A LOG-BARRIER NEWTON-CG METHOD FOR BOUND CONstrained optimization WITH COMPLEXITY GUARANTEES*

MICHAEL O’NEILL AND STEPHEN J. WRIGHT

Abstract. We describe an algorithm based on a logarithmic barrier function, Newton’s method, and linear conjugate gradients, that obtains an approximate minimizer of a smooth function over the nonnegative orthant. We develop a bound on the complexity of the approach, stated in terms of the required accuracy and the cost of a single gradient evaluation of the objective function and/or a matrix-vector multiplication involving the Hessian of the objective. The approach can be implemented without explicit calculation or storage of the Hessian.

1 Introduction

We consider the following constrained optimization problem:

\( \min_{x} f(x) \quad \text{s.t.} \quad x \geq 0, \)

where \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is a nonconvex function, twice uniformly Lipschitz continuously differentiable in the interior of the nonnegative orthant. We assume that explicit storage of the Hessian \( \nabla^2 f(x) \) for \( x > 0 \) is undesirable, but that Hessian-vector products of the form \( \nabla^2 f(x)v \) can be computed for arbitrary vectors \( v \). Computational differentiation software [19] can be used to evaluate such products at a cost that is a small multiple of the cost of evaluation of the gradient \( \nabla f \).

The problem (1) is well studied, with numerous algorithms being proposed over the years, based on such strategies as active set, gradient projection, and Newton’s method. Other possible approaches include interior-point and barrier methods, which generate iterates that remain strictly feasible. The primal log-barrier method minimizes the log-barrier function

\( \phi_\mu(x) = f(x) - \mu \sum_{i=1}^{n} \log(x_i), \)

for some decreasing sequence of positive scalars \( \mu \) [18]. The function \( \phi_\mu \) can be minimized using Newton’s method with a line search strategy that maintains strict positivity of the components of \( x \) as well as ensuring sufficient decrease at each iteration.

Our goal in this paper is to design and analyze a method with attractive worst-case complexity guarantees which are comparable to those that have been attained recently for unconstrained minimization of smooth nonconvex functions. The algorithm we describe in this paper combines the primal log-barrier formulation (2) with the Newton-Conjugate-Gradient ("Newton-CG") algorithm of [25]. We minimize the log-barrier function \( \phi_\mu \) for only a single value of \( \mu \), chosen judiciously to ensure that its approximate minimizer coincides with an approximate solution to (1) that satisfies our accuracy criteria. The Newton-CG method applied to \( \phi_\mu \) uses a safeguarded version of the linear CG method to minimize a slightly-damped second-order Taylor series approximation of \( \phi_\mu \) at each iteration. In contrast to its application to unconstrained optimization, the linear system is preconditioned to control the norm of its coefficient matrix to ensure that the number of CG iterations is bounded by a

---

*Version of April 5, 2019. Research supported by NSF Awards IIS-1447449, 1628384, 1634597, and 1740707; AFOSR Award FA9550-13-1-0138; Subcontracts 3F-30222 and 8F-30039 from Argonne National Laboratory; and Award N660011824020 from the DARPA Lagrange Program.
quantity that depends on the accuracy of the desired solution. The safeguarded CG method monitors its iterates for evidence of indefiniteness in the Hessian, and outputs a direction of negative curvature for this matrix if indefiniteness is detected. If no indefiniteness is detected, this CG procedure finds an approximate Newton step. In either case, we do a backtracking line search along the chosen direction, and show that the decrease in $\phi_\mu$ is sufficient to place an overall bound on the number of iterations, allowing worst-case complexity results to be proved.

Although practical efficiency of the method is not our main concern in this paper, we note that our method is a “long-step” interior-point method, of the kind that has been useful in other settings.

The rest of this paper is organized as follows. Section 2 reviews related work. In Section 3 we derive a first- and second-order approximate optimality condition for (1). Section 4 describes our log-barrier Newton-CG algorithm, while Section 5 presents the worst-case complexity analysis for the first- and second-order approximate KKT conditions. Some conclusions appear in Section 6.

**Assumptions, Background, Notation.** We make the following standard assumption throughout.

**Assumption 1.** The function $f$ is twice uniformly Lipschitz continuously differentiable on the interior of the nonnegative orthant $\mathbb{R}_+^n$. We denote by $L_g$ the Lipschitz constant for $\nabla f$ and $L_H$ the Lipschitz constant for $\nabla^2 f$ on this set.

These properties imply the following useful inequalities:

\begin{align}
\|\nabla f(y) - \nabla f(x) - \nabla^2 f(x)(y - x)\| &\leq \frac{1}{2} L_H \|x - y\|^2, \\
f(y) &\leq f(x) + \nabla f(x)^T (y - x) + \frac{1}{2} (y - x)^T \nabla^2 f(x)(y - x) + \frac{1}{6} L_H \|x - y\|^3,
\end{align}

for all $x,y > 0$. (Here and throughout we use $\| \cdot \|$ to denote the Euclidean norm, or its induced norm on matrices.)

Order notation $O$ is used in its usual sense, whereas $\tilde{O}$ represents $O$ with logarithmic terms omitted.

We define $e = (1, 1, \ldots, 1)^T$ to be the vector of ones and $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)^T$ to be the unit vector with 1 as the $i$th component and zeros elsewhere. The $i$th component of a vector $v$ is denoted by $v_i$ or $[v]_i$. Given a vector $x \in \mathbb{R}_+^n$ (where $\mathbb{R}_+^n$ is the nonnegative orthant), we denote by $X$ the diagonal matrix formed by the components of $x$, and by $\bar{x}$ the vector whose components are $\min(x_i, 1)$, and by $\bar{X}$ the diagonal matrix formed from $\bar{x}$. That is,

\begin{align}
X &= \text{diag} (x_1, x_2, \ldots, x_n), \quad \bar{x} = \min(x, e), \quad \bar{X} = \text{diag} (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n).
\end{align}

Our algorithm seeks a point $x$ satisfying the following approximate optimality conditions for (1):

\begin{align}
(6a) \quad x &> 0 \\
(6b) \quad \nabla f(x) &\geq -\epsilon_g e \\
(6c) \quad \|X \nabla f(x)\|_\infty &\leq \epsilon_g \\
(6d) \quad X \nabla^2 f(x) \bar{X} &\geq -\epsilon_H I,
\end{align}

\footnote{We use a threshold of 1 for clarity of presentation. Any other positive value could be used instead, with minimal effect on the results.}
for small positive tolerances $\epsilon_g$ and $\epsilon_H$. The conditions (6c) and (6d) differ from the scaled gradient and Hessian conditions used elsewhere, through the substitution of the bounded matrix $\bar{X}$ for $X$. The theoretical basis for these conditions as well as their relation to those used in previous works is presented in Section 3.

2 Related Work There is considerable recent work on algorithms for unconstrained smooth nonconvex optimization that have optimal worst-case iteration complexity for finding points that satisfy approximate first- and second-order optimality conditions. When applied to twice Lipschitz continuously differentiable functions, classical Newton-trust-region schemes [15] require at most $O\left(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\}\right)$ iterations [11] to find a point satisfying

$$\|\nabla f(x)\| \leq \epsilon_g \quad \text{and} \quad \lambda_{\text{min}}(\nabla^2 f(x)) \geq -\epsilon_H.$$  

However, for problems of this class, the optimal iteration complexity for finding a first-order optimal point is $O(\epsilon_g^{-3/2})$ [6]. This iteration complexity was first achieved by cubic regularization of Newton’s method when $\epsilon_H = \epsilon_g^{1/2}$ [23]. Since 2016, numerous other algorithms have also been proposed that match this iteration bound; see for example [4, 9, 16, 17, 22].

Some works also account for the computational cost of each iteration, thus yielding a bound on the overall computational complexity. Two independently proposed algorithms, respectively based on adapting accelerated gradient to the nonconvex setting [8] and approximately solving the cubic regularization subproblem [1], require $O(\epsilon_g^{-7/4})$ operations (with high probability, showing dependency only on $\epsilon_g$) to find a point $x$ that satisfies (7) when $\epsilon_H = \epsilon_g^{1/2}$. The difference of a factor of $\epsilon_g^{-1/4}$ with the iteration complexity bounds arises from the cost of computing a negative curvature direction of $\nabla^2 f(x_k)$ and/or the cost of solving a linear system. The probabilistic nature of the bound is due to the introduction of randomness in the curvature estimation process. A complexity bound of the same type was also established for a variant of accelerated gradient based only on gradient calculations, that periodically adds a random perturbation to the iterate when the gradient norm is small [21].

In another line of work, [26] developed a damped Newton algorithm which inexactly minimizes the Newton system by the method of conjugate gradients and requires at most $O(\min\{n\epsilon_g^{-3/2}, \epsilon_g^{-7/4}\})$ operations to satisfy (7), to high probability. For purposes of computational complexity, this paper defines the unit of computation to be one Hessian-vector product or one gradient evaluation. We also adopt this definition here; it relies implicitly on the observation from computational / algorithmic differentiation [19] that these two operations differ in cost only by a modest factor, independent of the dimension $n$. In a followup to [26], the paper [25] built on techniques from [7] to create a modified CG method to solve the Newton system. This algorithm, which is a foundation of the method described in this paper, again finds a point satisfying (7) in $O(\min\{n\epsilon_g^{-3/2}, \epsilon_g^{-7/4}\})$ operations, to high probability, and requires the same number of operations to find an approximate first-order critical point deterministically.

A number of algorithms have also been proposed for constrained optimization problems which require at most $O(\max\{\epsilon_g^{-3/2}, \epsilon_H^{-3}\})$ iterations to find a point which satisfies some first-order (and sometimes second-order) optimality condition(s). Although the optimality conditions vary between papers, [10, 12, 13] all achieve this iteration complexity bound for some first order optimality condition by solving a constrained cubic regularization subproblem at each iteration. Another recent work finds
a first-order point in $O(\epsilon_g^{-3/2})$ iterations for linear equality and bound constraints through the use of an active set method [5]. When optimizing on a single face of the polytope, this method also uses a cubic regularization model. However, these papers do not account for the cost of solving the subproblem at each iteration, noting either that this subproblem may be NP-hard, or suggesting that a simple first-order, gradient-based method can solve it reliably.

Turning to our bound-constrained problem (1), a second-order interior-point method was proposed in [3]. This method minimizes a preconditioned second-order trust-region model at each iteration and finds a point satisfying approximate second-order conditions in at most $O(\epsilon_g^{-3/2})$ iterations. However, the first-order conditions are strictly weaker than those used in the current work as they consist only of feasibility of $x$ along with a scaled gradient condition that is an “unbounded” version of (6c) in which $\bar{X}$ is replaced by $X$. The absence of (6b) in the optimality conditions weakens the connection to the KKT conditions for (1), in that the limit points of methods which satisfy the scaled gradient condition may not be local minimizers as $\epsilon_g$ approaches 0. (See [20, Section 2] for a discussion of this issue.) Our approximate optimality conditions (6) here do not suffer from these issues, as we show in Section 3. In a follow up to [3], an interior-point method for linear equality and bound constraints was proposed in [20]. This method, which also achieves iteration complexity $O(\epsilon_g^{-3/2})$, applies a constrained second-order trust-region algorithm to the log-barrier function, with a (potentially) small trust-region radius. The subproblem is solved using a bisection scheme, leading to an overall computational complexity similar to that of the cubic regularization method with direct subproblem solves for unconstrained optimization.

In this paper, we adapt the Newton-CG method of [25] for unconstrained optimization to the problem of minimizing the primal log-barrier function (2), for a small, fixed value of $\mu$. We target the optimality conditions (6), which avoid enforcing tighter conditions on Hessian and gradient components that correspond to components of $x$ that are far from zero at optimality. This change allows us to solve a preconditioned Newton system of linear equations at each iteration in which the norm of the matrix can be bounded by a constant independent of iteration number. The Capped CG method developed in [25] is used to solve this system, returning a useful search direction in a reasonable number of iterations. Our algorithm finds a point satisfying (6) in $\tilde{O}(\epsilon_g^{-3/2})$ iterations and $\tilde{O}(\min\{n\epsilon_g^{-3/2}, \epsilon_g^{-7/4}\})$ gradient evaluations/Hessian vector products when $\epsilon_H = \epsilon_g^{1/2}$, making our algorithm the first method for bound-constrained optimization with this overall computational complexity. Further, our algorithm has the appealing practical feature that it puts minimal restrictions on the step size, allowing the line search to take steps that are much closer to the boundary than the current iterate.

3 Approximate Optimality Conditions
We now discuss first- and second-order optimality criteria for (1) in a form that can be related to the approximate optimality criteria (6) that are targeted by our algorithm. We show that points satisfying these necessary conditions are the limits of sequences of points that satisfy our approximate criteria (6). We then compare our approximate criteria with similar conditions that have been proposed previously, and argue that ours are more appropriate.

3.1 Deriving Approximate Optimality Conditions from Exact Conditions
First-order conditions for $x$ to be a solution of (1) are that there exists a vector
s^* \in \mathbb{R}^n \text{ such that}

\begin{equation}
\nabla f(x) - \sum_{i=1}^{n} s^*_i e_i = 0, \quad (x, s^*) \geq 0, \quad x_i s^*_i = 0 \quad \text{for all } i = 1, 2, \ldots, n.
\end{equation}

Our second-order condition is a modified version of the condition derived in [2]. It requires the existence of a vector \( \theta^* \) such that

\begin{equation}
\nabla^2 f(x) + \sum_{i=1}^{n} \theta^*_i e_i e_i^\top \succeq 0, \quad \theta^* \geq 0, \quad x_i^2 \theta^*_i = 0 \quad \text{for all } i = 1, 2, \ldots, n.
\end{equation}

This is equivalent to a “weak” form of second-order necessary conditions for (1), namely \( \nabla^2 f(x) \geq 0 \), \( x_i^2 \theta^*_i = 0 \) for all \( i = 1, 2, \ldots, n \).

This is known to be an NP-hard problem [24].

The following result shows that a point \( x^* \) satisfying both (8) and (9) can be expressed in terms of the limit of sequences that satisfy approximate forms of these two optimality conditions.

**Theorem 1.** Let \( f \) be twice continuously differentiable on the interior of \( \mathbb{R}_+^n \). Let \( x^* \) be a local solution of (1). Then there exists a sequence of approximate solutions \( \{x^k\} \) with \( x^k > 0 \); sequences of approximate Lagrange multipliers \( \{s^k\} \) and \( \{\theta^k\} \), with \( s^k \geq 0 \) and \( \theta^k \geq 0 \); and a sequence of scalars \( \{\delta_k\} \) with \( \delta_k > 0 \) and \( \delta_k \to 0 \) such that the following conditions hold:

\begin{enumerate}
  \item[10a] \( x^k > 0 \) for all \( k \) and \( x^k \to x^* \),
  \item[10b] \( \nabla f(x^k) - \sum_{i=1}^{n} s_i^k e_i \to 0 \),
  \item[10c] \( \min\{x^k_i, 1\} s_i^k \to 0 \) for all \( i = 1, 2, \ldots, n \),
  \item[10d] \( d^\top \left( \nabla^2 f(x^k) + \sum_{i=1}^{n} \theta_i^k e_i e_i^\top + \delta_k I \right) d \geq 0 \) for all \( d \in \mathbb{R}^n \),
  \item[10e] \( \min\{x^k_i, 1\} \theta_i^k \to 0 \) for all \( i = 1 \ldots n \).
\end{enumerate}

The proof of this result follows similar logic to that of [20, Theorem 1]; we spell out the details in Appendix A.1.

Theorem 1 suggests that we should declare \( x \) to be an approximate interior solution of (1) when there exist \( s \in \mathbb{R}_+^n \) and \( \theta \in \mathbb{R}_+^n \) such that

\begin{enumerate}
  \item[11a] \( x > 0 \)
  \item[11b] \( \left\| \nabla f(x) - \sum_{i=1}^{n} s_i e_i \right\|_{\infty} \leq \epsilon_g \)
  \item[11c] \( \|\bar{X}s\|_{\infty} \leq \epsilon_g \)
  \item[11d] \( d^\top \left( \nabla^2 f(x) + \sum_{i=1}^{n} \theta_i e_i e_i^\top + \epsilon_H I \right) d \geq 0 \) for all \( d \in \mathbb{R}^n \)
  \item[11e] \( \|\bar{X}^2\theta\|_{\infty} \leq \epsilon_H \).
\end{enumerate}
We will now describe the connection between our approximate optimality conditions (6) and the conditions (11).

**Theorem 2.** Let $x$ be a point satisfying (6). Then there exist $s \in \mathbb{R}^n_+$ and $\theta \in \mathbb{R}^n_+$ such that (11) holds at $x$.

**Proof.** Clearly (11a) holds. Let $s_i := \max\{0, |\nabla f(x)|_i\}$ for $i = 1 \ldots n$, so that $s \in \mathbb{R}^n_+$ and, by direct substitution, we have (11b) and (11c). Our second-order condition (6d) is that

$$d^T (\bar{X} \nabla^2 f(x) \bar{X} + \epsilon_H I) d \geq 0,$$

for all $d \in \mathbb{R}^n$.

Since $\bar{X}^{-1}$ exists and is positive definite, we have

$$d^T \left( \nabla^2 f(x) + \sum_{i=1}^n \frac{\epsilon_H}{\min\{x_i, 1\}^2} e_i e_i^T \right) d \geq 0,$$

for all $d \in \mathbb{R}^n$.

Therefore, by choosing $\theta_i = \epsilon_H / \min\{x_i, 1\}^2$ for all $i = 1, 2, \ldots n$, we have that $\theta \geq 0$ and that (11d) and (11e) are both satisfied. \(\Box\)

### 3.2 Comparison with Previously Proposed Approximate Conditions

The conditions (8) and (9) directly motivate the conditions used in the interior-point method of [20], which are

\begin{align}
(12a) & \quad x > 0 \\
(12b) & \quad \nabla f(x) \geq -\epsilon_g e \\
(12c) & \quad \|X \nabla f(x)\|_{\infty} \leq \epsilon_g \\
(12d) & \quad d^T (X \nabla^2 f(x) X + \sqrt{\epsilon_g} I) d \geq 0.
\end{align}

The scaled first-order condition (12c) and scaled second-order condition (12d), are commonly used optimality conditions for (1) [3, 14]. However, without additional assumptions on the objective function $f$, sequences of $\epsilon$-scaled first-order points may not converge to local minimizers of $f$ [20]. This difficulty is remedied with the inclusion of (12b), introduced in [20], which follows directly from the first-order optimality conditions (8).

These conditions can be overly stringent for coordinates $i$ in which $x_i \gg 0$. In this case, the complementarity condition (12c), requires $|\nabla f(x)|_i$ to be very small. Similarly, (12d) requires that the Hessian in the subspace spanned by these coordinates can have only minimal negative curvature. Such requirements contrast sharply with the case of unconstrained minimization. In the limiting scenario in which all of the coordinates of $x$ are far from the boundary, these approximate first-order conditions are significantly harder to satisfy than in the (equivalent) unconstrained formulation.

To remedy this situation, our approximate optimality conditions (6) contain scalings by $x_i$ only when $x_i \in (0, 1]$. Our conditions thus interpolate between the bound-constrained case (when $x_i$ is small) and the unconstrained case (when $x_i$ is large) while also controlling the norm of the matrix used in our optimality conditions.

### 4 Log-Barrier Newton-CG Algorithm

We now give an overview of our Log-Barrier Newton-CG (LBNCG) algorithm, defined in Algorithm 1, along with its component parts.

The main branch in each iteration is conditional on the approximate first-order optimality conditions, (6b) and (6c). When one or both of these conditions are not
Procedure 3 is invoked to certify either that the second-order optimality condition of sufficient negative curvature for \( \bar{\phi} \) returns either an approximate solution to the linear system (14), or else a direction of sufficient negative curvature \( \bar{\phi} \) according to the definition (2) of the barrier function, we have

\[
\nabla \phi(x) = \nabla f(x) - \mu X^{-1} e \quad \text{and} \quad \nabla^2 \phi(x) = \nabla^2 f(x) + \mu X^{-2}.
\]

Algorithm 2, which is described further in Section 4.1 and in the earlier paper [25], returns either an approximate solution to the linear system (14), or else a direction of sufficient negative curvature for \( \bar{X}_k \nabla^2 \phi_{\mu}(x^k) \bar{X}_k \).

Alternatively, when (6b) and (6c) are satisfied, a “Minimum Eigenvalue Oracle” (Procedure 3) is invoked to certify either that the second-order optimality condition

\[
\nabla^2 f(x) \preceq \mu X^{-2}
\]

is satisfied, the Capped CG method (Algorithm 2) is applied to the damped, preconditioned Newton system

\[
\bar{X}_k \nabla^2 \phi_{\mu}(x^k) \bar{X}_k + 2\epsilon_H I) \bar{d} = \bar{X}_k \bar{\phi}_{\mu}(x^k),
\]

where according to the definition (2) of the barrier function \( \bar{\phi}_{\mu} \), we have

\[
\nabla \phi_{\mu}(x) = \nabla f(x) - \mu X^{-1} e \quad \text{and} \quad \nabla^2 \phi_{\mu}(x) = \nabla^2 f(x) + \mu X^{-2}.
\]

\[
\phi_{\mu}(x^k + \alpha_k \bar{X}_k \bar{d}^k) < \phi_{\mu}(x^k) - \frac{\eta}{6} \alpha_k^3 ||d^k||^3;
\]

\[
x^{k+1} \leftarrow x^k + \alpha_k \bar{X}_k \bar{d}^k;
\]

end for
particular choice of we keep the distinction between In our current context, this choice is embedded more deeply into the analysis, but finally, we note that when \( \hat{\mu} \) results. The particular choice \( \epsilon \) mindful of this effect, our focus is on the dependence on the tolerance \( \mu \). We note that a value of \( \beta \) close to its upper bound of 1 results in aggressive steps that may approach the zero bounds closely. Steps of this kind are favored in practical interior-point methods. We will see in later sections that a factor \((1 - \beta)\) emerges in the complexity results, leading to weaker bounds if \( \beta \) is too close to 1. Though we are mindful of this effect, our focus is on the dependence on the tolerance \( \epsilon_g \). The choice of \( \beta \) is independent of \( \epsilon_g \); we would not \( \beta \) in response to a change in the tolerance \( \epsilon_g \).

We set a number of parameters at the beginning of the algorithm, including the particular choice \( \epsilon_H = \epsilon_g^{1/2} \). This choice is commonly made in the unconstrained optimization literature too, for purposes of aligning two different complexity expressions. In our current context, this choice is embedded more deeply into the analysis, but we keep the distinction between \( \epsilon_H \) and \( \epsilon_g \) to maintain the generality of individual results. The particular choice \( \mu = \epsilon_g/A \) of the barrier parameter is key to the complexity result. Finally, we note that when \( M \) is an upper bound on \( \|\nabla^2 f(x)\| \) for all \( x \) of interest, we have

\[
\|X\nabla^2 \phi_{\mu}(x)\hat{X}\| \leq \|X\nabla^2 f(x)\hat{X}\| + \mu\|X^{-2}\hat{X}\| \leq \|\nabla^2 f(x)\| + \mu \leq \hat{M} + \mu,
\]

so that \( \|H\| \leq M_{\mu} \) for \( H \) defined as the input of Algorithm 2 in Algorithm 1.

### 4.1 Capped Conjugate Gradient
Algorithm 2 is a safeguarded version of the conjugate gradient (CG) procedure for either solving the linear system \((H + 2\epsilon I)y = -g\), or else detecting a direction \( d \) such that \( d^T H d \leq -\epsilon \|d\|^2 \). This method, which was described in [25], consists of classical CG iterations plus various checks to determine whether (a) the upper bound \( M \) on \( \|H\| \) is adequate, and (b) negative curvature in \( H \) has been detected. One of the techniques for detecting negative curvature is the too-slow-convergence criterion \( \|r\| > \sqrt{T\tau} \|r_0\| \) (where \( T \) and \( \tau \) both depend on the bound \( M \)). By Theorem 6, this behavior can occur only when there exists some \( i \in \{0, \ldots, j-1\} \) such that \( (y^{i+1} - y^i)^\top \hat{H}(y^{i+1} - y^i) < \epsilon \|y^{i+1} - y^i\|^2 \) holds. In this situation, Algorithm 2 returns \( d = y^{i+1} - y^i \) as a direction of sufficient negative curvature.

Algorithm 2 is called from Algorithm 1 with \( H = \hat{X}_k \nabla^2 \phi_{\mu}(x^k)\hat{X}_k \) which, as we note in (15), has norm bounded by \( M_{\mu} = \hat{M} + \mu \), where \( \hat{M} \) is the bound on \( \|\nabla^2 f(x^k)\| \). Hence the value of \( M \) in Algorithm 2 will never be larger than this value.

Altogether, the safeguards mentioned above and the diagonal preconditioning strategy guarantee that Capped CG requires \( \min \left\{ n, \hat{O}(\epsilon^{-1/2}) \right\} \) iterations to terminate. A derivation of this bound is given in Section 5.1.

### 4.2 Minimum Eigenvalue Oracle
The Minimum Eigenvalue Oracle (Procedure 3) is called when the approximate first-order conditions (6b), (6c) are satisfied. This procedure either verifies that the approximate second-order condition (6d) is
Algorithm 2 Capped Conjugate Gradient

Inputs: Symmetric matrix $H \in \mathbb{R}^{n \times n}$; vector $g \neq 0$; damping parameter $\epsilon \in (0, 1)$; desired relative accuracy parameter $\zeta \in (0, 1)$; desired accuracy $c_\mu \in (0, 1)$; Optional input: scalar $M \geq 0$ such that $\|H\| \leq M$ (set to 0 if not provided);

Outputs: d_type, $d$;

Secondary outputs: final values of $M$, $\kappa$, $\zeta_r$, $\tau$, and $T$;

Set

\[
\bar{H} := H + 2\epsilon I, \quad \kappa := \frac{M + 2\epsilon}{\epsilon}, \quad \zeta_r := \frac{\zeta}{3\kappa}, \quad \tau := \frac{\sqrt{\kappa}}{\sqrt{\kappa} + 1}, \quad T := \frac{4\kappa^4}{(1 - \sqrt{T})^2};
\]

$y^0 \leftarrow 0$, $r^0 \leftarrow g$, $p^0 \leftarrow -g$, $j \leftarrow 0$;

if $(p^0)^\top \bar{H}p^0 < \epsilon \|p^0\|^2$ then

Set $d = p^0$ and terminate with d_type=NC;

else if $\|Hp^0\| > M\|p^0\|$ then

Set $M \leftarrow \|Hp^0\|/\|p^0\|$ and update $\kappa, \zeta_r, \tau, T$ accordingly;

end if

while TRUE do

$\alpha_j \leftarrow (r^j)^\top r^j / (p^j)^\top \bar{H}p^j$; {Begin Standard CG Operations}

$y^{j+1} \leftarrow y^j + \alpha_j p^j$;

$r^{j+1} \leftarrow r^j + \alpha_j \bar{H}p^j$;

$\beta_{j+1} \leftarrow \|r^{j+1}\|^2 / \|r^j\|^2$;

$p^{j+1} \leftarrow -r^{j+1} + \beta_{j+1} p^j$; {End Standard CG Operations}

$j \leftarrow j + 1$;

if $\|Hp^j\| > M\|p^j\|$ then

Set $M \leftarrow \|Hp^j\|/\|p^j\|$ and update $\kappa, \zeta_r, \tau, T$ accordingly;

else if $\|Hy^j\| > M\|y^j\|$ then

Set $M \leftarrow \|Hy^j\|/\|y^j\|$ and update $\kappa, \zeta_r, \tau, T$ accordingly;

else if $\|Hp^j\| > M\|p^j\|$ then

Set $M \leftarrow \|Hp^j\|/\|p^j\|$ and update $\kappa, \zeta_r, \tau, T$ accordingly;

end if

if $(p^j)^\top \bar{H}y^j < \epsilon \|y^j\|^2$ then

Set $d \leftarrow y^j$ and terminate with d_type=NC;

else if $\|p^j\| \leq \zeta \|r^j\|$ and $\|r^j\|_\infty \leq c_\mu$ then

Set $d \leftarrow y^j$ and terminate with d_type=SOL;

else if $(p^j)^\top \bar{H}p^j < \epsilon \|p^j\|^2$ then

Set $d \leftarrow p^j$ and terminate with d_type=NC;

else if $\|p^j\| > \sqrt{T} \|r^j\|^2 \|r^0\|$ then

Compute $\alpha_j, y^{j+1}$ as in the main loop above;

Find $i \in \{0, \ldots, j - 1\}$ such that

\[
(16) \quad \frac{(y^{j+1} - y^j)^\top \bar{H}(y^{j+1} - y^j)}{\|y^{j+1} - y^j\|^2} < \epsilon;
\]

Set $d \leftarrow y^{j+1} - y^j$ and terminate with d_type=NC;

end if

end while
Procedure 3 Minimum Eigenvalue Oracle

Inputs: Symmetric matrix $H \in \mathbb{R}^{n \times n}$, tolerance $\epsilon > 0$;  
Optional input: Scalar $M > 0$ such that $\|H\| \leq M$;  
Outputs: An estimate $\lambda$ of $\lambda_{\min}(H)$ such that $\lambda \leq -\epsilon/2$, and vector $v$ with $\|v\| = 1$ such that $v^T Hv = \lambda$ OR a certificate that $\lambda_{\min}(H) \geq -\epsilon$. In the latter case, when the certificate is output, it is false with probability $\delta$ for some $\delta \in [0, 1)$.

satisfied as well (in which case the algorithm terminates), or else returns a direction of sufficient negative curvature for the scaled Hessian $\bar{X}_k \nabla^2 f(x^k) \bar{X}_k$, along which further progress can be made in reducing the barrier function $\phi_\mu$.

This procedure can be implemented via any method that finds the smallest eigenvalue of $H$ to an absolute precision of $\epsilon/2$ with probability at least $1 - \delta$. (A deterministic implementation based on a full eigenvalue decomposition would have $\delta = 0$.) In Section 5.3, we will establish complexity results under this general setting, and analyze the impact of the threshold $\delta$.

Several possibilities for implementing Procedure 3 have been proposed in the literature, with various guarantees. In our setting, in which Hessian-vector products and vector operations are the fundamental operations, Procedure 3 can be implemented using the Lanczos method with a random starting vector (see [8]). The following result from [25, Lemma 2] verifies its effectiveness.

**Lemma 3.** Suppose that the Lanczos method is used to estimate the smallest eigenvalue of $H$ starting with a random vector uniformly generated on the unit sphere, where $\|H\| \leq M$. For any $\delta \in [0, 1)$, this approach finds the smallest eigenvalue of $H$ to an absolute precision of $\epsilon/2$, together with a corresponding direction $v$, in at most

\[
\min \left\{ n, 1 + \left[ \frac{1}{2} \ln(2.75n/\delta^2) \sqrt{\frac{M}{\epsilon}} \right] \right\} \text{ iterations},
\]

with probability at least $1 - \delta$.

Procedure 3 can be implemented by outputting the approximate eigenvalue $\lambda$ for $H$, determined by the randomized Lanczos process, along with the corresponding direction $v$, provided that $\lambda \leq -\epsilon/2$. When $\lambda > -\epsilon/2$, Procedure 3 returns the certificate that $\lambda_{\min}(H) \geq -\epsilon$, which is correct with probability at least $1 - \delta$.

The second approach to implementing Procedure 3 is to apply the classical CG algorithm to solve a linear system in which the coefficient matrix is a shifted version of the matrix $H$ and the right-hand side is random. This procedure has essentially identical performance to Lanczos in terms of the number of iterations required to detect the required direction of sufficiently negative curvature or certify that no such direction exists. For details, we refer the reader to [25, Appendices A and B].

5 Complexity Analysis This section presents our complexity results for Algorithm 1. Section 5.1 describes the iteration complexity of Capped CG (Algorithm 2) and the properties of its outputs. Section 5.2 shows that Algorithm 1 deterministically finds a point satisfying the approximate first-order optimality conditions (6b), (6c) in at most $\tilde{O}\left(\epsilon_g^{-3/2}\right)$ iterations and at most $\tilde{O}\left(\max\left\{n \epsilon_g^{-3/2}, \epsilon_g^{-5/4}\right\}\right)$ gradient evaluations and/or Hessian-vector products. Finally, Section 5.3 shows that the same complexity holds (albeit with different constants) for finding a point which satisfies all approximate optimality conditions in (6) with high probability (rather than
Our results depend on the following assumption.

**Assumption 2.** The function $f$ is bounded below on $\mathbb{R}_+^n$, and the iterates $x^k$ of Algorithm 1 are bounded.

This assumption holds if, for example, the set of solutions for $f$ is compact and $f$ increases by a factor greater than $\mu$ times the logarithm of the distance from this solution set. Such conditions would ensure that $\phi_\mu(x) \to \infty$ as $\|x\| \to \infty$, and since our method is a descent method in $\phi_\mu$, the iterates would remain bounded. A direct consequence of this assumption and Assumption 1 is that there exists scalars $\phi_{low}$, $U_g > 0$, and $U_H > 0$ such that

$$\phi_\mu(x^k) \geq \phi_{low}, \quad \|\nabla f(x^k)\| \leq U_g, \quad \|\nabla^2 f(x^k)\| \leq U_H$$

holds for all iterates $x^k$ generated by Algorithm 1.

While this assumption is more stringent than boundedness of the level set of $f$ (an assumption that is common in unconstrained optimization), most papers that provide a worst case complexity result for constrained optimization make similar assumptions, such as boundedness of the constraint set in [20].

### 5.1 Properties of Capped CG

We begin this subsection by finding a lower bound on the norm of the right-hand side in the Newton system of Algorithm 1 (Lemma 4). We then derive a bound on the maximum number of iterations of the Capped CG method that can occur before returning a direction $\hat{d}^k$, which is either an approximate solution of (14) or a negative curvature direction for the diagonally scaled Hessian of the log-barrier function (Lemma 5). Theorem 6 verifies that the direction returned in the case of too-slow-decrease is in fact a vector with the required negative curvature properties. Finally, we present a number of properties of the search direction $d^k$ computed from the vector returned by Algorithm 2, which will be instrumental in the complexity analysis of the following sections (Lemma 7).

**Lemma 4.** Let $\mu = \epsilon_g/4$ and suppose that either (6b) or (6c) is violated at $x^k$. Then we have

$$\|\bar{X}_k \nabla \phi_\mu(x^k)\| \geq \mu.$$  

**Proof.** By definition of $\nabla \phi_\mu(x^k)$, we have

$$\|\bar{X}_k \nabla \phi_\mu(x^k)\| = \|\bar{X}_k \nabla f(x^k) - \mu \bar{X}_k X_k^{-1} e\|.$$  

Suppose first that (6b) is not satisfied at $x^k$. Thus, there exists at least one coordinate $i$ such that $[\nabla f(x^k)]_i < -\epsilon_g < 0$. If $x_i^k \leq 1$, it follows that

$$\bar{x}_i^k [\nabla f(x^k)]_i - \frac{\bar{x}_i^k}{x_i^k} \mu = \bar{x}_i^k [\nabla f(x^k)]_i - \mu < -\mu.$$  

If $x_i^k > 1$, we have $\bar{x}_i^k = 1$ so that

$$\bar{x}_i^k [\nabla f(x^k)]_i - \frac{\bar{x}_i^k}{x_i^k} \mu < [\nabla f(x^k)]_i < -\epsilon_g = -4\mu.$$  

In either case, we have from (20) that

$$\|\bar{X}_k \nabla \phi_\mu(x^k)\| \geq \mu.$$  

11
Now, suppose that (6c) does not hold, so that \(|\tilde{x}_i^k[\nabla f(x^k)]_i| > \epsilon_g\) for some \(i\). Thus,
\[
\|\tilde{X}_k \nabla \phi_\mu(x^k)\| \geq \left|\tilde{x}_i^k[\nabla f(x^k)]_i - \frac{\tilde{x}_i}{x_i} - \frac{\mu}{x_i}\right|
\geq |\tilde{x}_i^k[\nabla f(x^k)]_i| - \mu \frac{x_i}{x_i}
\geq \epsilon_g - \mu \geq 3\mu,
\]
proving the result. □

We now find the iteration bound on Algorithm 2 that was foreshadowed in Section 4.1. The precise bound in the following lemma is based on a quantity \(J(M, \epsilon, \zeta_r, c_\mu)\), for which the estimate in terms of the accuracy parameter is given following the lemma.

**Lemma 5.** The number of iterations of Algorithm 2 is bounded by
\[
\min\{n, J(M, \epsilon, \zeta_r, c_\mu)\},
\]
where \(J = J(M, \epsilon, \zeta_r, c_\mu)\) is the smallest integer such that
\[
\sqrt{T \tau}^J \|r^0\| \leq \min\left\{\hat{\zeta}_r \|r^0\|, c_\mu\right\},
\]
where \(M, \hat{\zeta}_r, T, \) and \(\tau\) are the values returned by the algorithm. If all iterates \(y_i\) generated by Algorithm 2 are stored, the number of matrix-vector multiplications required is bounded by \(\min\{n, J(M, \epsilon, \zeta_r, c_\mu)\} + 1\). If the iterates \(y_i\) must be regenerated in order to define the direction \(d\) returned after (16), this bound becomes \(2 \min\{n, J(M, \epsilon, \zeta_r, c_\mu)\} + 1\).

**Proof.** We omit a detailed proof, as the result and proof are identical to [25, Lemma 1] modulo a new definition of \(J\). We need only consider the case in which \(J < n\), where \(J\) is the index defined in the lemma. If \(\|r^J\| > \sqrt{T \tau}^J \|r^0\|\), the last termination test in Algorithm 2 ensures termination at iteration \(J\). In the alternative case \(\|r^J\| \leq \sqrt{T \tau}^J \|r^0\|\), we have by definition of \(J\) that
\[
\|r^J\| \leq \sqrt{T \tau}^J \|r^0\| \leq \min\left\{\hat{\zeta}_r \|r^0\|, c_\mu\right\}.
\]
Therefore, \(\|r^J\| \leq \hat{\zeta}_r \|r^0\|\) and \(\|r^J\|_\infty \leq \|r^J\| \leq c_\mu\) both hold. Thus, by the termination tests in Algorithm 2, termination occurs in this case as well, completing the proof. □

We can now estimate \(J(M, \epsilon, \zeta_r, c_\mu)\) when the Algorithm 2 is called by Algorithm 1. Here, we have \(r^0 = \tilde{X}_k \nabla \phi_\mu(x^k)\) and \(c_\mu = \tilde{\zeta}_\mu\), so that the right-hand side of condition (21) is
\[
\min\left\{\hat{\zeta}_r \|\tilde{X}_k \nabla \phi_\mu(x^k)\|, \tilde{\zeta}_\mu\right\}.
\]
Using the same argument as in [25], when the minimum in (22) is achieved by the first argument, we have
\[
J(M, \epsilon, \zeta_r, c_\mu) \leq \min\left\{n, \left(\sqrt{\kappa} + \frac{1}{2}\right) \ln\left(\frac{144(\sqrt{\kappa} + 1)^2 \kappa_0}{\zeta_r^2}\right)\right\}
\]
\[
= \min\left\{n, \tilde{O}\left(\epsilon^{-1/2}\right)\right\}.
\]
On the other hand, when the minimum in (22) is achieved by the second argument, an argument of [25] along with the bound
\[ \| \bar{X}_k \nabla \phi(x^k) \| \leq \| \bar{X}_k \nabla f(x^k) \| + \mu \| \bar{X}_k X_k^{-1} c \| \leq U_g + \mu \sqrt{n} \]
shows that
\[ J(M, \epsilon, \zeta, c, \mu) \leq \min \left\{ n, \left( \sqrt{\kappa} + \frac{1}{2} \right) \ln \left( \frac{16 (\sqrt{\kappa} + 1)^2 \kappa^4 (U_g + \mu \sqrt{n})^2}{\zeta^2 \mu^2} \right) \right\} \]
(24)
\[ = \min \left\{ n, \tilde{O}(\epsilon^{-1/2}) \right\}. \]
(25)
Therefore, in either case, we have that
\[ J(M, \epsilon, \zeta, c, \mu) \leq \min \left\{ n, \tilde{O}(\epsilon^{-1/2}) \right\}, \]
as claimed in Section 4.1.

The following theorem shows that when Algorithm 2 is terminated because of the test
\[ \| r^j \| > \sqrt{T \tau^j/2} \| r^0 \|, \]
then (16) will hold for some \( i = 0, 1, \ldots, j \), so that the outputs of Algorithm 2 are well defined.

**Theorem 6.** Suppose that the main loop of Algorithm 2 terminates with \( j = \hat{J} \), where
\[ \hat{J} \in \{ 1, \ldots, \min \{ n, J(M, \epsilon, \zeta, c, \mu) \} \}, \]
(where \( J(M, \epsilon, \zeta, c, \mu) \) is defined in Lemma 5) because the fourth termination test is satisfied and the three earlier conditions do not hold, that is,
\[ (y^j)^\top \bar{H} y^j \geq \epsilon \| y^j \|^2, \quad (p^j)^\top \bar{H} p^j \geq \epsilon \| p^j \|^2, \]
\[ \| r^j \| > \hat{c}, \quad \text{and/or} \quad \| r^j \|_{\infty} > c, \]
and
\[ \| r^j \| > \sqrt{T \tau^j/2} \| r^0 \| \]
(25)
where \( M, T, \) and \( \tau \) are the values returned by Algorithm 2. Then \( y^j+1 \) is computed by Algorithm 2, and we have
\[ \frac{(y^{j+1} - y^i)^\top \bar{H} (y^{j+1} - y^i)}{\| y^{j+1} - y^i \|^2} < \epsilon, \quad \text{for some} \ i = \{ 0, \ldots, \hat{J} - 1 \}. \]
(26)

**Proof.** This result follows directly from [25, Theorem 2] after noting that the properties of \( \hat{J} \) used in the proof do not depend on the definition of \( J(M, \epsilon, \zeta, c, \mu) \). In particular, \( \hat{J} \) simply needs to be an index such that (25) holds and the CG process has not stopped iterating before reaching \( \hat{J} \). Thus, the result holds once we account for the additional stopping criterion \( \| r^\mu \|_{\infty} \leq c \) in the new definition of \( J(M, \epsilon, \zeta, c, \mu) \).

We focus now on the main output of Algorithm 2, which is denoted by \( \hat{d}^k \) in Algorithm 1. The properties of \( \hat{d}^k \), which is obtained by scaling \( \hat{d}^k \), are essential to the first- and second-order complexity analysis of later sections.

**Lemma 7.** Let Assumptions 1 and 2 hold, and suppose that Algorithm 2 is invoked at iteration \( k \) of Algorithm 1. Let \( \hat{d}^k \) be the vector obtained in Algorithm 1 from the output \( \hat{d}^k \) of Algorithm 2. For each of the two possible settings of output flag \( d\text{\_type} \), we have the following.
1. When \( \texttt{d} \texttt{-type=SOL} \), the direction \( d_k \) satisfies

\[
\epsilon_H \|d_k\|^2 \leq (d_k)^\top (\tilde{X}_k \nabla^2 \phi_\mu(x^k) \tilde{X}_k + 2\epsilon_H I) d_k,
\]

\[
\|d_k\| \leq 1.1 \epsilon_H^{-1} \|\tilde{X}_k \nabla \phi_\mu(x^k)\|,
\]

\[
(d_k)^\top \tilde{X}_k \nabla \phi_\mu(x^k) = -\gamma_k (d_k)^\top (\tilde{X}_k \nabla^2 \phi_\mu(x^k) \tilde{X}_k + 2\epsilon_H I) d_k,
\]

where \( \gamma_k = \max \left\{ \frac{\|X_k^{-1} \tilde{X}_k d_k\|_\infty}{\beta}, 1 \right\} \). If \( \|X_k^{-1} \tilde{X}_k d_k\|_\infty \leq \beta \) holds, then \( d_k \) also satisfies

\[
\|\hat{r}_k\| \leq \frac{1}{2} \epsilon_H \zeta \|d_k\|,
\]

where \( \hat{r}_k \) is the residual of the scaled Newton system, defined by

\[
\hat{r}_k := (\tilde{X}_k \nabla^2 \phi_\mu(x^k) \tilde{X}_k + 2\epsilon_H I) \hat{d}_k + \tilde{X}_k \nabla \phi_\mu(x^k).
\]

2. When \( \texttt{d} \texttt{-type=NC} \), the direction \( d_k \) satisfies \( (d_k)^\top \tilde{X}_k \nabla \phi_\mu(x^k) \leq 0 \) and

\[
\frac{(d_k)^\top \tilde{X}_k \nabla^2 \phi_\mu(x^k) \tilde{X}_k d_k}{\|d_k\|^2} \leq -\|d_k\| \leq -\epsilon_H.
\]

**Proof.** For simplicity of notation, we use the following shorthand in the proof:

\[
H = \tilde{X}_k \nabla^2 \phi_\mu(x^k) \tilde{X}_k, \quad g = \tilde{X}_k \nabla \phi_\mu(x^k).
\]

Since Algorithm 1 invoked Algorithm 2, at least one of the conditions (6b) or (6c) must be violated at \( x^k \). Thus, by Lemma 4, we have \( \|g\| \geq \mu > 0 \), so the iterates of Algorithm 2 are well defined.

Consider first the case of \( \texttt{d} \texttt{-type=SOL} \). The bounds (27a) and (27b) follow by the same argument as in the first part of the proof of [25, Lemma 3]. We now prove (27c). The residual \( \hat{r}_k \) at the final iteration of CG procedure is orthogonal to all previous search directions, so that \( (d_k)^\top \hat{r}_k = 0 \) (see [25, Appendix A]). Since \( d_k \) and \( d_k \) are collinear, we have \( (d_k)^\top \hat{r}_k = 0 \), so from (29) it follows that

\[
(d_k)^\top g = -(d_k)^\top (H + 2\epsilon_H I) d_k.
\]

When \( \|X_k^{-1} \tilde{X}_k \hat{d}_k\|_\infty \leq \beta \), we have \( d_k = \hat{d}_k \), so

\[
(d_k)^\top g = -(d_k)^\top (H + 2\epsilon_H I) d_k = -(d_k)^\top (H + 2\epsilon_H I) d_k,
\]

proving (27c) in this case. When \( \|X_k^{-1} \tilde{X}_k \hat{d}_k\|_\infty > \beta \), we have

\[
d_k = \left( \frac{\beta}{\|X_k^{-1} \tilde{X}_k \hat{d}_k\|_\infty} \right) \hat{d}_k
\]

and thus

\[
(d_k)^\top g = -(d_k)^\top (H + 2\epsilon_H I) d_k = -\frac{\|X_k^{-1} \tilde{X}_k \hat{d}_k\|_\infty}{\beta} (d_k)^\top (H + 2\epsilon_H I) d_k,
\]

proving (27c) for this case as well.
Turning to (28), we note first that from termination conditions of Algorithm 2 that \(\|\bar{r}^k\| \leq \hat{\zeta}_r\|g\|\). Thus, using (29), we have that

\[
\|\bar{r}^k\| \leq \hat{\zeta}_r\|g\| \leq \hat{\zeta}_r \left(\|(H + 2\epsilon_H I)\dd^k\| + \|\bar{r}^k\|\right) \leq \hat{\zeta}_r \left(\|(M + 2\epsilon_H)\dd^k\| + \|\bar{r}^k\|\right),
\]

where \(M\) is the value that is returned by Algorithm 2, so that

\[
\|\bar{r}^k\| \leq \frac{\hat{\zeta}_r}{1 - \hat{\zeta}_r} (M + 2\epsilon_H)\|\dd^k\|.
\]

Using again that \(\hat{\zeta}_r = \zeta_r/(3\kappa) < 1/6\) and the definition of \(\hat{\zeta}_r\) in Algorithm 2,

\[
\frac{\hat{\zeta}_r}{1 - \hat{\zeta}_r} (M + 2\epsilon_H) \leq \frac{6}{5} \hat{\zeta}_r (M + 2\epsilon_H) = \frac{6}{5} \frac{\zeta_r\epsilon_H}{3} < \frac{1}{2} \zeta_r\epsilon_H,
\]

which yields (28) when we note that \(d^k = \dd^k\) when \(\|X_k^{-1}X_k\dd^k\|\infty \leq \beta\).

In the case of \(d\text{type}=\text{NC}\), we recall that Algorithm 1 defines

\[
d^k = -\text{sgn}(g^T \dd^k) \min \left\{ \frac{\|\dd^k||H\dd^k\|}{\|\dd^k\|^2} \beta, \frac{\beta}{\|X_k^{-1}X_k\dd^k\|\infty} \right\} \dd^k.
\]

We have from positivity of the ratios in the \(\min\{., .\}\) expression that

\[
\text{sgn}(g^T d_k) = -\text{sgn}(g^T \dd^k)^2 = -1,
\]

so that \(g^T d_k \leq 0\). Next, since \(d_k^k\) and \(d_k^k\) are collinear, we have

\[
\frac{(d_k^k)^T (H + 2\epsilon_H I)(d_k^k)}{\|d_k^k\|^2} = \frac{(\dd^k)^T (H + 2\epsilon_H I)(\dd^k)}{\|\dd^k\|^2} \leq \epsilon_H,
\]

so that

\[
\frac{(d_k^k)^T H(d_k^k)}{\|d_k^k\|^2} \leq -\epsilon_H.
\]

When the min in (32) is achieved by the first term, we have

\[
\|d^k\| = \|\dd^k\|/\|\dd^k\|^2 \geq \epsilon_H,
\]

proving (30) in this case. Otherwise, when the min in (32) is achieved by the second term, we have

\[
\beta = \|X_k^{-1}X_k d^k\|\infty \leq \|X_k^{-1}X_k d^k\| \leq \|X_k^{-1}X_k\|\|d^k\| \leq \|d^k\|.
\]

Using this bound, along with (33) and the fact that \(\beta \geq \epsilon_H\) (by definition), we have

\[
\|d^k\| \geq \min \left\{ \frac{\|\dd^k||H\dd^k\|}{\|\dd^k\|^2} \beta, \frac{\beta}{\|X_k^{-1}X_k\dd^k\|\infty} \right\} \geq \min \{\epsilon_H, \beta\} = \epsilon_H.
\]

In either case of the min in (32), we have \(\|d^k\| \leq (\dd^k)^T H\dd^k/\|\dd^k\|^2\), so that

\[
\frac{(d_k^k)^T Hd_k^k}{\|d_k^k\|^2} \leq -\|d_k^k\| \leq -\epsilon_H,
\]

proving (30). \(\square\)
5.2 First-Order Complexity Analysis We now derive a worst-case complexity result for the first-order optimality conditions (6a), (6b), and (6c). We show that when Algorithm 2 returns $d_{\text{type}} = \text{SOL}$ and a unit step is taken by the line search procedure in Algorithm 1 (that is, $\alpha_k = 1$), either the first-order optimality conditions hold at $x^{k+1}$, or else $\|d^k\|$ is large enough to make significant progress in reducing the function $\phi_{\mu}$. Theorem 12 and Corollary 13 state first-order complexity results in terms of the number of iterations of Algorithm 1 and the number of gradient evaluations and/or Hessian vector products, respectively.

Our results depend on the following technical result concerning the decrease of the log-barrier term in $\phi_{\mu}$. Its proof can be found in Appendix A.2.

**Lemma 8.** Given $x > 0$, define $X$, $\bar{X}$ as in (5), and suppose that $d \in \mathbb{R}^n$ is such that $\|X^{-1}\bar{X}d\|_\infty \leq \beta < 1$. Then,

$$
- \sum_{i=1}^{n} \log(x_i + \bar{x}_id_i) + \sum_{i=1}^{n} \log(x_i) 
\leq -e^T X^{-1} \bar{X} d + \frac{1}{2} d^T \bar{X} X^{-2} \bar{X} d + \frac{2 - \beta}{6(1 - \beta)^2} \|d\|^2.
$$

Our first result deals with the case in which a full step ($\alpha_k = 1$) is taken in Algorithm 1.

**Lemma 9.** Let Assumptions 1 and 2 hold, and suppose that Algorithm 2 is invoked at an iterate $x^k$ of Algorithm 1, and returns $d_{\text{type}} = \text{SOL}$. Then, when the unit step is taken (that is, $x^{k+1} = x^k + \bar{X}d^k$), we have either

$$
\|d^k\| \geq c_d \epsilon_H, \quad \text{where } c_d = \min \left\{ \frac{1 - \bar{\zeta}}{9}, \left( \frac{3}{2L_H} \right)^{1/2}, \frac{1}{2(L_H + 9/2 + \bar{\zeta})} \right\}
$$

or else we have both

$$
\nabla f(x^{k+1}) \geq -\epsilon_g e \quad \text{and} \quad \|\bar{X}_{k+1} \nabla f(x^{k+1})\|_\infty \leq \epsilon_g.
$$

**Proof.** We begin by noting that if the output $\bar{d}^k$ from Algorithm 2 satisfies $\|X_k^{-1} \bar{X}_kd^k\|_\infty \geq \beta$ then

$$
\epsilon_H \leq \beta = \|X_k^{-1} \bar{X}_kd^k\|_\infty \leq \|X_k^{-1} \bar{X}_kd^k\| \leq \|X_k^{-1} \bar{X}_k\| \|d^k\| \leq \|d^k\|,
$$

so the claim (35) holds, since $c_d \leq 1$. Thus, we assume for the remainder of the proof that $\|X_k^{-1} \bar{X}_kd^k\|_\infty < \beta$ and $d^k = \bar{d}^k$, and that $\|d^k\| < c_d \epsilon_H$. We show that the conditions (36) hold in this case.

We start by establishing that $\nabla f(x^{k+1}) \geq -\epsilon_g e$. Since $d_{\text{type}} = \text{SOL}$, we have that $\zeta \mu \geq \|\bar{r}^k\|_\infty$ where $\bar{r}^k$ is defined in (29). Using $\|\bar{X}_k X^{-2} \bar{X}_k\|_1 \leq 1$ and $\epsilon_H \|d^k\| < c_d \epsilon_H$, it follows that

$$
\zeta \mu \geq \|\bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k + 2 \epsilon_H I \|d^k + \bar{X}_k \nabla \phi_\mu(x^k)\|_\infty
\geq \|\bar{X}_k \nabla^2 f(x^k) \bar{X}_k d^k + \nabla \phi_\mu(x^k)\|_\infty - \mu \|\bar{X}_k X^{-2} \bar{X}_k d^k\|_\infty - 2 \epsilon_H \|d^k\|_\infty
\geq \|\bar{X}_k \nabla^2 f(x^k) \bar{X}_k d^k + \nabla \phi_\mu(x^k)\|_\infty - \mu \|\bar{X}_k X^{-2} \bar{X}_k\| \|d^k\| - 2 \epsilon_H \|d^k\|
\geq \|\bar{X}_k \nabla^2 f(x^k) \bar{X}_k d^k + \nabla \phi_\mu(x^k)\|_\infty - \mu \|d^k\| - 2 \epsilon_H \|d^k\|
\geq \|\bar{X}_k \nabla^2 f(x^k) \bar{X}_k d^k + \nabla \phi_\mu(x^k)\|_\infty - \mu \|d^k\| - 2 \epsilon_H \|d^k\|
\geq \|\bar{X}_k \nabla^2 f(x^k) \bar{X}_k d^k + \nabla \phi_\mu(x^k)\|_\infty - \mu \epsilon_H \mu - 2 c_d \epsilon_g.
$$

16
Since $\epsilon_H < 1$ and $\mu = \epsilon_g/4$, we have
\[
\tilde{\epsilon} + c_d \epsilon_H \mu + 2c_d \epsilon_g \leq \tilde{\epsilon} \mu + c_d \mu + 2c_d \epsilon_g = \mu (\tilde{\epsilon} + 9c_d).
\]
Then, by the definition of $c_d$, $\tilde{\epsilon} + 9c_d \leq 1$ so that $\tilde{\epsilon} \mu + c_d \epsilon_H \mu + 2c_d \epsilon_g \leq \mu$. Thus, by substituting into (37), we obtain
\[
\mu > \| \bar{X}_k (\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k)) - \mu \bar{X}_k X_k^{-1} e \|_\infty.
\]
By considering each component $i = 1, 2, \ldots, n$ in turn, we will show that
\[
\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k) > -\mu e.
\]
When $0 < x_i^k \leq 1$, it follows that $\bar{x}_i^k / x_i^k = 1$, so
\[
[\bar{X}_k (\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k))]_i - \mu | < \mu
\]
so that
\[
[\bar{X}_k (\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k))]_i > 0,
\]
establishing (39) for this component $i$. When $x_i^k > 1$, we have $\bar{x}_i^k = 1$ and $0 < \bar{x}_i^k / x_i^k < 1$, so from (38), we have
\[
-\mu < [\bar{X}_k (\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k))]_i - \frac{\bar{x}_i^k}{x_i^k} \mu < [\bar{X}_k (\nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k))]_i
\]
establishing (39) for this component too.

Finally, using (3), $\mu = \epsilon_g/4$, $\| d^k \| < c_d \epsilon_H$, $c_d \leq \sqrt{3/(2L_H)}$, and $\epsilon^2_H = \epsilon_g$, together with $\| \bar{X}_k \| \leq 1$, we have from (39) that
\[
\nabla f(x^{k+1}) = \nabla f(x^{k+1}) - \nabla^2 f(x^k) \bar{X}_k d^k - \nabla f(x^k) + \nabla^2 f(x^k) \bar{X}_k d^k + \nabla f(x^k)
\]
\[
> - \| \nabla f(x^{k+1}) - \nabla^2 f(x^k) \bar{X}_k d^k - \nabla f(x^k) \| e - \mu e
\]
\[
\geq - \frac{L_H}{2} \| \bar{X}_k \|^2 \| d^k \|^2 e - \mu e
\]
\[
> - \left( \frac{L_H}{2} c_d^2 + \frac{1}{4} \right) \epsilon_g e \geq -\epsilon_g e.
\]

We now focus on the second condition, $\| \bar{X}_k \nabla f(x^{k+1}) \| \leq \epsilon_g$. To begin, we show that
\[
\| \bar{X}_{k+1} \nabla f(x^{k+1}) \| \leq 2 \| \bar{X}_k \nabla f(x^{k+1}) \|_\infty.
\]
First, assume that $x_i^k \leq 1$ holds. Then, $\bar{x}_i^k = x_i^k$ so that $d_i^k = (\bar{x}_i^k / x_i^k) d_i^k \leq 1 < 1$, so
\[
\bar{x}_i^{k+1} \leq x_i^{k+1} = x_i^k + \bar{x}_i^k d_i^k = x_i^k (1 + d_i^k) < 2x_i^k.
\]
When $x_i^k > 1$, we have
\[
\bar{x}_i^{k+1} \leq 1 = x_i^k < 2x_i^k.
\]
Applying these two cases for each coordinate $i$, we obtain (40). Now, recall from the conditions stated at the start of the proof that $\| X_k^{-1} \bar{X}_k d^k \|_\infty < \beta$, so that $d^k = d^k$, where $\tilde{d}^k$ is the output of Algorithm 2 at iteration $k$. We thus have for $i^k$ defined
by (29) that (28) holds, by Lemma 7. Therefore, by (3), (28), \( \|X_k^{-1}e\|_\infty \leq 1 \), and \( \|X_k\| \leq 1 \), we have

\[
\|X_{k+1} \nabla f(x^{k+1})\|_\infty \\
\leq 2 \|X_k \nabla f(x^{k+1})\|_\infty \\
= 2 \|X_k \nabla f(x^{k+1}) - X_k \nabla f(x^k) + X_k \nabla f(x^k)\|_\infty \\
= 2 \|X_k \nabla f(x^{k+1}) - X_k \nabla f(x^k) - X_k \nabla^2 f(x^k) X_k d^k \\
- 2\epsilon \|X_k^{-1}e\|_\infty + \epsilon \|d^k\|_\infty \\
\leq 2 \|X_k \nabla f(x^{k+1}) - \nabla f(x^{k}) - \nabla^2 f(x^{k}) X_k d^k\|_\infty \\
+ 2\mu \|X_k X_k^{-1} e\|_\infty + 4\epsilon H \|d^k\|_\infty \\
\leq 2 \|X_k \nabla f(x^{k+1}) - \nabla f(x^{k}) - \nabla^2 f(x^{k}) X_k d^k\|_\infty \\
+ 2\mu \|X_k X_k^{-1} e\|_\infty + 4\epsilon H \|d^k\|_\infty + 2\epsilon \|d^k\|_\infty \\
\leq L_H \|X_k d^k\|_2^2 + 2\mu \|X_k X_k^{-2} X_k d^k\|_\infty + 4\epsilon H \|d^k\|_\infty + 2\mu + 2\epsilon \|d^k\|_\infty \\
\leq L_H \|X_k d^k\|_2^2 + 2\mu \|X_k X_k^{-2} X_k d^k\|_\infty + 4\epsilon H \|d^k\|_\infty + 2\mu + 2\epsilon \|d^k\|_\infty \\
< L_H c_d^2 \epsilon_g + 2\epsilon c_d e_H + 4\epsilon c_d \epsilon_g + \epsilon_g/2 + \epsilon_c c_d \epsilon_g,
\]

where we used \( \|X_k\| \leq 1 \), \( \|X_k^{-1}X_k\| \leq 1 \), \( \|d^k\| \leq c_d e_H, \epsilon^2_H = \epsilon_g \), and \( \mu = \epsilon_g/4 \) for the last inequality. Finally, since \( \epsilon_H < 1 \), \( c_d \leq 1 \), and \( c_d \leq 1/(2(L_H + 9/2 + \epsilon_c)) \), it follows that

\[
\|X_{k+1} \nabla f(x^{k+1})\|_\infty < L_H c_d^2 \epsilon_g + 2\epsilon c_d e_H + 4\epsilon c_d \epsilon_g + \epsilon_g/2 + \epsilon_c c_d \epsilon_g \\
\leq L_H c_d \epsilon_g + 2\epsilon c_d \epsilon_g + 4\epsilon c_d \epsilon_g + \epsilon_g/2 + \epsilon_c c_d \epsilon_g \\
\leq L_H c_d \epsilon_g + c_d \epsilon_g/2 + 4\epsilon c_d \epsilon_g + \epsilon_g/2 + \epsilon_c c_d \epsilon_g \\
\leq c_d \epsilon_g (L_H + 9/2 + \epsilon_c) + \epsilon_g/2 \\
\leq \epsilon_g/2 + \epsilon_g/2 = \epsilon_g,
\]

completing the proof. \( \square \)

Lemma 9 is useful in the following line search argument, because we need only consider cases in which \( \|d^k\| \geq c_d e_H \). We now show that sufficient decrease can be attained in \( \phi_\mu \) whenever \( d_{\text{type}} = \text{SOL} \) and \( x^{k+1} \) does not satisfy the approximate first-order conditions (6b) and (6c).

**Lemma 10.** Suppose that Assumptions 1 and 2 hold. Suppose that at iteration \( k \) of Algorithm 1, we have either \( |\nabla f(x^k)|_i \leq -\epsilon_g \) for some coordinate \( i \) or \( \|X_k \nabla f(x_k)\|_\infty \geq \epsilon_g \), so that Algorithm 2 is called. When Algorithm 2 outputs a direction \( d^k \) with \( d_{\text{type}} = \text{SOL} \), then either

(i) the backtracking line search terminates with \( \alpha_k = 1 \) and both (6b) and (6c) hold at \( x^{k+1} \), or

(ii) the backtracking line search requires at most \( j_k \leq j_{\text{sol}} + 1 \) iterations, where

\[
\begin{align*}
J_{\text{sol}} &= \left[ -\frac{1}{2} \log_\theta \left( \frac{6(1 - \beta)^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \right) \right]^{+}, \\
\Phi_\mu(x^k) - \Phi_\mu(x^{k+1}) &\geq c_{\text{sol}} c_H^3,
\end{align*}
\]

and the resulting step \( x^{k+1} = x^k + \alpha_k X_k d^k \) satisfies
where
\[
c_{\text{sol}} = \frac{\eta}{6} \min \left\{ c_d^3, \left[ \frac{6(1-\beta)^2}{(L_H + \eta)(1-\beta)^2 + (2-\beta)} \right]^3 \right\}
\]
and \( c_d \) is defined in (35).

**Proof.** This result follows by largely the same argument as that of the proof of [26, Lemma 13]. The main difference is due to the result of Lemma 8 which, together with (4), implies
\[
\phi_{\mu}(x^k + \theta^j \bar{X}_k d^k) - \phi_{\mu}(x^k) \leq \theta^j g^T d^k + \frac{\theta^j}{2} (d^k)^T H d^k + \frac{L_H(1-\beta)^2 + (2-\beta)}{6(1-\beta)^2} \theta^j \|d^k\|^3,
\]
where the notation \( g = \bar{X}_k \nabla \phi_{\mu}(x^k) \) and \( H = \bar{X}_k \nabla^2 \phi_{\mu}(x^k) \bar{X}_k \) is used once more. Replacing the Taylor series expansion around \( f \) in the proof of [26, Lemma 13] with this expression yields the result. We provide a full proof in Appendix A.3 for completeness.

Now we show that sufficient decrease occurs at any step where \( d_{\text{type}}=\text{NC} \).

**Lemma 11.** Suppose that Assumptions 1 and 2 hold. Suppose that at iteration \( k \) of Algorithm 1, we have either \( \|\nabla f(x^k)\|_i \leq -\epsilon_g \) for some coordinate \( i \) or \( \|\bar{X}_k \nabla f(x^k)\|_\infty \geq \epsilon_g \), so that Algorithm 2 is called. When Algorithm 2 outputs a direction \( \hat{d}^k \) with \( d_{\text{type}}=\text{NC} \), then the backtracking line search requires at most \( j_k \leq j_{nc} + 1 \) iterations, where
\[
j_{nc} = \left\lceil \log_{\theta} \left( \frac{3(1-\beta)^2}{(L_H + \eta)(1-\beta)^2 + (2-\beta)} \right) \right\rceil,
\]
and the resulting step \( x^{k+1} = x^k + \alpha_k \bar{X}_k d^k \) satisfies
\[
\phi_{\mu}(x^k) - \phi_{\mu}(x^{k+1}) \geq c_{nc} \epsilon_H,
\]
where
\[
c_{nc} = \frac{\eta}{6} \min \left\{ 1, \left[ \frac{3(1-\beta)^2}{(L_H + \eta)(1-\beta)^2 + (2-\beta)} \right]^3 \right\}.
\]

**Proof.** This result follows from the same argument as the proof of [26, Lemma 1]. The main difference in the proof once again revolves around the use of (43) in place of the Taylor expansion around \( f \). A full proof is provided in Appendix A.4 for completeness.

Now we are ready to bound the maximum number of iterations of Algorithm 1 that can occur before the approximate first-order optimality conditions are satisfied.

**Theorem 12.** Let Assumptions 1 and 2, hold. Then, defining
\[
\bar{K}_1 := \left\lceil \frac{\phi_{\mu}(x^0) - \phi_{\text{low}}}{\min \{c_{\text{sol}}, c_{nc}\} \epsilon_g} - 3/2 \right\rceil,
\]
some iterate \( x^k, k = 0, 1, \ldots, \bar{K}_1 + 1 \) generated by Algorithm 1 will satisfy
\[
\begin{align*}
(46a) \quad & x^k > 0 \\
(46b) \quad & \nabla f(x^k) \geq -\epsilon_g e \\
(46c) \quad & \|\bar{X}_k \nabla f(x^k)\|_\infty \leq \epsilon_g.
\end{align*}
\]
Proof. Suppose for contradiction that at least one of (46b) and (46c) is violated for all \( k = 0, 1, \ldots, \bar{K}_1 + 1 \), so that case A of Lemma 10 does not occur for all \( k = 0, 1, \ldots, \bar{K}_1 \). Then, for each \( l = 0, 1, \ldots, \bar{K}_1 \), the results of Lemma 10 and Lemma 11 guarantee that

\[
\phi_{\mu}(x^l) - \phi_{\mu}(x^{l+1}) \geq \min \{ c_{\text{sol}}, c_{\text{nc}} \} \epsilon_g^{3/2}
\]

holds, by recalling that \( \epsilon_g = \epsilon_g^{1/2} \). The contradiction follows from the same argument as [25, Theorem 3] with the replacement of \( f \) with \( \phi_{\mu} \) and \( f_{\text{low}} \) with \( \phi_{\text{low}} \).

Recalling that the workload of Algorithm 2 in terms of Hessian-vector products depends on the index \( J \) defined by the maximum of (23) and (24), we obtain the following corollary. (Note the mild assumption on the value of \( M \) used at each instance of Algorithm 2, which is satisfied provided that this algorithm is always invoked with an initial estimate of \( M \) in the range \([0, \bar{U}_H + \mu]\).)

**Corollary 13.** Suppose that the assumptions of Theorem 12 are satisfied, and let \( \bar{K}_1 \) be as defined in that theorem and \( J(M, \epsilon_H, \zeta_r, c_{\mu}) \) be as defined in (24). Suppose that the values of \( M \) used or calculated at each instance of Algorithm 2 satisfy \( M \leq \bar{U}_H + \mu \). Then the number of Hessian-vector products and/or gradient evaluations required by Algorithm 1 to output an iterate satisfying (46) is at most

\[
(2 \min \{ n, J(U_H + \mu, \epsilon_H, \zeta_r, c_{\mu}) \} + 2)(\bar{K}_1 + 1).
\]

For \( n \) sufficiently large, this bound is \( \tilde{O}\left(\epsilon_g^{-7/4}\right) \), while if

\[
J(U_H + \mu, \epsilon_H, \zeta_r, c_{\mu}) \geq n, \text{ the bound is } \tilde{O}\left(n \epsilon_g^{-3/2}\right).
\]

Proof. The result follows directly from the proof of [25, Corollary 1] using our new definition of \( J(U_H + \mu, \epsilon_H, \zeta_r, c_{\mu}) \) and \( \bar{K}_1 \). We give details for completeness. From Lemma 5, the number of Hessian-vector multiplications in the main loop of Algorithm 2 is bounded by \( \min \{ n, J(U_H, \epsilon_H, \zeta_r, c_{\mu}) \} + 1 \). An additional \( \min \{ n, J(U_H, \epsilon_H, \zeta_r, c_{\mu}) \} \) Hessian-vector products may be needed to return a direction satisfying (16), if Algorithm 2 does not store its iterates \( y_j \). Each iteration also requires a single evaluation of the gradient \( \nabla f \), giving a bound of \( (2 \min \{ n, J(U_H, \epsilon_H, \zeta_r, c_{\mu}) \} + 2) \) on the workload per iteration of Algorithm 1. We multiply this quantity by the iteration bound from Theorem 12 to obtain the result.

5.3 Second-Order Complexity Analysis We now find bounds on iteration and computational complexity of finding a point that satisfies all of the approximate optimality conditions in (6). In this section, as well as using results from Sections 5.1 and 5.2, we need to use the properties of the minimum eigenvalue oracle, Procedure 3. To this end, we make the following generic assumption.

**Assumption 3.** For every iteration \( k \) at which Algorithm 1 calls Procedure 3, and for a specified failure probability \( \delta \) with \( 0 \leq \delta \ll 1 \), Procedure 3 either certifies that \( X_k \nabla^2 f(x_k) X_k \succeq -\epsilon_H I \) or finds a vector of curvature smaller than \(-\epsilon_H/2\) in at most

\[
N_{\text{neq}} := \min \left\{ n, 1 + \left[ c_{\text{neq}} \epsilon_H^{-1/2} \right] \right\}
\]

Hessian-vector products, with probability \( 1 - \delta \), where \( c_{\text{neq}} \) depends at most logarithmically on \( \delta \) and \( \epsilon_H \).
Assumption 3 encompasses the strategies we mentioned in Section 4.2. Assuming the bound $U_H$ on $||H||$ is available, for both the Lanczos method with a random starting vector and the conjugate gradient algorithm with a random right-hand side, (47) holds with $C_{mcu} = \ln(2.76n/\delta^2) \sqrt{U_H}/2$. When a bound on $||H||$ is not available in advance, it can be estimated efficiently with minimal effect on the overall complexity of the method, see Appendix B.3 of [25].

The next lemma guarantees termination of the backtracking line search for a negative curvature direction. As for Lemma 10, the result is deterministic.

**Lemma 14.** Suppose that Assumptions 1, 2, and 3 hold. Suppose that at iteration $k$ of Algorithm 1, the search direction $d^k$ is of negative curvature type, obtained either directly from Procedure 3 or as the output of Algorithm 2 with $d_{type} = NC$. Then the backtracking line search terminates with step length $\alpha_k = \theta^j_k$ with $j_k \leq j_{nc} + 1$, where $j_{nc}$ is defined as in Lemma 11, and the decrease in the function value resulting from the chosen step length satisfies

$$\phi_\mu(x^k) - \phi_\mu(x^k + \alpha_k \bar{X}_k d^k) \geq \frac{c_{nc}}{64} \frac{\beta}{\eta},$$

with $c_{nc}$ is defined in Lemma 11.

**Proof.** Lemma 11 shows that the claim holds (with a factor of 1/64 to spare) when the direction of negative curvature is obtained from Algorithm 2. When the direction $v$ is obtained from Procedure 3, we have by $||v|| = 1$ that

$$v^\top \bar{X}_k \nabla^2 f(x^k) \bar{X}_k v \leq -\frac{1}{2} \epsilon_H.$$

Then, since $v^\top \bar{X}_k X_k^{-2} \bar{X}_k v \leq 1$, we have

$$v^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k v = v^\top \bar{X}_k \nabla^2 f(x^k) \bar{X}_k v + \mu v^\top \bar{X}_k X_k^{-2} \bar{X}_k v$$

$$\leq -\frac{1}{2} \epsilon_H + \mu \leq -\frac{1}{4} \epsilon_H,$$

where the last inequality follows from $\mu = \epsilon_g/4 = \epsilon_H^2/4$ and $\epsilon_H < 1$. Now, when

$$\min \left\{ \frac{|v^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k v|}{\|X_k^{-1} \bar{X}_k v\|_\infty}, \frac{\beta}{\|X_k^{-1} \bar{X}_k d^k\|_\infty} \right\} = |v^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k v|,$$

we have $\|d^k\| = |v^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k v| \geq \epsilon_H/4$. Otherwise, we have

$$\beta = \|X_k^{-1} \bar{X}_k d^k\|_\infty \leq \|X_k^{-1} \bar{X}_k d^k\| \leq \|X_k^{-1} \bar{X}_k\| \|d^k\| \leq \|d^k\|.$$

By combining the two cases, and using $\beta \geq \epsilon_H$, we have

$$\|d^k\| \geq \min \left\{ \frac{1}{4} \epsilon_H, \beta \right\} = \frac{1}{4} \epsilon_H.$$

Finally, we note that in either case, we have

$$\|d^k\| \leq -v^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k v = -\frac{(d^k)^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k d^k}{\|d^k\|^2}.$$

Therefore, we have

$$\frac{(d^k)^\top \bar{X}_k \nabla^2 \phi_\mu(x^k) \bar{X}_k d^k}{\|d^k\|^2} \leq -\|d^k\| \leq -\frac{1}{4} \epsilon_H.$$
The result can now be obtained by following the proof of Lemma 11, with \( \frac{1}{2} \epsilon_H \) replacing \( \epsilon_H \). \( \square \)

We are now ready to state our iteration complexity result for Algorithm 1.

**THEOREM 15.** Suppose that Assumptions 1, 2, and 3 hold and define

\[
K_2 := \left\lceil \frac{3(\phi_{\mu}(x^0) - \phi_{\text{low}})}{\min\{c_{\text{sol}}, c_{\text{nc}}/64\} \epsilon_g^{-3/2}} \right\rceil + 2,
\]

where constants \( c_{\text{sol}} \) and \( c_{\text{nc}} \) are defined in Lemmas 10 and 11, respectively. Then with probability at least \( (1 - \delta)^{K_2} \), Algorithm 1 terminates at a point satisfying (6) at most \( K_2 \) iterations. (With probability at most \( 1 - (1 - \delta)^{K_2} \), it terminates incorrectly within \( K_2 \) iterations at a point for which (6a), (6b), and (6c) hold but (6d) does not.)

**Proof.** Algorithm 1 terminates incorrectly with probability \( \delta \) at any iteration at which Procedure 3 is called, when Procedure 3 certifies erroneously that \( \lambda_{\min}(\nabla^2 f(x^k) \bar{X}_k) \geq -\epsilon_H \). Such an erroneous certificate only leads to termination. Therefore, an erroneous certificate at iteration \( k \) means that Procedure 3 did not produce an erroneous certificate at iterations 0 to \( k - 1 \). By a disjunction argument, we have that the overall probability of terminating with an erroneous certificate during the first \( K_2 \) iterations is bounded by \( 1 - (1 - \delta)^{K_2} \). Therefore, with probability at least \( (1 - \delta)^{K_2} \), no incorrect termination occurs in the first \( K_2 \) iterations.

Suppose now for contradiction that Algorithm 1 runs for \( K_2 \) iterations without terminating. That is, for all \( l = 0, 1, \ldots, K_2 \), we have at least one of: \( [\nabla f(x^l)]_i < -\epsilon_g \) for some coordinate \( i \), \( \|\bar{X}_l \nabla f(x^l)\|_{\infty} > \epsilon_g \), or \( \lambda_{\min}(\bar{X}_l \nabla^2 f(x^l) \bar{X}_l) < -\epsilon_H \). We perform the following partition of the set of iteration indices:

\[
K_1 \cup K_2 \cup K_3 = \{0, 1, \ldots, K_2 - 1\},
\]

where \( K_1 \), \( K_2 \), and \( K_3 \) are defined as follows.

**Case 1:** \( K_1 := \{l = 0, 1, \ldots, K_2 - 1 : \nabla f(x^l) \geq -\epsilon_g \text{ and } \|\bar{X}_l \nabla f(x^l)\|_{\infty} \leq \epsilon_g \} \).

**Case 2:** \( K_2 := \{l = 0, 1, \ldots, K_2 - 1 : [\nabla f(x^l)]_i < -\epsilon_g \text{ for some coordinate } i \text{ and/or } \|\bar{X}_l \nabla f(x^l)\|_{\infty} > \epsilon_g \text{ and } \|\nabla f(x^{l+1})\|_j < -\epsilon_g \text{ for some coordinate } j \text{ and/or } \|\bar{X}_{l+1} \nabla f(x^{l+1})\|_{\infty} > \epsilon_g \} \).

**Case 3:** \( K_3 := \{l = 0, 1, \ldots, K_2 - 1 : \|\bar{X}_l \nabla f(x^l)\|_{\infty} > \epsilon_g \text{ and } \nabla f(x^{l+1}) \geq -\epsilon_g \text{ and } \|\bar{X}_{l+1} \nabla f(x^{l+1})\|_{\infty} \leq \epsilon_g \} \).

Then, for all \( l \in K_1 \cup K_2 \), the results of Lemma 10, Lemma 11, and Lemma 14 guarantee that

\[
\phi_{\mu}(x^l) - \phi_{\mu}(x^{l+1}) \geq \min\{c_{\text{sol}}, c_{\text{nc}}/64\} \gamma^{3/2}
\]

holds. On the other hand, for \( l \in K_3 \), case A of Lemma 10 may occur, so sufficient decrease may not be ensured. Thus, cases 1, 2, and 3 coincide exactly with the three cases from the proof of [25, Theorem 4]. Following the same argument using these updated cases, constants, and substitution of \( \phi_{\mu} \) for \( f \) leads to the required contradiction.

\( \square \)

Finally, we provide an operation complexity result, a bound on the number of Hessian-vector products and gradient evaluations necessary for Algorithm 1 to find a point that satisfies (6).

**COROLLARY 16.** Suppose that assumptions of Theorem 15 hold, and let \( K_2 \) be defined as in (50). Suppose that the values of \( M \) used or calculated at each instance of Algorithm 2 satisfy \( M \leq U_H + \mu \). Then with probability at least \( (1 - \delta)^{K_2} \), Algorithm 1
terminates at a point satisfying (6) after at most

\[(52) \quad \text{max}\{2 \min\{n, J(U_H + \mu, \epsilon_H, \zeta_r, \epsilon_r, \mu)\} + 2, N_{\text{meo}}\} \tilde{K}_2\]

Hessian-vector products and/or gradient evaluations. (With probability at most \(1 - (1 - \delta)^{\tilde{K}_2}\), it terminates incorrectly with this complexity at a point for which for which (6a), (6b), and (6c) hold but (6d) does not.)

For \(n\) sufficiently large, the operation bound (52) is \(\tilde{O}\left(\epsilon_g^{-7/4}\right)\).

Proof. The proof follows by combining Theorem 15 (which bounds the number of iterations) with Lemma 5 and Assumption 3 (which bound the workload per iteration).

This complexity result matches the bound for unconstrained methods found in [1, 7, 8, 21, 25].

6 Discussion We have presented a log-barrier Newton-CG algorithm which combines recent advances in complexity of algorithms for large-scale unconstrained optimization with results on the primal log-barrier function for bound constraints. Our algorithm uses the Capped CG method of [25] to compute Newton-type steps for the log-barrier function, while monitoring convexity during the CG iterations to detect directions of negative curvature, when they exist. Once the algorithm has found a point satisfying the first-order optimality conditions, a Minimum Eigenvalue Oracle is used to find a direction of negative curvature for the scaled Hessian matrix or to certify (with high probability) that the second-order optimality conditions hold at the current iterate. Both types of steps can be computed using efficient iterative solvers, enabling good overall computational complexity results. The resulting method finds a point satisfying (6) in at most \(O\left(\epsilon_g^{-3/2}\right)\) iterations and at most \(\tilde{O}\left(\min\left\{n\epsilon_g^{-3/2}, \epsilon_g^{-7/4}\right\}\right)\) gradient evaluations and/or Hessian vector products. This overall computational complexity matches the best known results for twice Lipschitz continuously differentiable functions in unconstrained optimization. We have shown that essentially the same complexity holds for the bound-constrained case.

There are a number of ways to align our algorithm more closely with the interior-point methods in common use. One possible extension is to embed this method in a primal-dual interior-point framework, which is more popular than the primal log-barrier framework. A second is to extend to the more usual log-barrier approach of approximately minimizing \(\phi_\mu\) for a decreasing positive sequence of values of \(\mu\), rather than the “one-shot” approach using a small fixed value of \(\mu\) that we favor in this paper. Finally, generalizations of our approach to problems with more complex constraint sets, such as problems with general linear constraints, remains an open problem.

Acknowledgement We are most grateful to Clément Royer for his valuable advice and many suggestions during the preparation of this manuscript.

REFERENCES


Appendix A. Proofs of Technical Results.

A.1 Proof of Theorem 1. Proof. This proof follows similar logic to the proof
of [20, Theorem 1]. Let \( \delta > 0 \) be small enough such that

\[
\min_x f(x) + \frac{1}{4}\|x - x^*\|^4, \quad \text{s.t.} \quad x \geq 0, \quad \|x - x^*\|^2 \leq \delta,
\]

has \( x^* \) as its unique global solution. Now consider the application of the classical sequential log barrier method where, for each \( k \), we consider the solution of

\[
\min_x \psi_k(x) := f(x) + \frac{1}{4}\|x - x^*\|^4 - \mu_k \log(x_i), \quad \text{s.t.} \quad \|x - x^*\|^2 \leq \delta,
\]

for a positive sequence \( \{\mu_k\} \) with \( \mu_k \rightarrow 0 \). For each \( k \), a global solution \( x^k \) exists and all cluster points of \( \{x^k\} \) are global solutions of (53) [18]. By the ball constraint on \( x \), it follows that \( \{x^k\} \) is bounded so \( x^k \rightarrow x^* \), which implies (10a).

For sufficiently large \( k \), \( x^k \) is a local solution of the unconstrained problem

\[
\min_x \psi_k(x) = f(x) + \frac{1}{4}\|x - x^*\|^4 - \mu_k \sum_{i=1}^n \log(x_i).
\]

It follows that

\[
0 = \nabla \psi_k(x^k) = \nabla f(x^k) + \|x^k - x^*\|^2 (x^k - x^*) - \sum_{i=1}^n \frac{\mu_k}{x_i^k} e_i.
\]

By choosing \( s^k_i = \mu_k/x^k_i \) for \( i = 1, 2, \ldots, n \), we have (10b). Under this choice of \( s^k_i \), we have

\[
\min\{x^k_i, 1\} s^k_i = \frac{\mu_k \min\{x^k_i, 1\}}{x^k_i} \leq \mu_k \rightarrow 0,
\]

so that (10c) also holds.

Now for the second-order condition, we have

\[
d^\top \left( \nabla^2 f(x^k) + \sum_{i=1}^n \frac{\mu_k}{(x^k_i)^2} e_i e_i^\top + 2(x^k - x^*)(x^k - x^*)^\top + \|x^k - x^*\|^2 I \right) d \geq 0,
\]

for all \( d \in \mathbb{R}^n \). Let \( \theta^k_i = \mu_k/(x^k_i)^2 \) for all \( i = 1, 2, \ldots, n \). Then, let \( \delta_k = 3\|x^k - x^*\|^2 \), which is the largest eigenvalue of the matrix \( 2(x^k - x^*)(x^k - x^*)^\top + \|x^k - x^*\|^2 I \), so that \( \delta_k > 0 \) and \( \delta_k \rightarrow 0 \), implying (10d). Finally, we note that

\[
\min\{x_i, 1\}^2 \theta^k_i = \frac{\mu_k \min\{x_i, 1\}^2}{(x^k_i)^2} \leq \mu_k \rightarrow 0
\]

so that (10e) must hold. \( \square \)

**A.2 Proof of Lemma 8.** For scalar \( y > -1 \), define \( g(y) = -\log(1+y) \).

We have \( g'(y) = -1/(1+y) \), \( g''(y) = 1/(1+y)^2 \) and \( g^{(3)}(y) = -2/(1+y)^3 \). By Taylor’s theorem, we have

\[
g(y) = g(0) + yg'(0) + \frac{1}{2} y^2 g''(0) + \frac{1}{2} \int_0^y (y-t)^2 g^{(3)}(t)dt.
\]

Substituting \( t = yu \) and using \( |y| \leq \beta < 1 \), we have

\[
\frac{1}{2} \int_0^y (y-t)^2 g^{(3)}(t)dt = -y \int_0^1 (y-yu)^2 \frac{du}{(1+yu)^3} \leq |y|^3 \int_0^1 (1-u)^2 \frac{du}{(1-\beta u)^3}.
\]
Now, since \((1-u)^2\) is monotonically decreasing in \(u\) and \(1/(1-\beta u)^3\) is monotonically increasing in \(u\), we can apply Chebyshev’s integral inequality:

\[
|y|^3 \int_0^1 (1-u)^2 \frac{du}{(1-\beta u)^3} \leq |y|^3 \left[ \int_0^1 (1-u)^2 du \right] \left[ \int_0^1 \frac{du}{(1-\beta u)^3} \right]
\]

\[
= \frac{|y|^3}{3} \left[ \int_0^1 \frac{du}{(1-\beta u)^3} \right]
\]

\[
= \frac{|y|^3}{6} \frac{1}{\beta (1-\beta)^2} = \frac{|y|^3}{6} \frac{2-\beta}{(1-\beta)^2}.
\]

By combining with (54), we obtain

\[-\log(1+y) \leq -y + \frac{1}{2} y^2 + \frac{|y|^3}{6} \frac{2-\beta}{(1-\beta)^2}.
\]

Now, for some coordinate \(i\), let \(y = (\bar{x}_i/x_i) \, d_i\). Clearly, we have \(|y| \leq \beta\) so

\[-\log \left( 1 + \frac{\bar{x}_i}{x_i} d_i \right) \leq -\frac{\bar{x}_i}{x_i} d_i + \frac{1}{2} \left( \frac{\bar{x}_i}{x_i} d_i \right)^2 + \frac{|\bar{x}_i d_i|^3}{6} \frac{2-\beta}{(1-\beta)^2}
\]

(55)

hold. By the properties of logarithms, we have

\[-\log \left( x_i \left( 1 + \frac{\bar{x}_i}{x_i} d_i \right) \right) = -\log(x_i + \bar{x}_i d_i) = -\log(x_i) - \log \left( 1 + \frac{\bar{x}_i}{x_i} d_i \right).
\]

By rearranging this inequality and substituting from (55), we have

\[-\log(x_i + \bar{x}_i d_i) + \log(x_i) \leq -\frac{\bar{x}_i}{x_i} d_i + \frac{1}{2} \left( \frac{\bar{x}_i}{x_i} d_i \right)^2 + \frac{|d_i|^3}{6} \frac{2-\beta}{(1-\beta)^2}.
\]

By summing this inequality over \(i = 1, 2, \ldots, n\), we obtain

\[-\sum_{i=1}^n \log(x_i + \bar{x}_i d_i) + \sum_{i=1}^n \log(x_i)
\]

\[
\leq -e^\top X^{-1} \bar{X} d + \frac{1}{2} d^\top \bar{X} X^{-2} \bar{X} d + \sum_{i=1}^n \frac{|d_i|^3}{6} \frac{2-\beta}{(1-\beta)^2}
\]

\[
= -e^\top X^{-1} \bar{X} d + \frac{1}{2} d^\top \bar{X} X^{-2} \bar{X} d + \frac{2-\beta}{6(1-\beta)^2} \|d\|_3^3
\]

\[
\leq -e^\top X^{-1} \bar{X} d + \frac{1}{2} d^\top \bar{X} X^{-2} \bar{X} d + \frac{2-\beta}{6(1-\beta)^2} \|d\|_3^3,
\]

where \(|\|d\||_3\) denotes the \(\ell_3\) norm of \(d\). (The final inequality follows from \(|\|d\||_3 \leq |\|d\||_2\).

\[\square\]

**A.3 Proof of Lemma 10.** Proof. For simplicity of notation, we again use

\(H = X_k \nabla^2 \phi_{\mu}(x^k) X_k\) and \(g = X_k \nabla \phi_{\mu}(x^k)\) in the proof.
Suppose first that the unit step length \( \alpha_k = 1 \) is accepted. Then, if \( \|d^k\| < c_d \epsilon_H \), it follows from Lemma 9 that both (6b) and (6c) hold at \( x^{k+1} \), so we are in case A. Otherwise, \( \|d^k\| \geq c_d \epsilon_H \) holds, so it follows that

\[
\phi_{\mu}(x^k) - \phi_{\mu}(x^k + X_k d^k) \geq \frac{\eta}{6}\|d^k\|^3 \geq \frac{\eta c_d^3 \epsilon_H^3}{6} \geq c_{\text{sol}} \epsilon_H^3
\]

holds as well, so we are in case B.

For the remainder of the proof, we assume that \( \alpha_k < 1 \). Recall from the statement of Lemma 7 that

\[
\gamma_k = \max \left\{ \frac{\|X_k^{-1} \hat{X}_k d^k\|_{\infty}}{\beta}, 1 \right\}.
\]

For any \( j \geq 0 \) such that the sufficient decrease condition (13) does not hold, we have from (4), (27a), (27c), and Lemma 8 that

\[
\begin{align*}
- \frac{\eta}{6} \theta^3 j d^k \|d^k\|^3 &\leq \phi_{\mu}(x^k + \theta^3 \hat{X}_k d^k) - \phi_{\mu}(x^k) \\
&\leq \theta^j \nabla f(x^k)^\top \hat{X}_k d^k + \frac{\theta^j}{2} (d^k)^\top \hat{X}_k \nabla^2 f(x^k) \hat{X}_k d^k + \frac{L_H}{6} \theta^3 \|X_k d^k\|^3 \quad \text{by (4)} \\
&\quad - \mu \theta^j e^\top X_k^{-1} \hat{X}_k d^k + \frac{\mu \theta^j}{2} (d^k)^\top X_k X_k^{-2} \hat{X}_k d^k + \frac{\mu (2 - \beta)}{6(1 - \beta)^2} \theta^3 j d^k \|d^k\|^3 \quad \text{by Lemma 8} \\
&\quad = -\theta^j \gamma_k (d^k)^\top (H + 2 \epsilon_H I) d^k + \frac{\theta^j}{2} (d^k)^\top H d^k \\
&\quad + \frac{L_H}{6} \theta^3 \|X_k d^k\|^3 + \frac{\mu (2 - \beta)}{6(1 - \beta)^2} \theta^3 j d^k \|d^k\|^3 \\
&\quad = -\theta^j \left( \gamma_k - \frac{\theta^j}{2} \right) (d^k)^\top (H + 2 \epsilon_H I) d^k - \theta^j \epsilon_H d^k \|d^k\|^2 \\
&\quad + \frac{L_H}{6} \theta^3 \|X_k d^k\|^3 + \frac{\mu (2 - \beta)}{6(1 - \beta)^2} \theta^3 |d^k|^3 \\
&\quad \leq -\theta^j \gamma_k \epsilon_H \|d^k\|^2 + \frac{1}{2} \theta^j \epsilon_H \|d^k\|^2 - \theta^j \epsilon_H \|d^k\|^2 \\
&\quad + \frac{L_H (1 - \beta)^2 + (2 - \beta)}{6(1 - \beta)^2} \theta^3 \|d^k\|^3 \quad \text{by (27a)} \\
&\quad \leq -\theta^j \gamma_k \epsilon_H \|d^k\|^2 + \frac{L_H (1 - \beta)^2 + (2 - \beta)}{6(1 - \beta)^2} \theta^3 |d^k|^3.
\end{align*}
\]

Therefore, for any \( j \geq 0 \) at which sufficient decrease is not attained, we have by rearranging terms in the inequality above and using the definition of \( \gamma_k \) that

\[
\frac{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)}{6(1 - \beta)^2} \theta^j \geq \max \left\{ \frac{\|X_k^{-1} \hat{X}_k d^k\|_{\infty}}{\beta}, 1 \right\} \epsilon_H \|d^k\|-1
\]

(56)

Evaluating this expression at \( j = 0 \), we have that

\[
\|d^k\| \geq \frac{6(1 - \beta)^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta) \epsilon_H}.
\]

(57)
From (27b), we have
\[ \|d^k\| \leq 1.1 \epsilon_H^{-1} \|g\| \leq 1.1 \epsilon_H^{-1} (\|X_k \nabla f(x^k)\| + \mu \|X_k X_k^{-1} e\|) \leq 1.1 \epsilon_H^{-1} (U_g + \mu \sqrt{n}) \]

where we used \(\|X_k\| \leq 1\), \(\|\nabla f(x^k)\| \leq U_g\), and \(\|X_k X_k^{-1} e\| \leq \sqrt{n}\) in the final inequality. Thus, for any \(j > j_{sol}\) we have from definition (41) and this bound on \(\|d^k\|\) that
\[ \theta_{2j} \leq \theta_{2j_{sol}} \leq \frac{6(1 - \beta)^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta) \frac{\epsilon_H^2}{1.1(U_g + \mu \sqrt{n})}} \]
\[ \leq \frac{6(1 - \beta)^2 \epsilon_H}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \|d^k\|^{-1}. \]

Therefore, (56) cannot be satisfied for any \(j > j_{sol}\) so the line search must terminate with \(\alpha_k = \theta_{j_k}\) for some \(1 \leq j_k \leq j_{sol} + 1\). The previous index \(j_k - 1\) satisfies (56), so we also have
\[ \theta_{2(j_k - 1)} = \frac{\theta_{2j_k}}{\theta^2} \geq \frac{6(1 - \beta)^2 \epsilon_H}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \|d^k\|^{-1}. \]

It follows that
\[ \phi^\mu(x^k) - \phi^\mu(x^k + \theta_{j_k} X_k d^k) \geq \frac{\eta}{6} \theta_{3j_k} \|d^k\|^3 \]
\[ \geq \frac{\eta}{6} \left[ \frac{6(1 - \beta)^2 \theta_{j_k}^2 \epsilon_H}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \|d^k\|^{-1} \right]^{3/2} \|d^k\|^3 \]
\[ \geq \frac{\eta}{6} \left[ \frac{6(1 - \beta)^2 \theta_{j_k}^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \right]^{3/2} \epsilon_H^{3/2} \|d^k\|^{3/2} \]
\[ \geq \frac{\eta}{6} \left[ \frac{6(1 - \beta)^2 \theta_{j_k}^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \right]^{3} \epsilon_H \]

holds, where the final inequality comes from (57) and \(\theta < 1\). Thus, the conclusion holds in this case as well and the proof is complete. \(\square\)

A.4 Proof of Lemma 11. Proof. We again use the notation \(H = X_k \nabla^2 \phi^\mu(x^k) X_k\) and \(g = X_k \nabla \phi^\mu(x^k)\) in this proof.

We begin by noting that when the unit step, \(\alpha_k = 1\), is taken, we have
\[ \phi^\mu(x^k) - \phi^\mu(x^{k+1}) \geq \frac{\eta}{6} \|d^k\|^3 \geq \frac{\eta}{6} \epsilon_H^3 \]

where the final inequality follows from (30).

In the remainder of the proof, we assume that the unit step length is not accepted. Then, for any \(j \geq 0\) such that (13) does not hold, we have from (4) and (30) along
with the result of Lemma 8 that
\[
- \frac{\eta}{6} \theta^3 \|d^k\|^3 \\
\leq \phi_\mu(x^k + \theta^j \bar{X}k d^k) - \phi_\mu(x^k) \\
\leq \theta^j \nabla f(x^k)^\top \bar{X}k d^k + \frac{\theta^j}{2} (d^k)^\top \bar{X}k \nabla^2 f(x^k) \bar{X}k d^k + \frac{L_H}{6} \theta^3 \|\bar{X}k d^k\|^3 \\
- \mu \theta^j e^\top X_k^{-1} \bar{X}k d^k + \frac{\mu \theta^j}{2} (d^k)^\top X_k X_k^{-2} \bar{X}k d^k + \frac{\mu(2 - \beta)}{6(1 - \beta)^2} \theta^3 \|d^k\|^3 \\
\leq \theta^j g^\top d^k + \frac{\theta^j}{2} (d^k)^\top H d^k + \frac{L_H}{6} \theta^3 \|X_k d^k\|^3 + \frac{\mu(2 - \beta)}{6(1 - \beta)^2} \theta^3 \|d^k\|^3 \\
\leq - \frac{\theta^j}{2} \|d^k\|^3 + \frac{L_H (1 - \beta)^2 + (2 - \beta) \theta^j \|d^k\|^3}{6(1 - \beta)^2}, \\
\text{by Lemma 8}
\]

By rearranging this expression, we have for all such \(j\) that
\[
\theta^j \geq \frac{3(1 - \beta)^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)}
\]

which is true only for \(j \leq j_{nc}\). Thus, the line search must terminate for some \(j_k \leq j_{nc} + 1\). Since the line search failed to stop at iteration \(j_k - 1\), we must have
\[
\theta^{j_k - 1} = \frac{\theta^{j_k}}{\theta} \geq \frac{3(1 - \beta)^2}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)}
\]

Therefore, using \(\|d^k\| \geq \epsilon_H\) from (30), we have that
\[
\phi_\mu(x^k) - \phi_\mu(x^k + \theta^{j_k} \bar{X}k d^k) \geq \frac{\eta}{6} \theta^{j_k} \|d^k\|^3 \geq \frac{\eta}{6} \left[ \frac{3(1 - \beta)^2 \theta}{(L_H + \eta)(1 - \beta)^2 + (2 - \beta)} \right]^3 \epsilon_H^3,
\]
as required. \(\square\)