma)\alpha_k^2\|d_k\|_2^2 - \alpha_k\tau_k\nabla f(x_k)^T(d_k - d_k^{\mathrm{true}}) \\ =\ &\alpha_k\Delta l(x_k, \tau_k, \nabla f(x_k), d_k^{\mathrm{true}}) - \tfrac{1}{2}(\tau_k L + \Gamma)\alpha_k^2\|d_k\|_2^2 - \alpha_k\tau_k\nabla f(x_k)^T(d_k - d_k^{\mathrm{true}}).\end{aligned} \tag{29}$$

By (21), it follows that $\alpha_k^{\mathrm{suff}} \le \frac{2(1-\eta)\beta_k\Delta l(x_k,\tau_k,g_k,d_k)}{(\tau_k L+\Gamma)\|d_k\|_2^2}$. If $\|u_k\|_2^2 \ge \chi_k\|v_k\|_2^2$, then it follows from (17)–(18) that $\xi_k \le \xi_k^{\mathrm{trial}} = \frac{\Delta l(x_k,\tau_k,g_k,d_k)}{\tau_k\|d_k\|_2^2}$ and $\frac{2(1-\eta)\beta_k\Delta l(x_k,\tau_k,g_k,d_k)}{(\tau_k L+\Gamma)\|d_k\|_2^2} \ge \frac{2(1-\eta)\beta_k\xi_k\tau_k}{\tau_k L+\Gamma}$. On the other hand, if $\|u_k\|_2^2 < \chi_k\|v_k\|_2^2$, then it follows from (17)–(18) that $\xi_k \le \xi_k^{\mathrm{trial}} = \frac{\Delta l(x_k,\tau_k,g_k,d_k)}{\|d_k\|_2^2}$ and $\frac{2(1-\eta)\beta_k\Delta l(x_k,\tau_k,g_k,d_k)}{(\tau_k L+\Gamma)\|d_k\|_2^2} \ge \frac{2(1-\eta)\beta_k\xi_k}{\tau_k L+\Gamma}$. It follows from these facts and the supposition about $\{\beta_k\}$ that the projection in (22) never sets $\alpha_k > \alpha_k^{\rm suff}$. Thus, $\alpha_k \leq \alpha_k^{\rm suff} \leq \frac{2(1-\eta)\beta_k \Delta l(x_k,\tau_k,g_k,d_k)}{(\tau_k L+\Gamma)\|d_k\|_2^2}$. Hence, by (29),

$$\begin{aligned}
&\phi(x_k,\tau_k) - \phi(x_k + \alpha_k d_k, \tau_k) \\
\geq\ & \alpha_k \Delta l(x_k,\tau_k,\nabla f(x_k), d_k^{\rm true}) - \tfrac{1}{2}\alpha_k(\tau_k L + \Gamma)\left(\tfrac{2(1-\eta)\beta_k \Delta l(x_k,\tau_k,g_k,d_k)}{(\tau_k L+\Gamma)\|d_k\|_2^2}\right)\|d_k\|_2^2 - \alpha_k \tau_k \nabla f(x_k)^T(d_k - d_k^{\rm true}) \\
=\ & \alpha_k \Delta l(x_k,\tau_k,\nabla f(x_k), d_k^{\rm true}) - (1-\eta)\alpha_k\beta_k \Delta l(x_k,\tau_k,g_k,d_k) - \alpha_k\tau_k\nabla f(x_k)^T(d_k - d_k^{\rm true}),
\end{aligned}$$

which completes the proof. □ □

Our second result in this case offers a critical upper bound on the final term in the conclusion of Lemma 9. The result follows in a similar manner as [1, Lemma 3.11].

Lemma 10. *For any run under Assumption 4, it follows for any $k \geq k_\tau$ that*

$$\mathbb{E}_k[\alpha_k \tau_k \nabla f(x_k)^T(d_k - d_k^{\rm true})] \leq \beta_k^2 \theta \tau_{\min}\kappa_{\nabla f}\zeta^{-1}\sqrt{M}.$$

Proof. Proof. Using (2) and Lemma 4, the result follows in the same manner as [1, Lemma 3.11] since, under Assumption 4, the stepsize α_k for all $k \geq k_\tau$ is contained within an interval of the form

$$\left[\min\left\{\frac{2(1-\eta)\beta_k\hat{\xi}_{\min}\tau_{\min}}{\tau_{\min}L+\Gamma}, \frac{2(1-\eta)\beta_k\hat{\xi}_{\min}}{\tau_{\min}L+\Gamma}\right\}, \max\left\{\frac{2(1-\eta)\beta_k\hat{\xi}_{\min}\tau_{\min}}{\tau_{\min}L+\Gamma}, \frac{2(1-\eta)\beta_k\hat{\xi}_{\min}}{\tau_{\min}L+\Gamma}\right\} + \theta\beta_k^2\right],$$

where, for a given run of the algorithm, $\hat{\xi}_{\min} \in [\xi_{\min},\infty)$ is the positive real number such that $\xi_k = \hat{\xi}_{\min}$ for all $k \geq k_\tau$ whose existence follows from Lemma 3 and Assumption 4. □ □

Our next result in this case bounds the middle term in the conclusion of Lemma 9.

Lemma 11. *For any run under Assumption 4, it follows for any $k \geq k_\tau$ that*

$$\mathbb{E}_k[\Delta l(x_k,\tau_{\min},g_k,d_k)] \leq \Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\rm true}) + \tau_{\min}\zeta^{-1}M.$$

Proof. Proof. Consider arbitrary $k \geq k_\tau$ in any run under Assumption 4. By Assumption 4, it follows from the model reduction definition (13) and Lemma 5 that

$$\begin{aligned}
\mathbb{E}_k[\Delta l(x_k,\tau_k,g_k,d_k)] &= \mathbb{E}_k[-\tau_{\min}g_k^T d_k + \|c_k\|_2 - \|c_k + J_k d_k\|_2] \\
&\leq -\tau_{\min}\nabla f(x_k)^T d_k^{\rm true} + \tau_{\min}\zeta^{-1}M + \|c_k\|_2 - \|c_k + J_k d_k^{\rm true}\|_2 \\
&= \Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\rm true}) + \tau_{\min}\zeta^{-1}M,
\end{aligned}$$

as desired. □ □

We now prove our main theorem of this subsection, where $\mathbb{E}_{\tau,{\rm low}}[\,\cdot\,] := \mathbb{E}[\,\cdot \mid \text{Assumption 4 holds}]$.

Theorem 2. *Suppose that Assumption 4 holds and the sequence $\{\beta_k\}$ is chosen such that* (28) *holds for all $k \in \mathbb{N}$. For a given run of the algorithm, define $\hat{\xi}_{\min} \in \mathbb{R}_{>0}$ as the value in Assumption 4 such that $\xi_k = \hat{\xi}_{\min}$ for all $k \geq k_\tau$ and define*

$$\underline{A} := \min\left\{\tfrac{2(1-\eta)\hat{\xi}_{\min}\tau_{\min}}{\tau_{\min}L+\Gamma}, \tfrac{2(1-\eta)\hat{\xi}_{\min}}{\tau_{\min}L+\Gamma}\right\}, \quad \overline{A} := \max\left\{\tfrac{2(1-\eta)\hat{\xi}_{\min}\tau_{\min}}{\tau_{\min}L+\Gamma}, \tfrac{2(1-\eta)\hat{\xi}_{\min}}{\tau_{\min}L+\Gamma}\right\},$$
$$\text{and}\ \ \overline{M} := \tau_{\min}\zeta^{-1}((1-\eta)(\overline{A}+\theta)M + \theta\kappa_{\nabla f}\sqrt{M}).$$

If $\beta_k = \beta \in (0, \underline{A}/((1-\eta)(\overline{A}+\theta)))$ for all $k \geq k_\tau$, then

$$\begin{aligned}
&\mathbb{E}_{\tau,{\rm low}}\left[\frac{1}{k+1}\sum_{j=k_\tau}^{k_\tau+k}\Delta l(x_j,\tau_{\min},\nabla f(x_j),d_j^{\rm true})\right] \\
&\leq \frac{\beta\overline{M}}{\underline{A}-(1-\eta)(\overline{A}+\theta)\beta} + \frac{\mathbb{E}_{\tau,{\rm low}}[\phi(x_{k_\tau},\tau_{\min})]-\phi_{\min}}{(k+1)\beta(\underline{A}-(1-\eta)(\overline{A}+\theta)\beta)} \xrightarrow{k\to\infty} \frac{\beta\overline{M}}{\underline{A}-(1-\eta)(\overline{A}+\theta)\beta},
\end{aligned} \tag{30}$$

where, in the context of Assumption 1, $\phi_{\min} \in \mathbb{R}_{>0}$ is a lower bound for $\phi(\cdot,\tau_{\min})$ over $\mathcal{X}$. On the other hand, if $\sum_{k=k_\tau}^{\infty} \beta_k = \infty$ and $\sum_{k=k_\tau}^{\infty} \beta_k^2 < \infty$, then

$$\lim_{k\to\infty} \mathbb{E}_{\tau,\text{low}}\left[\frac{1}{(\sum_{j=k_\tau}^{k_\tau+k}\beta_j)} \sum_{j=k_\tau}^{k_\tau+k} \beta_j \Delta l(x_j,\tau_{\min},\nabla f(x_j),d_j^{\text{true}})\right] = 0. \tag{31}$$

Proof. Proof. Consider arbitrary $k \geq k_\tau$ in any run under Assumption 4. From the definitions of $\underline{A}$ and $\overline{A}$ in the statement of the theorem, the manner in which the stepsizes are set by (22), and the fact that $\beta_k \in (0,1]$, it follows that $\underline{A}\beta_k \leq \alpha_k \leq (\overline{A}+\theta)\beta_k$. Hence, it follows from Lemmas 9–11 and the conditions of the theorem that

$$\begin{aligned}
&\phi(x_k,\tau_{\min}) - \mathbb{E}_k[\phi(x_k+\alpha_k d_k,\tau_{\min})] \\
&\geq \mathbb{E}_k[\alpha_k\Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\text{true}}) - (1-\eta)\alpha_k\beta_k\Delta l(x_k,\tau_{\min},g_k,d_k) - \alpha_k\tau_{\min}\nabla f(x_k)^T(d_k-d_k^{\text{true}})] \\
&\geq \beta_k(\underline{A}-(1-\eta)(\overline{A}+\theta)\beta_k)\Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\text{true}}) - \beta_k^2\overline{M}.
\end{aligned}$$

If $\beta_k = \beta \in (0,\underline{A}/((1-\eta)(\overline{A}+\theta)))$ for all $k \geq k_\tau$, then total expectation under Assumption 4 yields

$$\begin{aligned}
&\mathbb{E}_{\tau,\text{low}}[\phi(x_k,\tau_{\min})] - \mathbb{E}_{\tau,\text{low}}[\phi(x_k+\alpha_k d_k,\tau_{\min})] \\
&\qquad\geq \beta(\underline{A}-(1-\eta)(\overline{A}+\theta)\beta)\mathbb{E}_{\tau,\text{low}}[\Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\text{true}})] - \beta^2\overline{M} \ \text{ for all } \ k \geq k_\tau.
\end{aligned}$$

Summing this inequality for $j \in \{k_\tau,\dots,k_\tau+k\}$, it follows under Assumption 1 that

$$\begin{aligned}
&\mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau},\tau_{\min})] - \phi_{\min} \\
&\geq \mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau},\tau_{\min})] - \mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau+k+1},\tau_{\min})] \\
&\geq \beta(\underline{A}-(1-\eta)(\overline{A}+\theta)\beta)\mathbb{E}_{\tau,\text{low}}\left[\sum_{j=k_\tau}^{k_\tau+k}\Delta l(x_j,\tau_{\min},\nabla f(x_j),d_j^{\text{true}})\right] - (k+1)\beta^2\overline{M},
\end{aligned}$$

from which (30) follows. On the other hand, if $\{\beta_k\}$ satisfies $\sum_{k=k_\tau}^{\infty}\beta_k = \infty$ and $\sum_{k=k_\tau}^{\infty}\beta_k^2 < \infty$, then it follows for sufficiently large $k \geq k_\tau$ that $\beta_k \leq \eta\underline{A}/((1-\eta)(\overline{A}+\theta))$; hence, without loss of generality, let us assume that this inequality holds for all $k \geq k_\tau$, which implies that $\underline{A}-(1-\eta)(\overline{A}+\theta)\beta_k \geq (1-\eta)\underline{A}$ for all $k \geq k_\tau$. As above, it follows that

$$\begin{aligned}
&\mathbb{E}_{\tau,\text{low}}[\phi(x_k,\tau_{\min})] - \mathbb{E}_{\tau,\text{low}}[\phi(x_k+\alpha_k d_k,\tau_{\min})] \\
&\qquad\geq (1-\eta)\underline{A}\beta_k\mathbb{E}_{\tau,\text{low}}[\Delta l(x_k,\tau_{\min},\nabla f(x_k),d_k^{\text{true}})] - \beta_k^2\overline{M} \ \text{ for all } \ k \geq k_\tau.
\end{aligned}$$

Summing this inequality for $j \in \{k_\tau,\dots,k_\tau+k\}$, it follows under Assumption 1 that

$$\begin{aligned}
\mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau},\tau_{\min})] - \phi_{\min} &\geq \mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau},\tau_{\min})] - \mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau+k+1},\tau_{\min})] \\
&\geq (1-\eta)\underline{A}\mathbb{E}_{\tau,\text{low}}\left[\sum_{j=k_\tau}^{k_\tau+k}\beta_j\Delta l(x_j,\tau_{\min},\nabla f(x_j),d_j^{\text{true}})\right] - \overline{M}\sum_{j=k_\tau}^{k_\tau+k}\beta_j^2.
\end{aligned}$$

Rearranging this inequality yields

$$\mathbb{E}_{\tau,\text{low}}\left[\sum_{j=k_\tau}^{k_\tau+k}\beta_j\Delta l(x_j,\tau_{\min},g_j,d_j^{\text{true}})\right] \leq \frac{\mathbb{E}_{\tau,\text{low}}[\phi(x_{k_\tau},\tau_{\min})]-\phi_{\min}}{(1-\eta)\underline{A}} + \frac{\overline{M}}{(1-\eta)\underline{A}}\sum_{j=k_\tau}^{k_\tau+k}\beta_j^2,$$

from which (31) follows. □ □

We end this subsection with a corollary in which we connect the result of Theorem 2 to first-order stationarity measures (recall (3)). For this corollary, we require the following lemma.

Lemma 12. *For all $k \in \mathbb{N}$, it holds that $\|u_k^{\rm true}\|_2 \geq \kappa_H^{-1}\|Z_k^T(\nabla f(x_k) + H_k v_k)\|_2$.*

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run. As in the proof of Lemma 4, $Z_k^T H_k Z_k w_k^{\rm true} = -Z_k^T(\nabla f(x_k) + H_k v_k)$, meaning with Assumption 3 that $\|u_k^{\rm true}\|_2 \geq \kappa_H^{-1}\|Z_k^T(\nabla f(x_k) + H_k v_k)\|_2$. □ □

Corollary 1. *Under the conditions of Theorem 2, the following hold true.*

(a) If $\beta_k = \beta \in (0, \underline{A}/((1-\eta)(\overline{A}+\theta)))$ for all $k \geq k_\tau$, then

$$\mathbb{E}_{\tau,{\rm low}}\left[\frac{1}{k+1}\sum_{j=k_\tau}^{k_\tau+k}\left(\frac{\|Z_j^T(\nabla f(x_j) + H_j v_k)\|_2^2}{\kappa_H^2} + \frac{\kappa_v\|J_j^T c_j\|_2^2}{\kappa_c}\right)\right]$$
$$\leq \frac{\beta\overline{M}}{\underline{A} - (1-\eta)(\overline{A}+\theta)\beta} + \frac{\mathbb{E}_{\tau,{\rm low}}[\phi(x_{k_\tau},\tau_{\min})] - \phi_{\min}}{(k+1)\beta(\underline{A} - (1-\eta)(\overline{A}+\theta)\beta)} \xrightarrow{k\to\infty} \frac{\beta\overline{M}}{\underline{A} - (1-\eta)(\overline{A}+\theta)\beta}.$$

(b) If $\sum_{k=k_\tau}^\infty \beta_k = \infty$ and $\sum_{k=k_\tau}^\infty \beta_k^2 < \infty$, then

$$\lim_{k\to\infty}\mathbb{E}_{\tau,{\rm low}}\left[\frac{1}{\left(\sum_{j=k_\tau}^{k_\tau+k}\beta_j\right)}\sum_{j=k_\tau}^{k_\tau+k}\beta_j\left(\frac{\|Z_j^T(\nabla f(x_j) + H_j v_j)\|_2^2}{\kappa_H^2} + \frac{\kappa_v\|J_j^T c_j\|_2^2}{\kappa_c}\right)\right] = 0,$$

from which it follows that

$$\liminf_{k\to\infty}\ \mathbb{E}_{\tau,{\rm low}}\left[\frac{\|Z_k^T(\nabla f(x_k) + H_k v_k)\|_2^2}{\kappa_H^2} + \frac{\kappa_v\|J_k^T c_k\|_2^2}{\kappa_c}\right] = 0.$$

Proof. Proof. For all $k \in \mathbb{N}$, it follows under Assumption 4 that (14) holds with $\nabla f(x_k)$ in place of g_k and $u_k^{\rm true}$ in place of u_k. The result follows from this fact, Theorem 2, and Lemmas 1 and 12. □ □

Observe that if the singular values of J_k are bounded below by $\sigma_J \in \mathbb{R}_{>0}$ for all $k \geq k_J$ for some $k_J \in \mathbb{N}$, then (as in the proof of Lemma 6) it follows that $\|J_k^T c_k\|_2 \geq \sigma_J\|c_k\|_2$ for all $k \geq k_J$. In this case, the results of Corollary 1 hold with $\sigma_J\|c_k\|_2$ in place of $\|J_k^T c_k\|_2$. Overall, Corollary 1 offers results for the stochastic setting that parallel the limits (24) and (26) for the deterministic setting. The only difference is the presence of $Z_k^T H_k v_k$ in the term involving the reduced gradient $Z_k^T\nabla f(x_k)$ for all $k \in \mathbb{N}$. However, this does not significantly weaken the conclusion. After all, it follows from (5) (see also Lemma 1) that $\|v_k\|_2 \leq \omega\|J_k^T c_k\|_2$ for all $k \in \mathbb{N}$. Hence, since Corollary 1 shows that at least a subsequence of $\{\|J_k^T c_k\|_2\}$ tends to vanish in expectation, it follows that $\{\|v_k\|_2\}$ vanishes in expectation along the same subsequence of iterations. This, along with Assumption 3 and the orthonormality of Z_k, shows that $\{\|Z_k^T H_k v_k\|_2\}$ exhibits this same behavior, which means that from the corollary one finds that a subsequence of $\{\|Z_k^T\nabla f(x_k)\|_2\}$ vanishes in expectation.

Let us conclude this subsection with a few remarks on how one should interpret its main conclusions. First, one learns from the requirements on $\{\beta_k\}$ in Lemma 9 and Theorem 2/Corollary 1 that, rather than employ a prescribed sequence $\{\beta_k\}$, one should instead prescribe $\{\hat\beta_j\}_{j=0}^\infty \subset (0,1]$ and for each $k \in \mathbb{N}$ set β_k based on whether or not an adaptive parameter changes its value. In particular, anytime $k \in \mathbb{N}$ sees either $\tau_k < \tau_{k-1}$, $\chi_k > \chi_{k-1}$, $\zeta_k < \zeta_{k-1}$, or $\xi_k < \xi_{k-1}$, the algorithm should set $\beta_{k+j} \leftarrow \lambda\hat\beta_j$ for $j = 0, 1, 2, \ldots$ (continuing indefinitely or until $\hat k \in \mathbb{N}$ with $\hat k > k$ sees $\tau_{\hat k} < \tau_{\hat k-1}$, $\chi_{\hat k} > \chi_{\hat k-1}$, $\zeta_{\hat k} < \zeta_{\hat k-1}$, or $\xi_{\hat k} < \xi_{\hat k-1}$), where $\lambda \in \mathbb{R}_{>0}$ is chosen sufficiently small such that (28) holds. Since such a "reset" of $j \leftarrow 0$ will occur only a finite number of times under event $E_{\tau,{\rm low}}$, one of the desirable results in Theorem 2/Corollary 1 can be attained if $\{\hat\beta_j\}$ is chosen as an appropriate constant or diminishing sequence. Second, let us note that due to the generality of Assumption 4, it is possible that for different runs of the algorithm the corresponding terminal merit parameter value, namely, $\tau_{\min}$, in Assumption 4 could become arbitrarily close to zero. (This is in contrast to the conditions of Lemma 8, which guarantee a *uniform* lower bound for the merit parameter over all runs satisfying these conditions.) Hence, while our main conclusions of this subsection hold under looser conditions than those in Lemma 8, one should be wary in practice if/when the merit parameter sequence reaches small numerical values.

4.2 Vanishing Merit Parameter

Let us now consider the behavior of the algorithm in settings in which the merit parameter vanishes; in particular, we make Assumption 5 below.

Assumption 5. *In a run of the algorithm, event $E_{\tau,\text{zero}}$ occurs, i.e., $\{\tau_k\} \searrow 0$. In addition, along the lines of Assumption 2, the stochastic gradient sequence $\{g_k\}$ satisfies $\mathbb{E}_{k,\tau,\text{zero}}[g_k] = \nabla f(x_k)$ and $\|g_k - \nabla f(x_k)\|_2^2 \leq M$, where $\mathbb{E}_{k,\tau,\text{zero}}$ denotes expectation with respect to the distribution of ι conditioned on the event that $E_{\tau,\text{zero}}$ occurs and the algorithm has reached x_k in iteration $k \in \mathbb{N}$.*

Recalling Theorem 1 and Lemma 8, one may conclude in general that the merit parameter sequence may vanish for one of two reasons: a (sub)sequence of constraint Jacobians tends toward rank deficiency or a (sub)sequence of stochastic gradient estimates diverges. Our assumption here assumes that the latter event does not occur. (In our remarks in Section 4.4, we discuss the obstacles that arise in proving convergence guarantees when the merit parameter vanishes and the stochastic gradient estimates diverge.) Given our setting of constrained optimization, it is reasonable and consistent with Theorem 1 to have convergence toward stationarity with respect to the constraint violation measure as the primary goal in these circumstances.

For the sake of brevity in the rest of this subsection, let us temporarily redefine $\mathbb{E}_k := \mathbb{E}_{k,\tau,\text{zero}}$.

Our first result in this subsection is an alternative of Lemma 9.

Lemma 13. *Under Assumption 5 and assuming that $\{\beta_k\}$ is chosen such that* (28) *holds for all $k \in \mathbb{N}$, it follows for all $k \in \mathbb{N}$ that*

$$\begin{aligned}&\|c_k\|_2 - \|c(x_k + \alpha_k d_k)\|_2 \\ &\geq \alpha_k(1-(1-\eta)\beta_k)\Delta l(x_k,\tau_k,g_k,d_k) - \tau_k(f_k - f(x_k+\alpha_k d_k)) - \alpha_k\tau_k(\nabla f(x_k) - g_k)^T d_k.\end{aligned}$$

Proof. Proof. Consider arbitrary $k \in \mathbb{N}$ in any run under Assumption 5. As in the proof of Lemma 9, from (21)–(22) and the supposition about $\{\beta_k\}$, one finds $\alpha_k \in (0,1]$. Hence, with (19), one has

$$\begin{aligned}\phi(x_k,\tau_k) - \phi(x_k+\alpha_k d_k,\tau_k) &\geq -\alpha_k(\tau_k\nabla f(x_k)^T d_k - \|c_k\|_2 + \|c_k + J_k d_k\|_2) - \tfrac{1}{2}(\tau_k L + \Gamma)\alpha_k^2\|d_k\|_2^2 \\ &= \alpha_k\Delta l(x_k,\tau_k,g_k,d_k) - \tfrac{1}{2}(\tau_k L+\Gamma)\alpha_k^2\|d_k\|_2^2 - \alpha_k\tau_k(\nabla f(x_k)-g_k)^T d_k.\end{aligned}$$

Now following the same arguments as in the proof of Lemma 9, it follows that $-\frac{1}{2}(\tau_k L + \Gamma)\alpha_k^2\|d_k\|_2^2 \geq -(1-\eta)\alpha_k\beta_k\Delta l(x_k,\tau_k,g_k,d_k)$, which combined with the above yields the desired conclusion. □ □

Our next result yields a bound on the final term in the conclusion of Lemma 13.

Lemma 14. *For any run under Assumption 5, there exists $\kappa_\beta \in \mathbb{R}_{>0}$ such that*

$$\alpha_k\tau_k(\nabla f(x_k) - g_k)^T d_k \leq \beta_k\tau_{k-1}\kappa_\beta \quad \textit{for all} \quad k \in \mathbb{N}.$$

Proof. Proof. The existence of $\kappa_d \in \mathbb{R}_{>0}$ such that, in any run under Assumption 5, one finds $\|d_k\|_2 \leq \kappa_d$ for all $k \in \mathbb{N}$ follows from Assumption 5, the fact that $\|d_k\|_2^2 = \|v_k\|_2^2 + \|u_k\|_2^2$ for all $k \in \mathbb{N}$, Lemma 7, Lemma 1, and Assumption 1. Now consider arbitrary $k \in \mathbb{N}$ in any run under Assumption 5. If $(\nabla f(x_k) - g_k)^T d_k < 0$, then the desired conclusion follows trivially (for any $\kappa_\beta \in \mathbb{R}_{>0}$). Hence, let us proceed under the assumption that $(\nabla f(x_k) - g_k)^T d_k \geq 0$. It follows from (22), the facts that $0 \leq \tau_k \leq \tau_{k-1}$, $\tau_k \leq \tau_{-1}$, $\xi_k \leq \xi_{-1}$, and $\beta_k \leq 1$ for all $k \in \mathbb{N}$, the Cauchy-Schwarz inequality, and Assumption 5 that

$$\begin{aligned}\alpha_k \tau_k (\nabla f(x_k) - g_k)^T d_k &\leq \left(\frac{2(1-\eta)\beta_k \xi_k}{\tau_k L + \Gamma} \max\{\tau_k, 1\} + \theta\beta_k^2 \right) \tau_k \|\nabla f(x_k) - g_k\|_2 \|d_k\|_2 \\ &\leq \left(\frac{2(1-\eta)\beta_k \xi_{-1}}{\Gamma} \max\{\tau_{-1}, 1\} + \theta\beta_k \right) \tau_{k-1} \sqrt{M} \kappa_d,\end{aligned}$$

from which the desired conclusion follows with $\kappa_\beta := \left(\frac{2(1-\eta)\xi_{-1}}{\Gamma} \max\{\tau_{-1}, 1\} + \theta \right) \sqrt{M} \kappa_d$. □ □

Our third result in this subsection offers a formula for a positive lower bound on the stepsize that is applicable at points that are not stationary for the constraint infeasibility measure. For this lemma and its subsequent consequences, we define for arbitrary $\gamma \in \mathbb{R}_{>0}$ the subset

$$\mathcal{X}_\gamma := \{x \in \mathbb{R}^n : \|J(x)^T c(x)\|_2 \geq \gamma\}. \tag{32}$$

Lemma 15. *There exists $\underline{\alpha} \in \mathbb{R}_{>0}$ such that $\alpha_k \geq \underline{\alpha}\beta_k$ for each $k \in \mathbb{N}$ such that $\|u_k\|_2^2 < \chi_k \|v_k\|_2^2$. On the other hand, for each $\gamma \in \mathbb{R}_{>0}$, there exists $\epsilon_\gamma \in \mathbb{R}_{>0}$ (proportional to γ^2) such that*

$$x_k \in \mathcal{X}_\gamma \quad \textit{implies} \quad \alpha_k \geq \min\{\epsilon_\gamma \beta_k, \epsilon_\gamma \beta_k \tau_k + \theta\beta_k^2\} \quad \textit{whenever} \quad \|u_k\|_2^2 \geq \chi_k \|v_k\|_2^2.$$

Proof. Proof. Define $\mathcal{K}_\gamma := \{k \in \mathbb{N} : x_k \in \mathcal{X}_\gamma\}$. By Lemma 1, it follows that $\|v_k\|_2 \geq \underline{\omega} \|J_k^T c_k\|_2^2 \geq \underline{\omega}\gamma^2$ for all $k \in \mathcal{K}_\gamma$. Consequently, by Lemma 7, it follows that

$$\|u_k\|_2 \leq \frac{\kappa_u}{\underline{\omega}\gamma^2} \|v_k\|_2 \quad \text{for all} \quad k \in \mathcal{K}_\gamma. \tag{33}$$

It follows from (22) that $\alpha_k \geq 2(1-\eta)\beta_k \xi_k / (\tau_k L + \Gamma)$ whenever $\|u_k\|_2^2 < \chi_k \|v_k\|_2^2$. Otherwise, whenever $\|u_k\|_2^2 \geq \chi_k \|v_k\|_2^2$, it follows using the arguments in Lemma 9 and (22) that

$$\alpha_k = \min\left\{ \frac{2(1-\eta)\beta_k \Delta l(x_k, \tau_k, g_k, d_k)}{(\tau_k L + \Gamma)\|d_k\|_2^2}, \frac{2(1-\eta)\beta_k \xi_k \tau_k}{\tau_k L + \Gamma} + \theta\beta_k^2, 1 \right\},$$

which along with (14), Lemma 1, (2), and (33) imply that

$$\begin{aligned}\alpha_k &\geq \min\left\{ \frac{2(1-\eta)\beta_k \sigma(\|c_k\|_2 - \|c_k + J_k v_k\|_2)}{(\tau_k L + \Gamma)(\|u_k\|_2^2 + \|v_k\|_2^2)}, \frac{2(1-\eta)\beta_k \xi_k \tau_k}{\tau_k L + \Gamma} + \theta\beta_k^2, 1 \right\} \\ &\geq \min\left\{ \frac{2(1-\eta)\beta_k \sigma \kappa_v \|J_k^T c_k\|^2}{(\tau_k L + \Gamma)(\frac{\kappa_u^2}{\underline{\omega}^2\gamma^4} + 1)\omega^2 \|c_k\| \|J_k^T c_k\|^2}, \frac{2(1-\eta)\beta_k \xi_k \tau_k}{\tau_k L + \Gamma} + \theta\beta_k^2, 1 \right\} \\ &\geq \min\left\{ \frac{2(1-\eta)\beta_k \sigma \kappa_v \underline{\omega}^2 \gamma^4}{(\tau_k L + \Gamma)\kappa_c \omega^2 (\kappa_u^2 + \underline{\omega}^2\gamma^4)}, \frac{2(1-\eta)\beta_k \xi_k \tau_k}{\tau_k L + \Gamma} + \theta\beta_k^2, 1 \right\}.\end{aligned}$$

Combining the cases above with Lemma 3 yields the desired conclusion. □ □

We now prove our main theorem of this subsection, where $\mathbb{E}_{\tau,\text{zero}}[\,\cdot\,] = \mathbb{E}[\,\cdot \mid \text{Assumption 5 holds}]$.

Theorem 3. *Suppose that Assumption 5 holds and that $\beta_k = \beta$ for all $k \in \mathbb{N}$, where $\beta \in (0,1)$ is chosen such that (28) holds for all $k \in \mathbb{N}$. Then, $\liminf_{k \to \infty} \|J_k^T c_k\|_2 = 0$.*

Proof. Proof. To derive a contradiction, suppose that there exists $k_\gamma \in \mathbb{N}$ and $\gamma \in \mathbb{R}_{>0}$ such that $x_k \in \mathcal{X}_\gamma$ for all $k \geq k_\gamma$. By Lemmas 13–15, (2), (14), the fact that $\beta \in (0,1)$, Lemma 1, and Assumption 1, it follows that there exists $\underline{\epsilon}_\gamma \in \mathbb{R}_{>0}$ such that

$$\|c_k\|_2 - \|c(x_k + \alpha_k d_k)\|_2$$

$$
\begin{aligned}
&\geq \alpha_k(1-(1-\eta)\beta)\Delta l(x_k,\tau_k,g_k,d_k) - \tau_k(f_k - f(x_k+\alpha_k d_k)) - \alpha_k\tau_k(\nabla f(x_k) - g_k)^T d_k \\
&\geq \underline{\epsilon}_\gamma \beta\eta\sigma(\|c_k\|_2 - \|c_k + J_k v_k\|_2) - \tau_{k-1}(f_{\sup} - f_{\inf}) - \beta\tau_{k-1}\kappa_\beta \\
&\geq \underline{\epsilon}_\gamma \beta\eta\sigma\kappa_v\kappa_c^{-1}\|J_k^T c_k\|_2^2 - \tau_{k-1}(f_{\sup} - f_{\inf} + \beta\kappa_\beta) \ \text{ for all } \ k \geq k_\gamma. \qquad (34)
\end{aligned}
$$

Since $\|J_k^T c_k\|_2 \geq \gamma$ for all $k \geq k_\gamma$ and $\{\tau_k\} \searrow 0$ under Assumption 5, it follows that there exists $k_\tau \geq k_\gamma$ such that $\tau_{k-1}(f_{\sup} - f_{\inf} + \beta\kappa_\beta) \leq \frac{1}{2}\underline{\epsilon}_\gamma\beta\eta\sigma\kappa_v\kappa_c^{-1}\|J_k^T c_k\|_2^2$ for all $k \geq k_\tau$. Hence, summing (34) for $j \in \{k_\tau, \dots, k_\tau + k\}$, it follows with (2) that

$$
\kappa_c \geq \|c_{k_\tau}\|_2 - \|c_{k_\tau+k+1}\|_2 \geq \frac{1}{2}\underline{\epsilon}_\gamma\beta\eta\sigma\kappa_v\kappa_c^{-1}\sum_{j=k_\tau}^{k_\tau+k}\|J_j^T c_j\|_2.
$$

It follows from this fact that $\{J_k^T c_k\}_{k\geq k_\tau} \to 0$, yielding a contradiction, as desired. □ □

4.3 Constant, Insufficiently Small Merit Parameter

Our goal now is to consider the event that the algorithm generates a merit parameter sequence that eventually remains constant, but at a value that is too large in the sense that the conditions of Assumption 4 do not hold. Such an event for the algorithm in [1] is addressed in Proposition 3.16 in that article, where under a reasonable assumption (paralleling (37a), which we discuss later on) it is shown that, in a given run of the algorithm, the probability is zero of the merit parameter settling on too large of a value. The same can be said of our algorithm, as discussed in this subsection. That said, this does not address what might be the total probability, over all runs of the algorithm, of the event that the merit parameter remains too large. We discuss in this section what it means to assume that this total probability is equal to zero, and how such an assumption may be justified in general.

For our purposes in this section, we make some mild simplifications. First, as shown in Lemmas 2 and 3, each of the sequences $\{\chi_k\}$, $\{\zeta_k\}$, and $\{\xi_k\}$ has a nontrivial, uniform bound that holds over any run of the algorithm. Hence, for simplicity, we shall assume that the initial values of these sequences are chosen such that they are constant over $k \in \mathbb{N}$. (Our discussions in this subsection can be generalized to situations when this is not the case; the conversation merely becomes more cumbersome, which we have chosen to avoid.) Second, it follows from properties of the deterministic instance of our algorithm (recall Theorem 1) that if a subsequence of $\{\tau_k^{\text{trial,true}}\}$ converges to zero, then a subsequence of the sequence of minimum singular values of the constraint Jacobians $\{J_k\}$ vanishes as well. Hence, we shall consider in this subsection events in which there exists $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ such that $\tau_k^{\text{trial,true}} \geq \tau_{\min}^{\text{trial,true}}$ for all $k \in \mathbb{N}$ in any run of the algorithm. (We will remark on the consequences of this assumption further in Section 4.4.) It follows from this and (12) that if the cardinality of the set of iteration indices $\{k \in \mathbb{N} : \tau_k < \tau_{k-1}\}$ ever exceeds

$$
\bar{s}(\tau_{\min}^{\text{trial,true}}) := \left\lceil \frac{\log(\tau_{\min}^{\text{trial,true}}/\tau_{-1})}{\log(1-\epsilon_\tau)} \right\rceil \in \mathbb{N}, \qquad (35)
$$

then for all subsequent $k \in \mathbb{N}$ one has $\tau_{k-1} \leq \tau_{\min}^{\text{trial,true}} \leq \tau_k^{\text{trial,true}}$. This property of $\bar{s}(\tau_{\min}^{\text{trial,true}})$ is relevant in our event of interest for this subsection, which we now define.

Definition 1. The event $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ for some $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ occurs in a run if and only if $\tau_k^{\text{trial,true}} \geq \tau_{\min}^{\text{trial,true}}$ for all $k \in \mathbb{N}$ and there exists an infinite index set $\mathcal{K} \subseteq \mathbb{N}$ such that

$$
\tau_k^{\text{trial,true}} < \tau_{k-1} \ \text{ for all } \ k \in \mathcal{K}. \qquad (36)
$$

Considering a given run of our algorithm in which it is presumed that $\tau_k^{\text{trial,true}} \geq \tau_{\min}^{\text{trial,true}}$ for some $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ and for all $k \in \mathbb{N}$, one has under a reasonable assumption (specifically, that (37a) in the lemma below holds for all $k \in \mathbb{N}$) that the probability is zero that $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ occurs. We prove this now using the same argument as in the proof of [1, Proposition 3.16]. For this, we require the following lemma, proved here for our setting, which is slightly different than for the algorithm in [1] (due to the slightly different formula for setting the merit parameter).

Lemma 16. *For any $k \in \mathbb{N}$ in any run of the algorithm, it follows for any $p \in (0,1]$ that*

$$\mathbb{P}_k[g_k^T d_k + u_k^T H_k u_k \geq \nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}}] \geq p \tag{37a}$$

$$\textit{implies} \;\; \mathbb{P}_k[\tau_k < \tau_{k-1} | \tau_k^{\text{trial,true}} < \tau_{k-1}] \geq p. \tag{37b}$$

Proof. Proof. Consider any $k \in \mathbb{N}$ in any run of the algorithm such that $\tau_k^{\text{trial,true}} < \tau_{k-1} \in \mathbb{R}_{>0}$. Then, it follows from (11) that $\tau_k^{\text{trial,true}} < \infty$, $\nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}} > 0$, and

$$\tau_k^{\text{trial,true}} = \frac{(1-\sigma)(\|c_k\|_2 - \|c_k + J_k d_k^{\text{true}}\|_2)}{\nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}}} < \tau_{k-1},$$

from which it follows that

$$(1-\sigma)(\|c_k\|_2 - \|c_k + J_k d_k^{\text{true}}\|_2) < (\nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}})\tau_{k-1}. \tag{38}$$

If, in addition, a realization of g_k yields

$$g_k^T d_k + u_k^T H_k u_k \geq \nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}}, \tag{39}$$

then it follows from (38) and the fact that $J_k d_k^{\text{true}} = J_k d_k$ that

$$(1-\sigma)(\|c_k\|_2 - \|c_k + J_k d_k\|_2) < (g_k^T d_k + u_k^T H_k u_k)\tau_{k-1}.$$

It follows from this inequality and Lemma 1 that $g_k^T d_k + u_k^T H_k u_k > 0$, and with (12) it holds that

$$\tau_k \leq \tau_k^{\text{trial}} = \frac{(1-\sigma)(\|c_k\|_2 - \|c_k + J_k d_k\|_2)}{g_k^T d_k + u_k^T H_k u_k} < \tau_{k-1}.$$

Hence, conditioned on the event that $\tau_k^{\text{trial,true}} < \tau_{k-1}$, one finds that (39) implies that $\tau_k < \tau_{k-1}$. Therefore, under the conditions of the lemma and the fact that, conditioned on the events leading up to iteration number k one has that both $\tau_k^{\text{trial,true}}$ and τ_{k-1} are deterministic, it follows that

$$\begin{aligned}
&\mathbb{P}_k[\tau_k < \tau_{k-1} | \tau_k^{\text{trial,true}} < \tau_{k-1}] \\
&\geq \mathbb{P}_k[g_k^T d_k + u_k^T H_k u_k \geq \nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}} | \tau_k^{\text{trial,true}} < \tau_{k-1}] \\
&= \mathbb{P}_k[g_k^T d_k + u_k^T H_k u_k \geq \nabla f(x_k)^T d_k^{\text{true}} + (u_k^{\text{true}})^T H_k u_k^{\text{true}}] \geq p,
\end{aligned}$$

as desired. □ □

We can now prove the following result for our algorithm. (We remark that [1] also discusses an illustrative example in which (37a) holds for all $k \in \mathbb{N}$; see Example 3.17 in that article.)

Proposition 1. *If, in a given run of our algorithm, there exist $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ and $p \in (0,1]$ such that $\tau_k^{\text{trial,true}} \geq \tau_{\min}^{\text{trial,true}}$ and (37a) hold for all $k \in \mathbb{N}$, then the probability is zero that the event $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ occurs in the run.*

Proof. Proof. Under the conditions of the proposition, the conclusion follows from Lemma 16 using the same argument as in the proof of [1, Proposition 3.16]. □ □

The analysis above shows that if $\{\tau_k^{\text{trial,true}}\}$ is bounded below uniformly by a positive real number, then the probability is zero that $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ occurs in a given run. From this property, it follows under this condition that the probability is zero that $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ occurs in a countable number of runs. However, this analysis does not address what may be the total probability, over all possible runs of the algorithm, that $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ may occur. Determining this total probability would require careful consideration of the distributions of the stochastic gradient estimates at each possible iterate that go beyond our scope. To assume that the total probability of this event is zero amounts to making the following assumption, for which we define the (random) index sets

$$\begin{aligned} \mathcal{K}_\tau^{\text{should}} &:= \{k \in \mathbb{N} : \tau_k^{\text{trial,true}} < \tau_{k-1}\} \\ \text{and } \ \mathcal{K}_\tau^{\text{actual}} &:= \{k \in \mathbb{N} : \tau_k^{\text{trial,true}} < \tau_{k-1} \text{ and } \tau_k < \tau_{k-1}\} \subseteq \mathcal{K}_\tau^{\text{should}}. \end{aligned}$$

Assumption 6. *Given $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ and $\bar{s} := \bar{s}(\tau_{\min}^{\text{trial,true}})$ defined in* (35)*, one has that*

$$\lim_{s\to\infty} \mathbb{P}\left[\left(|\mathcal{K}_\tau^{should}| \geq s\right) \wedge \left(|\mathcal{K}_\tau^{actual}| < \bar{s}\right)\right] = 0. \tag{40}$$

(*Here, as earlier in the paper, $\mathbb{P}[\cdot]$ denotes total probability over all runs of the algorithm.*)

Let us now present an argument as to how Assumption 6 may be justified. The index set $\mathcal{K}_\tau^{\text{should}}$ denotes iterations in which computing a search direction with the *true* gradient of the objective would lead to a merit parameter decrease, whereas the subset $\mathcal{K}_\tau^{\text{actual}}$ denotes such iterations in which computing a search direction with the stochastic gradient does indeed lead to a merit parameter decrease. Put another way, the former set represents iterations in which the algorithm *should* decrease the merit parameter, whereas the latter set represents iterations in which it should *and actually does* decrease it. Assumption 6 essentially states that as $s \to \infty$, the probability vanishes that the algorithm encounters $s \geq \bar{s}$ iterations in which the merit parameter *should* be decreased, yet it is *actually* decreased fewer than $\bar{s}$ times. This assumption may be seen as strong since it makes this statement in terms of total probability over all runs of the algorithm, rather than with respect to the state of the algorithm in a given run. To argue how such an assumption may be justified, let us use the following consequence of Chernoff's bound for a specific example.

Proposition 2. *Given $s \in \mathbb{N}$, suppose that $\{Z_j\}_{j=0}^s$ is a sequence of independent Bernoulli random variables with $sp = \sum_{j=0}^s \mathbb{P}_{Z_j}[Z_j = 1]$ for some $p \in (0,1]$. Then, for any $\bar{s} \in \mathbb{N}$ with $\bar{s} \leq sp$,*

$$\mathbb{P}_{Z_0,\dots,Z_s}\left[\sum_{j=0}^s Z_j \leq \bar{s}\right] \leq \exp\left(\frac{-(sp-\bar{s})^2}{2sp}\right). \tag{41}$$

Proof. Proof. By the multiplicative Chernoff bound with factor $\delta = (sp - \bar{s})/(sp) \in [0,1]$, it follows that

$$\mathbb{P}_{Z_0,\dots,Z_s}\left[\sum_{j=0}^s Z_j \leq \bar{s}\right] = \mathbb{P}_{Z_0,\dots,Z_s}\left[\sum_{j=0}^s Z_j \leq (1-\delta)sp\right] \leq \exp\left(\frac{-\delta^2 sp}{2}\right) = \exp\left(\frac{-(sp-\bar{s})^2}{2sp}\right),$$

which is the desired conclusion. □ □

For casting Lemma 2 in our context, let $\{Z_j\}_{j=0}^s$ be a sequence where, for all $j \in [s]$, one has $\mathbb{P}_{Z_j}[Z_j = 1]$ equal to the probability that the merit parameter is *actually* decreased in the jth iteration in which it *should* be decreased; in other words, $Z_j = 1$ if and only if $\tau_k < \tau_{k-1}$, where $k \in \mathbb{N}$ corresponds to the jth iteration in which $\tau_k^{\text{trial,true}} < \tau_{k-1}$. Lemma 2 shows that if these random variables exhibit the behavior of independent Bernoulli random variables, then if the probability of a successful decrease on a decrease opportunity is bounded below by a positive number (i.e., $\mathbb{P}[\tau_k < \tau_{k-1} | \tau_k^{\text{trial,true}} < \tau_{k-1}] \geq p$, a total probability extension of (37b)), then one finds that if the number of decrease opportunities s is large enough such that $sp \geq \bar{s}$, then the probability that the number of actual decreases stays below $\bar{s}$ vanishes exponentially as $s \to \infty$. The challenge of employing this result to the actual behavior of our algorithm is that the elements of $\{Z_j\}_{j=0}^{s}$ defined in this manner are not independent, since in each run the probability of a merit parameter decrease in each subsequent iteration depends on whether each given decrease opportunity is successful.

Overall, while our analysis here has not provided *a priori* assumptions about the distributions of the stochastic gradients that guarantee that $E_{\tau,\text{big}}(\tau_{\min}^{\text{trial,true}})$ for a given $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$ occurs with probability zero—which, in general, may not be possible due to the stochastic nature of our algorithm and our loose assumptions about the nonlinear and potentially nonconvex problem (1)—we have shown that in a given run of the algorithm such an event occurs with probability zero, and discussed what it may mean to assume that the total probability of the event is zero.

4.4 Complementary Events

Our analyses in Sections 4.1, 4.2, and 4.3 do not cover all possible events. Ignoring events in which the stochastic gradients are biased and/or have unbounded variance, the events that complement $E_{\tau,\text{low}}$, $E_{\tau,\text{zero}}$, and $E_{\tau,\text{big}}$ are the following:

- $E_{\tau,\text{zero,bad}}$: $\{\tau_k\} \searrow 0$ and for all $M \in \mathbb{R}_{>0}$ there exists $k \in \mathbb{N}$ such that $\|g_k - \nabla f(x_k)\|_2^2 > M$;
- $E_{\tau,\text{big,bad}}$: $\{\tau_k^{\text{trial,true}}\} \searrow 0$ and there exists $\tau_{\text{big}} \in \mathbb{R}_{>0}$ such that $\tau_k = \tau_{\text{big}}$ for all $k \in \mathbb{N}$.

The event $E_{\tau,\text{zero,bad}}$ represents cases in which the merit parameter vanishes while the stochastic gradient estimates do not remain in a bounded set. The difficulty of proving a guarantee for this setting can be seen as follows. If the merit parameter vanishes, then this is an indication that less emphasis should be placed on the objective over the course of the optimization process, which may indicate that the constraints are infeasible or degenerate. However, if a subsequence of stochastic gradient estimates diverges at the same time, then each large (in norm) stochastic gradient estimate may suggest that a significant amount of progress can be made in reducing the objective function, despite the merit parameter having reached a small value (since it is vanishing). This disrupts the balance that the merit parameter attempts to negotiate between the objective and the constraint violation terms in the merit function. Our analysis of the event $E_{\tau,\text{zero}}$ in Section 4.2 shows that if the stochastic gradient estimates remain bounded, then the algorithm can effectively transition to solving the deterministic problem of minimizing constraint violation. However, it remains an open question whether it is possible to obtain a similar guarantee if/when a subsequence of stochastic gradient estimates diverges. Ultimately, one can argue that scenarios of unbounded noise, such as described here, might only be of theoretical interest rather than real, practical interest. For instance, if f is defined by a (large) finite sum of component functions whose gradients (evaluated at points in a set containing the iterates) are always contained in a ball of uniform radius about the gradient of f—a common scenario in practice—then $E_{\tau,\text{zero,bad}}$ cannot occur.

Now consider the event $E_{\tau,\text{big,bad}}$. We have shown in Section 4.3 that if $\{\tau_k^{\text{trial,true}}\}$ is bounded below by $\tau_{\min}^{\text{trial,true}} \in \mathbb{R}_{>0}$, Assumption 6 holds, and the sequences $\{\chi_k\}$, $\{\zeta_k\}$, and $\{\xi_k\}$ are constant, then $E_{\tau,\text{big}}$ occurs with probability zero. However, this does not account for the fact that over different realizations of the algorithm the lower bound for $\{\tau_k^{\text{trial,true}}\}$ may be arbitrarily small. Nonetheless, we contend that $E_{\tau,\text{big,bad}}$ can be ignored for practical purposes since the adverse effect that it may have on the algorithm is observable. In particular, if the merit parameter remains fixed at a value that is too large, then the worst that may occur is that $\{\|J_k^T c_k\|_2\}$ does not vanish. A practical implementation of the algorithm would monitor this quantity in any case (since, by Corollary 1, even in $E_{\tau,\text{low}}$ one only knows that the limit inferior of the expectation of $\{\|J_k^T c_k\|_2\}$ vanishes) and reduce the merit parameter if progress toward reducing constraint violation is inadequate. Hence, $E_{\tau,\text{big,bad}}$ (and $E_{\tau,\text{big}}$ for that matter) is an event that at most suggests practical measures of the algorithm that should be employed for $E_{\tau,\text{low}}$ in any case.

5 Numerical Experiments

The goal of our numerical experiments is to compare the empirical performance of our proposed stochastic SQP method (Algorithm 1) against some alternative approaches on problems from a couple of test set collections. We implemented our algorithm in Matlab. Our code is publicly available.[1] We first consider equality constrained problems from the CUTEst collection [7], then consider two types of constrained logistic regression problems with datasets from the LIBSVM collection [3]. We compare the performance of our method versus a stochastic subgradient algorithm employed to minimize the exact penalty function (10) and, in one set of our logistic regression experiments where it is applicable, versus a stochastic projected gradient method. These algorithms were chosen since, like our method, they operate in the highly stochastic regime. We do not compare against the aforementioned method from [14] since, as previously mentioned, that approach may refine stochastic gradient estimates during each iteration as needed by a line search. Hence, that method offers different types of convergence guarantees and is not applicable in our regime of interest.

In all of our experiments, results are given in terms of feasibility and stationarity errors at the *best* iterate, which is determined as follows. If, for a given problem instance, an algorithm produced an iterate that was sufficiently feasible in the sense that $\|c_k\|_\infty \leq 10^{-6}\max\{1, \|c_0\|_\infty\}$ for some $k \in \mathbb{N}$, then, with the largest $k \in \mathbb{N}$ satisfying this condition, the feasibility error was reported as $\|c_k\|_\infty$ and the stationarity error was reported as $\|\nabla f(x_k) + J_k^T y_k\|_\infty$, where y_k was computed as a least-squares multiplier using the true gradient $\nabla f(x_k)$ and J_k. (The multiplier y_k and corresponding stationarity error are not needed by our algorithm; they are computed merely so that we could record the error for our experimental results.) If, for a given problem instance, an algorithm did not produce a sufficiently feasible iterate, then the feasibility and stationarity errors were computed in the same manner at the least infeasible iterate (with respect to the measure of infeasibility $\|\cdot\|_\infty$).

5.1 Implementation Details

For all methods, Lipschitz constant estimates for the objective gradient and constraint Jacobian—playing the roles of L and Γ, respectively—were computed using differences of gradients near the initial point. Once these values were computed, they were kept constant for all subsequent iterations. This procedure was performed in such a way that, for each problem instance, all algorithms used the same values for these estimates.

As mentioned in Section 3, there are various extensions of our stepsize selection scheme with which one can prove, with appropriate modifications to our analysis, comparable convergence guarantees as are offered by our algorithm. We included one such extension in our software implementation for our experiments. In particular, in addition to $\alpha_k^{\rm suff}$ in (21), one can directly consider the upper bound in (19) with the gradient $\nabla f(x_k)$ replaced by its estimate g_k, i.e.,

$$\begin{aligned}&\alpha\tau_k g_k^T d_k + |1-\alpha|\|c_k\|_2 - \|c_k\|_2 + \alpha\|c_k + J_k d_k\|_2 + \tfrac{1}{2}(\tau_k L + \Gamma)\alpha^2\|d_k\|_2^2 \\ = \; & -\alpha\Delta l(x_k, \tau_k, g_k, d_k) + |1-\alpha|\|c_k\|_2 - (1-\alpha)\|c_k\|_2 + \tfrac{1}{2}(\tau_k L + \Gamma)\alpha^2\|d_k\|_2^2,\end{aligned}$$

and consider the stepsize that minimizes this as a function of α (with scale factor β_k), namely,

$$\alpha_k^{\rm min} := \max\left\{\min\left\{\frac{\beta_k\Delta l(x_k, \tau_k, g_k, d_k)}{(\tau_k L + \Gamma)\|d_k\|_2^2}, 1\right\}, \frac{\beta_k\Delta l(x_k, \tau_k, g_k, d_k) - 2\|c_k\|_2}{(\tau_k L + \Gamma)\|d_k\|_2^2}\right\}. \tag{42}$$

(Such a value is used in [1].) The algorithm can then set a trial stepsize as any satisfying

$$\alpha_k^{\rm trial} \in [\min\{\alpha_k^{\rm suff}, \alpha_k^{\rm min}\}, \max\{\alpha_k^{\rm suff}, \alpha_k^{\rm min}\}] \tag{43}$$

and set α_k as the projection of this value, rather than $\alpha_k^{\rm suff}$, for all $k \in \mathbb{N}$. (The projection interval in (22) should be modified, specifically with each instance of $2(1-\eta)$ replaced by $\min\{2(1-\eta), 1\}$, to account for the fact that the lower value in (43) may be smaller than $\alpha_k^{\rm suff}$. A similar modification is needed in the analysis, specifically in the requirements for $\{\beta_k\}$ in Lemma 9.)

One can also consider rules that allow even larger stepsizes to be taken. For example, rather than consider the upper bound offered by the last expression in (19), one can consider any stepsize that ensures that the

[1] `https://github.com/frankecurtis/StochasticSQP`

penultimate expression in (19) is less than or equal to the right-hand side of (20) with $\nabla f(x_k)$ replaced by g_k. Such a value can be found with a one-dimensional search over α with negligible computational cost. Our analysis can be extended to account for this option as well. However, for our experimental purposes here, we do not consider such an approach.

For our stochastic SQP method, we set $H_k \leftarrow I$ and $\alpha_k^{\rm trial} \leftarrow \max\{\alpha_k^{\rm suff}, \alpha_k^{\min}\}$ for all $k \in \mathbb{N}$. Other parameters were set as $\tau_{-1} \leftarrow 1$, $\chi_{-1} \leftarrow 10^{-3}$, $\zeta_{-1} \leftarrow 10^3$, $\xi_{-1} \leftarrow 1$, $\omega \leftarrow 10^2$, $\epsilon_v \leftarrow 1$, $\sigma \leftarrow 1/2$, $\epsilon_\tau \leftarrow 10^{-2}$, $\epsilon_\chi \leftarrow 10^{-2}$, $\epsilon_\zeta \leftarrow 10^{-2}$, $\epsilon_\xi \leftarrow 10^{-2}$, $\eta \leftarrow 1/2$, and $\theta \leftarrow 10^4$. For the stochastic subgradient method, the merit parameter value and stepsize were tuned for each problem instance, and for the stochastic projected gradient method, the stepsize was tuned for each problem instance; details are given in the following subsections. In all experiments, both the stochastic subgradient and stochastic projected gradient method were given many more iterations to find each of their best iterates for a problem instance; this is reasonable since the search direction computation for our method is more expensive than for the other methods. Again, further details are given below.

5.2 CUTEst problems

In our first set of experiments, we consider equality constrained problems from the CUTEst collection. Specifically, of the 136 such problems in the collection, we selected those for which *(i)* f is not a constant function, and *(ii)* $n+m+1 \leq 1000$. This selection resulted in a set of 67 problems. In order to consider the context in which the LICQ does not hold, for each problem we duplicated the last constraint. (This does not affect the feasible region nor the set of stationary points, but ensures that the problem instances are degenerate.) Each problem comes with an initial point, which we used in our experiments. To make each problem stochastic, we added noise to each gradient computation. Specifically, for each run of an algorithm, we fixed a *noise level* as $\epsilon_N \in \{10^{-8}, 10^{-4}, 10^{-2}, 10^{-1}\}$, and in each iteration set the stochastic gradient estimate as $g_k \leftarrow \mathcal{N}(\nabla f(x_k), \epsilon_N I)$. For each problem and noise level, we ran 10 instances with different random seeds. This led to a total of 670 runs of each algorithm for each noise level.

We set a budget of 1000 iterations for our stochastic SQP algorithm and a more generous budget of 10000 iterations for the stochastic subgradient method. We followed the same strategy as in [1] to tune the merit parameter τ for the stochastic subgradient method, but also tuned the stepsizes through the sequence $\{\beta_k\}$. Specifically, for each problem instance, we ran the stochastic subgradient method for 11 different values of τ and 4 different values of β, namely, $\tau \in \{10^{-10}, 10^{-9}, \dots, 10^0\}$ and $\beta \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^0\}$, set the stepsize as $\frac{\beta\tau}{\tau L + \Gamma}$, and selected the combination of τ and β for that problem instance that led to the best iterate overall. (We found through this process that the selected (τ, β) pairs were relatively evenly distributed over their ranges, meaning that this extensive tuning effort was useful to obtain better results for the stochastic subgradient method.) For our stochastic SQP method, we set $\beta_k \leftarrow 1$ for all $k \in \mathbb{N}$. Overall, between the additional iterations allowed in each run of the stochastic subgradient method, the different merit parameter values tested, and the different stepsizes tested, the stochastic subgradient method was given 440 times the number of iterations that were given to our stochastic SQP method for each problem.

The results of this experiment are reported in the form of box plots in Figure 1. One finds that the best iterates from our stochastic SQP algorithm generally correspond to much lower feasibility and stationarity errors for all noise levels. The stationarity errors for our method degrade as the noise level increases, but this is not surprising since these experiments are run with $\{\beta_k\}$ being a constant sequence. It is interesting, however, that our algorithm typically finds iterates that are sufficiently feasible, even for relatively high noise levels. This shows that our approach handles the deterministic constraints well despite the stochasticity of the objective gradient estimates. Finally, we remark that for these experiments our algorithm found $\tau_{k-1} \leq \tau_k^{\rm trial,true}$ to hold in roughly 98% of all iterations for all runs (across all noise levels), and found this inequality to hold in the last 50 iterations in 100% of all runs. This provides evidence for our claim that the merit parameter not reaching a sufficiently small value is not an issue of practical concern.

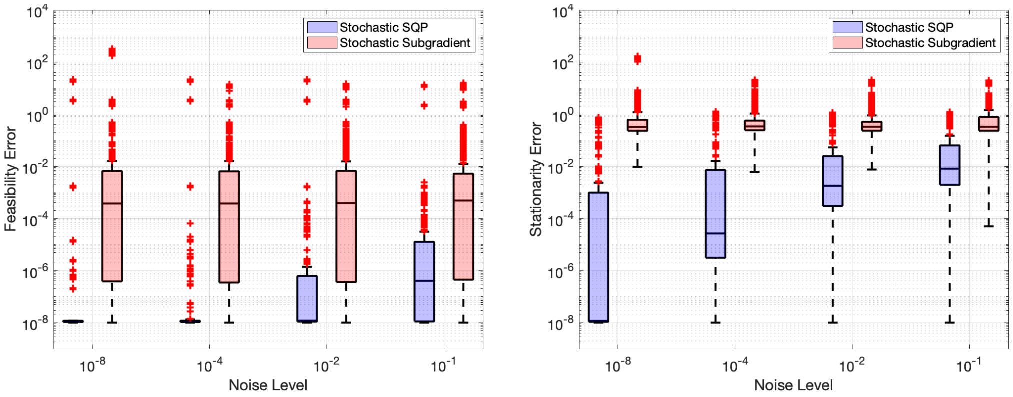

Figure 1: Box plots for feasibility errors (left) and stationarity errors (right) when our stochastic SQP method and a stochastic subgradient method are employed to solve equality constrained problems from the CUTEst collection.

5.3 Constrained Logistic Regression

In our next sets of experiments, we consider equality constrained logistic regression problems of the form

$$\min_{x\in\mathbb{R}^n}\ f(x) = \frac{1}{N}\sum_{i=1}^{N}\log\left(1+e^{-y_i(X_i^Tx)}\right)\quad \text{s.t.}\quad Ax=b,\quad \|x\|_2^2=1, \tag{44}$$

where $X \in \mathbb{R}^{n\times N}$ contains feature data for N data points (with X_i representing the ith column of X), $y \in \{-1,1\}^N$ contains corresponding label data, $A \in \mathbb{R}^{(m+1)\times n}$ and $b \in \mathbb{R}^{m+1}$. For instances of (X, y), we consider 11 binary classification datasets from the LIBSVM collection [3]; specifically, we consider all of the datasets for which $12 \le n \le 1000$ and $256 \le N \le 100000$. (For datasets with multiple versions, e.g., the $\{\texttt{a1a},\dots,\texttt{a9a}\}$ datasets, we consider only the largest version.) The names of the datasets that we used and their sizes are given in Table 1. For the linear constraints, we generated random A and b for each problem. Specifically, the first $m = 10$ rows of A and first m entries in b were set as random values with each entry being drawn from a standard normal distribution. Then, to ensure that the LICQ was not satisfied (at any algorithm iterate), we duplicated the last constraint, making $m + 1$ linear constraints overall. For all problems and algorithms, the initial iterate was set to the vector of all ones of appropriate dimension.

Table 1: Names and sizes of datasets. (Source: [3].)

dataset	**dimension** (n)	**datapoints** (N)
a9a	123	$32,561$
australian	14	690
heart	13	270
ijcnn1	22	$49,990$
ionosphere	34	351
madelon	500	$2,000$
mushrooms	112	$8,124$
phising	68	$11,055$
sonar	60	208
splice	60	$1,000$
w8a	300	$49,749$

For one set of experiments, we consider problems of the form (44) except without the norm constraint. For this set of experiments, the performance of all three algorithms—stochastic SQP, subgradient, and projected gradient—are compared. For each dataset, we considered two noise levels, where the level is dictated by

Table 2: Average feasibility and stationarity errors, along with 95% confidence intervals, when our stochastic SQP method, a stochastic subgradient method, and a stochastic projected gradient method are employed to solve logistic regression problems with linear constraints (only). The results for the best-performing algorithm are shown in bold.

		Stochastic Subgradient		Stochastic Projected Gradient	Stochastic SQP	
dataset	batch	Feasibility	Stationarity	Stationarity	Feasibility	Stationarity
a9a	16	$8.30e-03 \pm 2.32e-03$	$1.64e-01 \pm 3.55e-03$	$3.64e-02 \pm 2.95e-03$	**$1.22e-15 \pm 2.18e-16$**	**$9.99e-03 \pm 6.92e-03$**
a9a	128	$1.16e-02 \pm 4.60e-05$	$1.69e-01 \pm 2.51e-02$	$1.69e-02 \pm 2.79e-03$	**$1.64e-15 \pm 4.00e-16$**	**$7.33e-03 \pm 4.68e-05$**
australian	16	$7.94e-02 \pm 1.60e-05$	$7.94e-02 \pm 1.60e-05$	$9.17e-02 \pm 4.32e-04$	**$5.72e-06 \pm 1.56e-06$**	**$2.67e-02 \pm 6.43e-04$**
australian	128	$5.02e-01 \pm 7.04e-05$	$5.02e-01 \pm 7.04e-05$	$1.11e-02 \pm 7.19e-05$	**$6.58e-05 \pm 7.90e-07$**	**$5.50e-02 \pm 1.08e-03$**
heart	16	$3.66e-01 \pm 4.37e-03$	$3.28e+01 \pm 7.02e+00$	**$3.17e+01 \pm 6.72e+00$**	**$8.83e-03 \pm 2.77e-03$**	$3.39e+01 \pm 9.85e+00$
heart	128	$1.52e+00 \pm 4.96e-02$	**$1.23e+01 \pm 1.40e+01$**	$3.29e+01 \pm 3.21e+00$	**$1.26e-01 \pm 7.86e-04$**	$3.24e+01 \pm 1.76e+00$
ijccn1	16	$3.58e-03 \pm 2.00e-05$	$4.70e-02 \pm 6.45e-07$	$7.41e-02 \pm 3.33e-07$	**$3.03e-15 \pm 6.20e-16$**	**$1.93e-03 \pm 4.07e-06$**
ijccn1	128	$3.90e-02 \pm 4.01e-06$	$5.17e-02 \pm 1.65e-07$	$3.88e-02 \pm 6.15e-07$	**$2.16e-09 \pm 2.62e-09$**	**$1.70e-02 \pm 5.19e-05$**
ionosphere	16	$5.41e-01 \pm 8.80e-05$	$5.41e-01 \pm 8.80e-05$	$9.77e-01 \pm 8.55e-03$	**$9.61e-07 \pm 2.77e-09$**	**$4.17e-02 \pm 1.08e-03$**
ionosphere	128	$5.76e+00 \pm 3.76e-05$	$5.76e+00 \pm 3.76e-05$	$5.98e+00 \pm 3.21e-03$	**$1.31e-05 \pm 1.14e-09$**	**$1.55e-01 \pm 2.61e-03$**
madelon	16	$3.06e-02 \pm 1.85e-02$	$5.46e+01 \pm 1.25e+01$	$2.11e+01 \pm 2.72e+00$	**$2.88e-08 \pm 5.51e-08$**	**$1.09e+01 \pm 3.00e+00$**
madelon	128	$1.87e+00 \pm 7.62e-01$	$2.21e+01 \pm 1.55e+01$	**$2.16e+01 \pm 4.17e+00$**	**$5.81e-01 \pm 1.63e-02$**	$4.81e+01 \pm 4.75e+00$
mushrooms	16	$2.19e-01 \pm 6.55e-04$	$2.19e-01 \pm 6.55e-04$	$7.31e-03 \pm 3.21e-06$	**$2.08e-15 \pm 3.28e-16$**	**$5.95e-03 \pm 3.21e-05$**
mushrooms	128	$4.73e-01 \pm 4.37e-05$	$4.73e-01 \pm 4.37e-05$	$3.31e-02 \pm 7.13e-05$	**$1.66e-09 \pm 6.20e-14$**	**$3.28e-02 \pm 9.15e-04$**
phishing	16	$2.67e-02 \pm 2.76e-07$	$3.47e-02 \pm 1.39e-09$	**$2.20e-05 \pm 9.29e-06$**	**$4.26e-15 \pm 1.27e-15$**	$3.37e-03 \pm 1.27e-06$
phishing	128	$3.06e-01 \pm 1.13e-06$	$3.06e-01 \pm 1.13e-06$	$2.29e-01 \pm 8.88e-03$	**$1.83e-15 \pm 4.99e-16$**	**$2.20e-02 \pm 7.29e-03$**
sonar	16	$1.33e+00 \pm 1.08e-04$	$1.33e+00 \pm 1.08e-04$	$6.13e-01 \pm 2.22e-03$	**$7.02e-07 \pm 1.60e-07$**	**$2.34e-02 \pm 2.03e-04$**
sonar	128	$1.33e+01 \pm 1.48e-04$	$1.33e+01 \pm 1.48e-04$	$6.46e-02 \pm 4.73e-03$	**$2.07e-06 \pm 6.70e-10$**	**$2.98e-02 \pm 1.71e-03$**
splice	16	$2.56e-03 \pm 3.39e-04$	$4.56e-01 \pm 3.55e-02$	$9.65e-01 \pm 3.19e-03$	**$7.49e-14 \pm 1.03e-13$**	**$2.19e-02 \pm 4.33e-03$**
splice	128	$3.14e-01 \pm 1.09e-04$	$4.83e-01 \pm 4.65e-05$	$1.23e+00 \pm 9.44e-05$	**$3.54e-08 \pm 5.74e-09$**	**$1.07e-02 \pm 3.16e-04$**
w8a	16	$2.38e-02 \pm 1.75e-03$	$1.47e-01 \pm 1.89e-06$	$9.85e-04 \pm 3.31e-05$	**$7.35e-15 \pm 6.98e-16$**	**$6.07e-05 \pm 6.46e-05$**
w8a	128	$1.79e-02 \pm 1.25e-03$	$1.49e-01 \pm 4.64e-03$	$3.41e-02 \pm 7.43e-03$	**$5.96e-15 \pm 5.67e-16$**	**$1.20e-03 \pm 1.85e-03$**

the mini-batch size of each stochastic gradient estimate (recall (8)). For the mini-batch sizes, we employed $b_k \in \{16, 128\}$ for all problems. For each dataset and mini-batch size, we ran 5 instances with different random seeds.

A budget of 5 epochs (i.e., number of effective passes over the dataset) was used for all methods. For our stochastic SQP method, we used $\beta_k = 10^{-1}$ for all $k \in \mathbb{N}$. For the stochastic subgradient method, the merit parameter and stepsize were tuned like in Section 5.2 over the sets $\beta \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$ and $\tau \in \{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}\}$. For the stochastic projected gradient method, the stepsize was tuned using the formula $\frac{\beta}{L}$ over $\beta \in \{10^{-8}, 10^{-7}, \dots, 10^{1}, 10^{2}\}$. Overall, this meant that the stochastic subgradient and stochastic projected gradient methods were effectively run for 16 and 11 times the number of epochs, respectively, that were allowed for our method.

The results for this experiment are reported in Table 2. For every dataset and mini-batch size, we report the average feasibility and stationarity errors for the best iterates of each run along with a 95% confidence interval. The results show that our method consistently outperforms the two alternative approaches despite the fact that each of the other methods were tuned with various choices of the merit and/or stepsize parameter. For a second set of experiments, we consider problems of the form (44) with the norm constraint. The settings for the experiment were the same as above, except that the stochastic projected gradient method is not considered. The results are stated in Table 3. Again, our method regularly outperforms the stochastic subgradient method in terms of the best iterates found. For the experiments without the norm constraint, our algorithm found $\tau_{k-1} \leq \tau_k^{\text{trial,true}}$ to hold in roughly 98% of all iterations for all runs, and found this inequality to hold in all iterations in the last epoch in 100% of all runs. With the norm constraint, our algorithm found $\tau_{k-1} \leq \tau_k^{\text{trial,true}}$ to hold in roughly 97% of all iterations for all runs, and found this inequality to hold in all iterations in the last epoch in 99% of all runs.

6 Conclusion

We have proposed, analyzed, and tested a stochastic SQP method for solving equality constrained optimization problems in which the objective function is defined by an expectation of a stochastic function. Our algorithm is specifically designed for cases when the LICQ does not necessarily hold in every iteration. The convergence guarantees that we have proved for our method consider situations when the merit parameter sequence eventually remains fixed at a value that is sufficiently small, in which case the algorithm drives

Table 3: Average feasibility and stationarity errors, along with 95% confidence intervals, when our stochastic SQP method and a stochastic subgradient method are employed to solve logistic regression problems with linear constraints and a squared ℓ_2-norm constraint. The results for the best-performing algorithm are shown in bold.

		Stochastic Subgradient		Stochastic SQP	
dataset	batch	Feasibility	Stationarity	Feasibility	Stationarity
a9a	16	$4.62e-03 \pm 3.27e-04$	$1.24e-01 \pm 7.52e-02$	$\mathbf{5.52e-05 \pm 5.04e-09}$	$\mathbf{6.07e-03 \pm 2.32e-05}$
a9a	128	$4.27e-03 \pm 3.92e-04$	$1.90e-01 \pm 3.03e-03$	$\mathbf{6.38e-05 \pm 1.12e-08}$	$\mathbf{4.40e-03 \pm 1.41e-05}$
australian	16	$1.51e-01 \pm 1.07e-05$	$1.51e-01 \pm 1.07e-05$	$\mathbf{1.52e-04 \pm 5.58e-06}$	$\mathbf{5.65e-03 \pm 3.73e-05}$
australian	128	$3.96e-01 \pm 1.87e-04$	$3.96e-01 \pm 1.87e-04$	$\mathbf{3.83e-04 \pm 5.45e-05}$	$\mathbf{1.68e-02 \pm 3.29e-03}$
heart	16	$1.57e+00 \pm 5.76e-01$	$2.86e+01 \pm 1.00e+01$	$\mathbf{9.29e-01 \pm 3.47e-02}$	$\mathbf{2.65e+01 \pm 1.81e+01}$
heart	128	$\mathbf{1.33e+00 \pm 6.69e-01}$	$\mathbf{1.69e+01 \pm 2.23e+00}$	$1.88e+00 \pm 1.42e-01$	$2.93e+00 \pm 1.26e+00$
ijcnn1	16	$5.36e-02 \pm 9.37e-07$	$5.36e-02 \pm 9.37e-07$	$\mathbf{3.70e-02 \pm 9.24e-05}$	$\mathbf{4.60e-02 \pm 8.32e-03}$
ijcnn1	128	$5.41e-02 \pm 1.04e-06$	$5.41e-02 \pm 1.04e-06$	$\mathbf{3.64e-02 \pm 1.06e-04}$	$\mathbf{3.64e-02 \pm 1.06e-04}$
ionosphere	16	$3.35e-01 \pm 1.06e-03$	$3.35e-01 \pm 1.06e-03$	$\mathbf{5.79e-03 \pm 1.44e-04}$	$\mathbf{1.21e-02 \pm 4.96e-03}$
ionosphere	128	$8.70e-01 \pm 1.43e-03$	$8.70e-01 \pm 1.43e-03$	$\mathbf{5.92e-03 \pm 2.18e-05}$	$\mathbf{4.31e-02 \pm 3.52e-04}$
madelon	16	$2.66e+00 \pm 6.84e-01$	$3.86e+01 \pm 3.28e+01$	$\mathbf{3.74e-01 \pm 8.55e-02}$	$\mathbf{4.70e-01 \pm 3.27e-02}$
madelon	128	$\mathbf{2.21e+01 \pm 4.90e-01}$	$\mathbf{4.77e+01 \pm 4.84e+00}$	$7.21e+01 \pm 5.28e+00$	$7.21e+01 \pm 5.28e+00$
mushrooms	16	$1.01e-01 \pm 5.79e-05$	$1.55e-01 \pm 8.22e-06$	$\mathbf{4.06e-04 \pm 8.76e-09}$	$\mathbf{4.65e-03 \pm 3.65e-05}$
mushrooms	128	$9.72e-01 \pm 9.94e-06$	$9.72e-01 \pm 9.94e-06$	$\mathbf{6.96e-04 \pm 1.52e-09}$	$\mathbf{3.34e-03 \pm 2.35e-07}$
phishing	16	$1.30e-01 \pm 1.61e-06$	$1.30e-01 \pm 1.61e-06$	$\mathbf{3.65e-05 \pm 2.44e-08}$	$\mathbf{8.17e-03 \pm 2.43e-05}$
phishing	128	$1.53e-01 \pm 3.37e-08$	$1.53e-01 \pm 3.37e-08$	$\mathbf{1.26e-04 \pm 3.30e-09}$	$\mathbf{8.45e-04 \pm 2.73e-07}$
sonar	16	$6.45e-01 \pm 5.62e-04$	$6.45e-01 \pm 5.62e-04$	$\mathbf{3.38e-03 \pm 8.81e-06}$	$\mathbf{1.48e-02 \pm 2.58e-04}$
sonar	128	$5.04e+00 \pm 4.44e-03$	$5.04e+00 \pm 4.44e-03$	$\mathbf{5.71e-03 \pm 8.61e-06}$	$\mathbf{2.16e-02 \pm 8.48e-05}$
splice	16	$\mathbf{1.96e-03 \pm 1.78e-04}$	$4.94e-01 \pm 7.35e-03$	$3.96e-03 \pm 7.12e-07$	$\mathbf{1.03e-02 \pm 1.14e-05}$
splice	128	$1.40e+00 \pm 7.90e-05$	$1.40e+00 \pm 7.90e-05$	$\mathbf{5.52e-03 \pm 3.72e-06}$	$\mathbf{1.04e-02 \pm 1.06e-04}$
w8a	16	$1.32e-02 \pm 6.83e-04$	$1.15e-01 \pm 1.33e-02$	$\mathbf{2.15e-04 \pm 2.24e-09}$	$\mathbf{1.83e-03 \pm 8.90e-07}$
w8a	128	$5.35e-02 \pm 7.79e-02$	$1.33e-01 \pm 1.74e-07$	$\mathbf{1.67e-04 \pm 6.01e-09}$	$\mathbf{1.00e-03 \pm 1.01e-06}$

stationarity measures for the constrained optimization problem to zero, and situations when the merit parameter vanishes, which may indicate that the problem is degenerate and/or infeasible. Numerical experiments demonstrate that our algorithm consistently outperforms alternative approaches in the highly stochastic regime.

A Deterministic Analysis

In this appendix, we prove that Theorem 1 holds, where in particular we consider the context when $g_k = \nabla f(x_k)$ and $\beta_k = \beta$ satisfying (23) for all $k \in \mathbb{N}$. For this purpose, we introduce a second termination condition in Algorithm 1. In particular, after line 7, we terminate the algorithm if both $\|g_k + J_k^T y_k\|_2 = 0$ and $\|c_k\|_2 = 0$. In this manner, if the algorithm terminates finitely, then it returns an infeasible stationary point (recall (4)) or primal-dual stationary point for problem (1) and there is nothing left to prove. Hence, without loss of generality, we proceed under the assumption that the algorithm runs for all $k \in \mathbb{N}$.

Throughout our analysis in this appendix, we simply refer to the tangential direction as u_k, the full search direction as $d_k = v_k + u_k$, etc., even though it is assumed throughout this appendix that these are the *true* quantities computed using the true gradient $\nabla f(x_k)$ for all $k \in \mathbb{N}$.

It follows in this context that both Lemma 1 and Lemma 2 hold. In addition, Lemma 3 holds, where, in the proof, the case that $d_k = 0$ can be ignored due to the following lemma.

Lemma 17. *For all $k \in \mathbb{N}$, one finds that $d_k = v_k + u_k \neq 0$.*

Proof. Proof. For all $k \in \mathbb{N}$, the facts that $v_k \in \text{Range}(J_k^T)$ and $u_k \in \text{Null}(J_k)$ imply $d_k = v_k + u_k = 0$ if and only if $v_k = 0$ and $u_k = 0$. Since we suppose in our analysis that the algorithm does not terminate finitely with an infeasible stationary point, it follows for all $k \in \mathbb{N}$ that $\|J_k^T c_k\|_2 > 0$ or $\|c_k\|_2 = 0$. If $\|J_k^T c_k\|_2 > 0$, then Lemma 1 implies that $v_k \neq 0$, and the desired conclusion follows. Hence, we may proceed under the assumption that $\|c_k\|_2 = 0$. In this case, it follows under Assumption 3 that $g_k + J_k^T y_k = 0$ if and only if $u_k = 0$, which under our supposition that the algorithm does not terminate finitely means that $u_k \neq 0$. □ □

We now prove a lower bound on the reduction in the merit function that occurs in each iteration. This is a special case of Lemmas 9 and 13 for the deterministic setting.

Lemma 18. *For all $k \in \mathbb{N}$, it holds that $\phi(x_k, \tau_k) - \phi(x_k + \alpha_k d_k, \tau_k) \geq \eta \alpha_k \Delta l(x_k, \tau_k, g_k, d_k)$.*

Proof. Proof. For all $k \in \mathbb{N}$, it follows by the definition of $\alpha_k^{\rm suff}$ that (recall (20))

$$\phi(x_k + \alpha d_k, \tau_k) - \phi(x_k, \tau_k) \leq -\eta \alpha \Delta l(x_k, \tau_k, g_k, d_k) \text{ for all } \alpha \in [0, \alpha_k^{\rm suff}].$$

If $\|u_k\|_2^2 \geq \chi_k \|v_k\|_2^2$, then the only way that $\alpha_k > \alpha_k^{\rm suff}$ is if

$$\frac{2(1-\eta)\beta\xi_k\tau_k}{\tau_k L + \Gamma} > \min\left\{\frac{2(1-\eta)\beta\Delta l(x_k, \tau_k, g_k, d_k)}{(\tau_k L + \Gamma)\|d_k\|_2^2}, 1\right\}.$$

By (23), the left-hand side of this inequality is less than 1, meaning $\alpha_k > \alpha_k^{\rm suff}$ only if

$$\frac{2(1-\eta)\beta\xi_k\tau_k}{\tau_k L + \Gamma} > \frac{2(1-\eta)\beta\Delta l(x_k, \tau_k, g_k, d_k)}{(\tau_k L + \Gamma)\|d_k\|_2^2} \iff \xi_k\tau_k > \frac{\Delta l(x_k, \tau_k, g_k, d_k)}{\|d_k\|_2^2}.$$

However, this is not true since $\xi_k \leq \xi_k^{\rm trial}$ for all $k \in \mathbb{N}$. Following a similar argument for the case when $\|u_k\|_2^2 < \chi_k \|v_k\|_2^2$, the desired conclusion follows. □ □

For our purposes going forward, let us define the shifted merit function $\tilde{\phi} : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ by

$$\tilde{\phi}(x, \tau) = \tau(f(x) - f_{\rm inf}) + \|c(x)\|_2.$$

Lemma 19. *For all $k \in \mathbb{N}$, it holds that $\tilde{\phi}(x_k, \tau_k) - \tilde{\phi}(x_{k+1}, \tau_{k+1}) \geq \eta \alpha_k \Delta l(x_k, \tau_k, g_k, d_k)$.*

Proof. Proof. For arbitrary $k \in \mathbb{N}$, it follows from Lemma 18 that

$$\begin{aligned}\tau_{k+1}(f(x_k + \alpha_k d_k) - f_{\rm inf}) + \|c(x_k + \alpha_k d_k)\|_2 &\leq \tau_k(f(x_k + \alpha_k d_k) - f_{\rm inf}) + \|c(x_k + \alpha_k d_k)\|_2 \\ &\leq \tau_k(f(x_k) - f_{\rm inf}) + \|c_k\|_2 - \eta\alpha_k\Delta l(x_k, \tau_k, g_k, d_k),\end{aligned}$$

from which the desired conclusion follows. □ □

We now prove our first main result of this appendix.

Lemma 20. *The sequence $\{\|J_k^T c_k\|_2\}$ vanishes. Moreover, if there exist $k_J \in \mathbb{N}$ and $\sigma_J \in \mathbb{R}_{>0}$ such that the singular values of J_k are bounded below by σ_J for all $k \geq k_J$, then $\{\|c_k\|_2\}$ vanishes.*

Proof. Proof. Let $\gamma \in \mathbb{R}_{>0}$ be arbitrary. Our aim is to prove that the number of iterations with $x_k \in \mathcal{X}_\gamma$ (recall (32)) is finite. Since γ has been chosen arbitrarily in $\mathbb{R}_{>0}$, the conclusion will follow. By Lemma 15 and the fact that $\{\beta_k\}$ is chosen as a constant sequence, it follows that there exists $\underline{\alpha} \in \mathbb{R}_{>0}$ such that $\alpha_k \geq \underline{\alpha}$ for all $k \in \mathcal{K}_\gamma$ (regardless of whether the search direction is tangentially or normally dominated). Hence, using Lemmas 1 and 19, it follows that

$$\tilde{\phi}(x_k, \tau_k) - \tilde{\phi}(x_{k+1}, \tau_{k+1}) \geq \eta\underline{\alpha}\Delta l(x_k, \tau_k, g_k, d_k) \geq \eta\underline{\alpha}\sigma(\|c_k\|_2 - \|c_k + J_k v_k\|_2) \geq \eta\underline{\alpha}\sigma\kappa_v\kappa_c^{-1}\gamma^2.$$

Hence, the desired conclusion follows since $\{\tilde{\phi}(x_k, \tau_k)\}$ is monotonically nonincreasing by Lemma 19 and is bounded below under Assumption 1. □ □

We now show a consequence of the merit parameter eventually remaining constant.

Lemma 21. *If there exists $k_\tau \in \mathbb{N}$ and $\tau_{\min} \in \mathbb{R}_{>0}$ such that $\tau_k = \tau_{\min}$ for all $k \geq k_\tau$, then*

$$0 = \lim_{k\to\infty}\|u_k\|_2 = \lim_{k\to\infty}\|d_k\|_2 = \lim_{k\to\infty}\|g_k + J_k^T y_k\|_2 = \lim_{k\to\infty}\|Z_k^T g_k\|_2.$$

Proof. Proof. Under Assumption 1 and the conditions of the lemma, Lemmas 15 and 19 imply that $\Delta l(x_k, \tau_k, g_k, d_k)\} \to 0$, which with (14) and Lemma 1 implies that $\{\|u_k\|_2\} \to 0$, $\{\|v_k\|_2\} \to 0$, and $\{\|J_k^T c_k\|_2\} \to 0$. The remainder of the conclusion follows from Assumption 3 and (9). □ □

The proof of Theorem 1 can now be completed.

Proof. Proof of Theorem 1. The result follows from Lemmas 8, 20, and 21. □

Acknowledgments.

This material is based upon work supported by the U.S. National Science Foundation's Division of Computing and Communication Foundations under award number CCF-1740796 and by the Office of Naval Research under award number N00014-21-1-2532.

References

[1] 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.

[2] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

[3] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):1–27, 2011.

[4] Changan Chen, Frederick Tung, Naveen Vedula, and Greg Mori. Constraint-aware deep neural network compression. In *Proceedings of the European Conference on Computer Vision (ECVC)*, pages 400–415, 2018.

[5] Frank E. Curtis, Jorge Nocedal, and Andreas Wächter. A matrix-free algorithm for equality constrained optimization problems with rank deficient Jacobians. *SIAM Journal on Optimization*, 20(3):1224–1249, 2009.

[6] N. I. M. Gould, S. Lucidi, M. Roma, and Ph. L. Toint. Solving the trust-region subproblem using the Lanczos method. *SIAM Journal on Optimization*, 9(2):504–525, 1999.

[7] Nicolas I. M. Gould, Dominique Orban, and Philippe L. Toint. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications*, 60:545–557, 2015.

[8] S. P. Han. A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3):297–309, 1977.

[9] S. P. Han and O. L. Mangasarian. Exact penalty functions in nonlinear programming. *Mathematical Programming*, 17:251–269, 1979.

[10] Elad Hazan and Haipeng Luo. Variance-reduced and projection-free stochastic optimization. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 1263–1271, 2016.

[11] Soumava Kumar Roy, Zakaria Mhammedi, and Mehrtash Harandi. Geometry aware constrained optimization techniques for deep learning. In *Proceedings of Computer Vision and Pattern Recognition (CVPR)*, pages 4460–4469, 2018.

[12] Francesco Locatello, Alp Yurtsever, Olivier Fercoq, and Volkan Cevher. Stochastic Frank-Wolfe for composite convex minimization. In *Proceedings of Neural Information Processing Systems (NeurIPS)*, pages 14269–14279, 2019.

[13] Haihao Lu and Robert M Freund. Generalized stochastic Frank-Wolfe algorithm with stochastic substitute gradient for structured convex opt. *Math. Prog*, pages 1–33, 2020.

[14] Sen Na, Mihai Anitescu, and Mladen Kolar. An adaptive stochastic sequential quadratic programming with differentiable exact augmented Lagrangians. *arXiv preprint arXiv:2102.05320*, 2021.

[15] Yatin Nandwani, Abhishek Pathak, and Parag Singla. A primal-dual formulation for deep learning with constraints. In *Proceedings of Neural Information Processing Systems (NeurIPS)*, pages 12157–12168, 2019.

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

[17] E. O. Omojokun. *Trust Region Algorithms for Optimization with Nonlinear Equality and Inequality Constraints.* PhD thesis, University of Colorado, Boulder, CO, USA, 1989.

[18] M. J. D. Powell. A fast algorithm for nonlinearly constrained optimization calculations. In *Numerical Analysis*, Lecture Notes in Mathematics, pages 144–157. Springer, Berlin, 1978.

[19] Sathya N Ravi, Tuan Dinh, Vishnu Suresh Lokhande, and Vikas Singh. Explicitly imposing constraints in deep networks via conditional gradients gives improved generalization and faster convergence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4772–4779, 2019.

[20] Sashank J Reddi, Suvrit Sra, Barnabás Póczos, and Alex Smola. Stochastic Frank-Wolfe methods for nonconvex optimization. In *2016 54th Annual Allerton Conference*, pages 1244–1251. IEEE, 2016.

[21] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[22] Herbert Robbins and David Siegmund. A convergence theorem for nonnegative almost supermartingales and some applications. In Jagdish S. Rustagi, editor, *Optimizing Methods in Statistics.* Academic Press, 1971.

[23] T. Steihaug. The conjugate gradient method and trust regions in large scale optimization. *SIAM Journal on Numerical Analysis*, 20(3):626–637, 1983.

[24] Mingrui Zhang, Zebang Shen, Aryan Mokhtari, Hamed Hassani, and Amin Karbasi. One sample stochastic Frank-Wolfe. In *AISTATS*, pages 4012–4023, 2020.