A LINE-SEARCH DESCENT ALGORITHM FOR STRICT SADDLE FUNCTIONS WITH COMPLEXITY GUARANTEES*

MICHAEL O'NEILL AND STEPHEN J. WRIGHT

Abstract. We describe a line-search algorithm which achieves the best-known worst-case complexity results for problems with a certain "strict saddle" property that has been observed to hold in low-rank matrix optimization problems. Our algorithm is adaptive, in the sense that it makes use of backtracking line searches and does not require prior knowledge of the parameters that define the strict saddle property.

1 Introduction. Formulation of machine learning (ML) problems as nonconvex optimization problems has produced significant advances in several key areas. While general nonconvex optimization is difficult, both in theory and in practice, the problems arising from ML applications often have structure that makes them solvable by local descent methods. For example, for functions with the "strict saddle" property, nonconvex optimization methods can efficiently find local (and often global) minimizers [16].

In this work, we design an optimization algorithm for a class of low-rank matrix problems that includes matrix completion, matrix sensing, and Poisson prinicipal component analysis. Our method seeks a rank-r minimizer of the function $f(X)$, where $f : \mathbb{R}^{n\times m} \to \mathbb{R}$. The matrix X is parametrized explicitly as the outer product of two matrices $U \in \mathbb{R}^{n\times r}$ and $V \in \mathbb{R}^{m\times r}$, where $r \leq \min(m, n)$. We make use throughout of the notation

$$W = \begin{bmatrix} U \\ V \end{bmatrix} \in \mathbb{R}^{(m+n)\times r}. \tag{1}$$

The problem is reformulated in terms of W and an objective function F as follows:

$$\min_W F(W) := f(UV^T), \quad \text{where } W, U, V \text{ are related as in (1).} \tag{2}$$

Under suitable assumptions as well as the use of a specific regularizer, these problems obey the "robust strict saddle property" described by [23]. This property divides the search space into three regions: one in which the gradient of F is large, a second in which the Hessian of F has a direction of large negative curvature, and a third that is a neighborhood of the solution set, inside which a local regularity condition holds.

In this paper, we describe and analyze an algorithm with a favorable worst-case complexity for this class of problems. We characterize the maximum number of iterations as well as the maximum number of gradient evaluations required to find an approximate second order solution, showing that these quantities have at most a logarithmic dependence on the accuracy parameter for the solution. This is a vast improvement over the worst-case complexity of methods developed for general smooth, nonconvex optimization problems, which have a polynomial dependence on the inverse of the accuracy parameter (see, for example, [3, 15]).

While other algorithms for optimizing robust strict saddle problems have previously been developed, knowledge of the strict saddle parameters is required to obtain

*Version of June 14, 2020. Research supported by NSF Awards 1628384, 1634597, and 1740707; Subcontract 8F-30039 from Argonne National Laboratory; and Award N660011824020 from the DARPA Lagrange Program.

a worst case complexity that depends at most logarithmically on the solution accuracy [16]. For low-rank matrix problems, the strict saddle parameters depend on the singular values of the matrices of the optimal solution set [23], and are thus unlikely to be known a-priori. Therefore, an essential component of any implementable method for low-rank matrix problems is adaptivity to the optimization geometry, a property achieved by the method developed in this paper. Our method maintains an estimate of the key strict saddle parameter that is used to predict which of the three regions described above contains the current iterate. This prediction determines whether a negative gradient step or a negative curvature step is taken. When the method infers that the iterate is in a neighborhood of the solution set, a monitoring strategy is employed to detect fast convergence, or else flag that an incorrect prediction has been made. By reducing our parameter estimate after incorrect predictions, our method naturally adapts to the optimization landscape and achieves essentially the same behavior as an algorithm for which the critical parameter is known.

Notation and Background. We make use in several places of "hat" notation for matrices in the form of W in (1) in which the elements in the bottom half of the matrix are negated, that is

$$\hat{W} := \begin{bmatrix} U \\ -V \end{bmatrix}. \tag{3}$$

We use the notation $\langle A, B\rangle = \text{trace}\,(A^\top B)$.

For a scalar function $h(Z)$ with a matrix variable $Z \in \mathbb{R}^{p\times q}$, the gradient $\nabla h(Z)$ is an $p\times q$ matrix whose (i,j)-th entry is $\frac{\partial h(Z)}{\partial Z_{i,j}}$ for all $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, q$. In some places, we take the Hessian of $\nabla^2 h(Z)$ to be an $pq \times pq$ matrix whose (i,j) element is $\frac{\partial^2 h(Z)}{\partial z_i \partial z_j}$ where z_i is the i-th coordinate of the vectorization of Z. In other places, the Hessian is represented as a bilinear form defined by $[\nabla^2 h(Z)](A, B) = \sum_{i,j,k,l} \frac{\partial^2 h(Z)}{\partial Z_{i,j} Z_{k,l}} A_{i,j} B_{k,l}$ for any $A, B \in \mathbb{R}^{p\times q}$. We also write $\langle A, \nabla^2 h(Z) B\rangle$ for this same object.

Under these definitions, we can define the maximum and minimum eigenvalues of $\nabla^2 h(Z)$ as

$$\lambda_{\max}(\nabla^2 h(Z)) = \max_D \frac{\langle D, \nabla^2 h(Z) D\rangle}{\|D\|_F^2}, \quad \lambda_{\min}(\nabla^2 h(Z)) = \min_D \frac{\langle D, \nabla^2 h(Z) D\rangle}{\|D\|_F^2}. \tag{4}$$

Occasionally we need to refer to the gradient and Hessian of the original function f, prior to reparameterization. We denote the gradient by $\nabla f(X)$ and the Hessian by $\nabla^2 f(X)$, where $X = UV^\top$.

Our algorithm seeks a point that approximately satisfies second-order necessary optimality conditions for a regularized objective function $G : \mathbb{R}^{(m+n)\times r} \to \mathbb{R}$ to be defined later in (12), that is,

$$\|\nabla G(W)\|_F \le \epsilon_g, \qquad \lambda_{\min}(\nabla^2 G(W)) \ge -\epsilon_H, \tag{5}$$

for small positive tolerances ϵ_g and ϵ_H.

We assume that explicit storage and calculation of the Hessian $\nabla^2 G(W)$ is undesirable, but that products of the form $\nabla^2 G(W) Z$ can be computed efficiently for arbitrary matrices $Z \in \mathbb{R}^{(n+m)\times r}$. Computational differentiation techniques can be used to evaluate such products at a cost that is a small multiple of the cost of the gradient evaluation $\nabla G(W)$ [8].

2 Related Work. One major class of algorithms that have been developed to solve low-rank matrix problems utilizes customized initialization procedures to find a starting point which lies in the basin of attraction of a global minimizer. A standard optimization procedure, such as gradient descent, initialized at this point, typically converges to the minimizer. Due to the local regularity of the function in a neighborhood around the solution set, many of these methods have a linear rate of convergence after the initialization step. However, the spectral methods used to find a suitable starting point are often computationally intensive and involve procedures such as reduced singular value decomposition and/or projection onto the set of rank r matrices. These methods have been applied to a wide variety of problems including phase retrieval [2], blind-deconvolution [11], matrix completion [9, 20], and matrix sensing [21].

Another line of work focuses on characterizing the set of critical points of f. Many low-rank recovery problems are shown to obey the strict saddle assumption, in which all saddle points of the function exhibit directions of negative curvature in the Hessian. Additionally, these problems often have the favorable property that all local minimizers are global minimizers. Examples include dictionary learning [17], phase retrieval [19], tensor decomposition [5], matrix completion [6], and matrix sensing [1, 6]. When all these properties hold, gradient descent initialized at a random starting point converges in the limit, with probability 1, to a global minimizer [10]. Two recent works demonstrate that randomly initialized gradient descent has a global linear rate of convergence when applied to phase retrieval [4] and dictionary learning [7].

A number of works go a step further and characterize the global optimization geometry of these problems. These problems satisfy the robust strict saddle property, in which the domain is partitioned such that any point is either in a neighborhood of a global solution, has a direction of sufficient negative curvature in the Hessian, or has a large gradient norm. This property has been shown to hold for tensor decomposition [5], phase retrieval [19], dictionary learning [18], and general low-rank matrix problems [23]. Due to the partitioning, methods developed for general non-convex problems with saddle point escaping mechanisms are of interest in thie context. Indeed, methods such as gradient descent with occasional perturbations to escape saddle points appear to converge at a global linear rate. However, a close reading of these methods reveals that knowledge of the strict saddle parameters defining the separate regions of the domain is required to escape saddle points efficiently and obtain linear convergence rates. In particular, for gradient descent with perturbations, these parameters are used to decide when the perturbations should be applied. The same issue arises for methods developed specifically for strict saddle functions, such as the second-order trust region method of [16] and the Newton-based method of [12], the latter requiring knowledge of a strict saddle parameter to flip the eigenvalues of the Hessian matrix at every iteration. Unfortunately, for low-rank matrix problems of the form (2), these parameters correspond to the first and r-th singular value of the optimal solution [23] — information that is unlikely to be known a-priori.

In this work, we develop the first method for low-rank matrix problems whose worst-case complexity depends at most logarithmically on the solution accuracy, without relying on an expensive initialization procedure or knowledge of the strict saddle parameters. The method maintains its estimate of the crucial strict saddle parameter along with gradient and negative curvature information to infer which of the three regions in the partition mentioned above is occupied by the current iterate. By choosing appropriate steps based on this inference, the method converges to an approximate second-order stationary point from any starting point while dependence

on the approximation tolerances in (5) is only logarithmic.

3 Robust Strict Saddle Property and Assumptions. Here we provide the background and assumptions needed to describe the robust strict saddle property for low-rank matrix problems, as well as the additional assumptions required by our optimization algorithm. Section 3.1 provides defitions for functions invariant under orthogonal transformations, our local regularity condition, and the robust strict saddle property. Section 3.2 discusses the regularization term that we add to $F(W)$ and provides definitions for the gradient and Hessian of the regularized function. Finally, we describe our assumptions and the strict saddle parameters in Section 3.3

3.1 Regularity Condition and Robust Strict Saddle Property. Let $\mathcal{O}_r := \{R \in \mathbb{R}^{r\times r} : R^\top R = I\}$ be the set of $r \times r$ orthogonal matrices. We have the following definition.

DEFINITION 1. *Given a function $h(Z) : \mathbb{R}^{p\times r} \to \mathbb{R}$ we say that h is invariant under orthogonal transformations if*

$$h(ZR) = h(Z),$$

for all $Z \in \mathbb{R}^{p\times r}$ and $R \in \mathcal{O}_r$.

It is easy to verify that F defined in (2) satisfies this property.

We note that the Frobenius norm of Z is invariant under orthogonal transformation as well, i.e. $\|ZR\|_F = \|Z\|_F$ for all $R \in \mathcal{O}_r$. We can define the distance between two matrices Z^1 and Z^2 as follows:

$$\operatorname{dist}(Z^1, Z^2) := \min_{R\in\mathcal{O}_r} \|Z^1 - Z^2R\|_F. \tag{6}$$

For convenience, we denote by $R(Z^1, Z^2)$ the orthogonal matrix that achieves the minimum in (6), that is,

$$R(Z^1, Z^2) := \operatorname{argmin}_{R\in\mathcal{O}_r} \|Z^1 - Z^2R\|_F \tag{7}$$

We can now define the local regularity condition of interest in this work; these conditions were defined in a slightly more general setting in [2, 21].

DEFINITION 2. *Suppose $h : \mathbb{R}^{p\times r} \to \mathbb{R}$ is invariant under orthogonal transformations. Let $Z^* \in \mathbb{R}^{p\times r}$ be a local minimium of h. Define the ball of radius δ around Z^* as*

$$B(Z^*, \delta) := \left\{Z \in \mathbb{R}^{p\times r} : \operatorname{dist}(Z, Z^*) \le \delta\right\},$$

where $\operatorname{dist}(\cdot,\cdot)$ is defined in (6). Then, we say that $h(Z)$ satisfies the (α, β, δ)-regularity condition at Z^* (where α, β, and δ are all positive quantities) if for all $Z \in B(Z^*, \delta)$, we have

$$\langle \nabla h(Z), Z - Z^*R\rangle \ge \alpha \operatorname{dist}(Z, Z^*)^2 + \beta \|\nabla h(Z)\|_F^2, \quad \text{where } R = R(Z, Z^*). \tag{8}$$

Note that α and β in Definition 2 must satisfy $\alpha\beta \le 1/4$ because of the Cauchy-Schwarz inequality, which indicates that for any $R \in \mathcal{O}_r$ we have

$$\langle \nabla h(Z), Z - Z^*R\rangle \le \operatorname{dist}(Z, Z^*)\|\nabla h(Z)\|_F,$$

and the inequality of arithmetic and geometric means,

$$\alpha \operatorname{dist}(Z, Z^*)^2 + \beta \|\nabla h(Z)\|_F^2 \ge 2\sqrt{\alpha\beta}\operatorname{dist}(Z, Z^*)\|\nabla h(Z)\|_F.$$

In addition, (8) implies that

$$\beta\|\nabla h(Z)\|_F \leq \operatorname{dist}(Z, Z^*), \tag{9}$$

holds for all $Z \in B(Z^*, \delta)$, by the Cauchy-Schwarz inequality and $\alpha \operatorname{dist}(Z, Z^*)^2 \geq 0$.

One important consequence of the regularity condition is local convergence of gradient descent at a linear rate.

LEMMA 3. *Let the function $h : \mathbb{R}^{p\times r} \to \mathbb{R}$ restricted to a δ neigborhood of $Z^* \in \mathbb{R}^{p\times r}$ satisfies the (α, β, δ)-regularity condition and suppose that $Z^0 \in B(Z^*, \delta)$. Then, after $k+1$ steps of gradient descent applied to h starting from Z^0, with stepsizes $\nu_j \in (0, 2\beta]$ for all $j = 0, 1, \ldots k$, we have*

$$\operatorname{dist}^2(Z^{k+1}, Z^*) \leq \left[\prod_{j=0}^{k}(1 - 2\nu_j\alpha)\right] \operatorname{dist}^2(Z^0, Z^*), \tag{10}$$

so that $Z^{k+1} \in B(x^*, \delta)$.

Proof. This proof follows a similar argument to that of [2, Lemma 7.10] Denote $R(Z, Z^*)$ be defined as in (7). By the definition of the distance (6), our regularity condition (8), and $\nu_j \leq 2\beta$, we have when $\operatorname{dist}(Z^j, Z^*) \leq \delta$ that

$$\begin{aligned}
&\operatorname{dist}^2(Z^{j+1}, Z^*) \\
&= \|Z^{j+1} - Z^* R(Z^{j+1}, Z^*)\|_F^2 \\
&\leq \|Z^{j+1} - Z^* R(Z^j, Z^*)\|_F^2 && \text{by (6)} \\
&= \|Z^j - \nu_j \nabla h(Z^j) - Z^* R(Z^j, Z^*)\|_F^2 \\
&= \|Z^j - Z^* R(Z^j, Z^*)\|_F^2 + \nu_j^2 \|\nabla h(Z^j)\|_F^2 \\
&\quad - 2\nu_j \langle \nabla h(Z^j), Z^j - Z^* R(Z^j, Z^*)\rangle \\
&\leq (1 - 2\nu_j\alpha)\operatorname{dist}^2(Z^j, Z^*) - \nu_j(2\beta - \nu_j)\|\nabla h(Z^j)\|_F^2 && \text{by (8)} \\
&\leq (1 - 2\nu_j\alpha)\operatorname{dist}^2(Z^j, Z^*) && \text{by } \nu_j \leq 2\beta.
\end{aligned}$$

Since $\alpha\beta \leq 1/4$ and $\nu_j \leq 2\beta$, we have that $0 \leq 1 - 2\nu_j\alpha \leq 1$. Thus $\operatorname{dist}(Z^{j+1}, Z^*) \leq \delta$ too. By applying this argument inductively for $j = 0, 1, \ldots, k$, we obtain the result. ∎

We are now ready to define the robust strict saddle property, for functions invariant under orthogonal transformations.

DEFINITION 4. *Suppose that the twice continuously differentiable function $h(Z) : \mathbb{R}^{p\times r} \to \mathbb{R}$ is invariant under orthogonal transformations. For the positive quantities α, β, γ, ϵ, δ, function h satisfies the $(\alpha, \beta, \gamma, \epsilon, \delta)$-robust strict saddle property if at any point Z, at least one of the following applies:*

1. *There exists a local minimum $Z^* \in \mathbb{R}^{p\times r}$ such that $\operatorname{dist}(Z, Z^*) \leq \delta$, and the function h restricted to the neighborhood $\operatorname{dist}(Z', Z^*) \leq 2\delta$ satisfies the $(\alpha, \beta, 2\delta)$-regularity condition at Z^* of Definition 2;*
2. *$\lambda_{\min}(\nabla^2 h(Z)) \leq -\gamma$; or*
3. *$\|\nabla h(Z)\|_F \geq \epsilon$.*

Under this property, each element Z of the domain belongs to at least one of three sets, each of which has a property that guarantees fast convergence of descent methods. The parameters that define these regions for low-rank matrix problems are discussed in Section 3.3.

3.2 Regularization. Let $X^* \in \mathbb{R}^{n\times m}$ be a critical point of f defined in (2), that is, $\nabla f(X^*) = 0$. Suppose that X^* has rank $r \leq \min(m, n)$ (see Assumption 1 below), and let $X^* = \Phi\Sigma\Psi^\top$ be the SVD of X^*, where $\Phi \in \mathbb{R}^{n\times r}$ and $\Psi \in \mathbb{R}^{m\times r}$ have orthonormal columns and Σ is positive diagonal. Define

$$U^* = \Phi\Sigma^{1/2}R, \quad V^* = \Psi\Sigma^{1/2}R \tag{11}$$

for some $R \in \mathcal{O}_r$. To remove ambiguity in the matrix W that corresponds to X^*, we add to $F(W)$ the regularization term ρ defined by

$$\rho(W) := \frac{1}{4}\left\|U^\top U - V^\top V\right\|_F^2.$$

The regularized optimization problem that we solve in this paper is thus

$$\min_{U\in\mathbb{R}^{n\times r}, V\in\mathbb{R}^{m\times r}} G(W) := F(W) + \frac{1}{2}\rho(W). \tag{12}$$

The regularization parameter $1/2$ is chosen for convenience and is sufficient to ensure the robust strict saddle property holds. Note that for $(U, V) = (U^*, V^*)$ defined in (11), and for any $R \in \mathcal{O}_r$, with W^* and $\hat{W}^*$ defined as in (1) and (3), we have

$$(\hat{W}^*)^\top W^* = (U^*)^T U^* - (V^*)^T V^* = R^T\Sigma R - R^T\Sigma R = 0. \tag{13}$$

We can show from (11) together with the definitions of X^* and W^* that

$$\|W^*\|^2 = 2\|X^*\|, \quad \|W^*(W^*)^T\|_F = 2\|X^*\|_F. \tag{14}$$

(We include a proof of these claims in Appendix A, for completeness.)

For the gradient of $G(W)$, we have

$$\nabla G(W) = \begin{bmatrix}\nabla f(X)V \\ (\nabla f(X))^\top U\end{bmatrix} + \frac{1}{2}\hat{W}\hat{W}^\top W, \tag{15}$$

where $X = UV^\top$. Given matrices D and $\hat{D}$ defined by

$$D = \begin{bmatrix}S \\ Y\end{bmatrix}, \quad \hat{D} = \begin{bmatrix}S \\ -Y\end{bmatrix}, \quad \text{where } S \in \mathbb{R}^{n\times r} \text{ and } Y \in \mathbb{R}^{m\times r}, \tag{16}$$

the bilinear form of the Hessian of G is given by

$$\begin{aligned}[\nabla^2 G(W)](D, D) &= [\nabla^2 f(X)](SV^\top + UY^\top, SV^\top + UY^\top) + 2\langle\nabla f(X), SY^\top\rangle \\ &\quad + \frac{1}{2}\langle\hat{W}^\top W, \hat{D}^\top D\rangle + \frac{1}{4}\|\hat{W}^\top D + D^\top\hat{W}\|_F^2,\end{aligned} \tag{17}$$

where $X = UV^\top$.

3.3 Assumptions and Strict Saddle Parameters. We make the following assumptions on $f(X)$, which are identical to those found in [23]. The first is about existence of a rank-r critical point for f.

ASSUMPTION 1. *$f(X)$ has a critical point $X^* \in \mathbb{R}^{n\times m}$ with rank r.*

The second assumption is a restricted strong convexity condition for f.

ASSUMPTION 2. *The twice continuously differentiable function $f : \mathbb{R}^{n\times m} \to \mathbb{R}$ is $(2r, 4r)$-restricted strongly convex and smooth, that is, for any matrices $X, T \in \mathbb{R}^{n\times m}$ with $\operatorname{rank}(X) \le 2r$ and $\operatorname{rank}(T) \le 4r$, the Hessian $\nabla^2 f(X)$ satisfies*

$$a\|T\|_F^2 \le [\nabla^2 f(X)](T,T) \le b\|T\|_F^2, \tag{18}$$

for some positive scalars a and b.

Assumption 2 implies that the original function, prior to splitting the variable X into U and V, obeys a form of restricted strong convexity and smoothness. This assumption is satisfied when $4r$-RIP holds, which occurs with high probability (under certain assumptions) for such problems as low-rank matrix completion and matrix sensing [14].

Now, we are able to define the robust strict saddle conditions for G. We state the following slightly abbreviated version of [23, Theorem 1].

THEOREM 5. *Let $G(W)$ be defined as in (12). For the critical point $X^* \in \mathbb{R}^{n\times m}$, with rank r, suppose that $X^* = U^*(V^*)^T$, where $U^* \in \mathbb{R}^{n\times r}$ and $V^* \in \mathbb{R}^{m\times r}$ are defined as in (11), and define $W^* = \begin{bmatrix} U^* \\ V^* \end{bmatrix}$. Let $\operatorname{dist}(\cdot,\cdot)$ be defined as in (6), and let $\sigma_r(Z) > 0$ denote the r-th singular value of the matrix Z. Suppose that Assumptions 1 and 2 are satisfied for positive a and b such that*

$$\frac{b-a}{a+b} \le \frac{1}{100}\frac{\sigma_r^{3/2}(X^*)}{\|X^*\|_F\|X^*\|^{1/2}}.$$

Define the following regions of the space of matrices $\mathbb{R}^{(m+n)\times r}$:

$$\begin{aligned}
\mathcal{R}_1 &:= \left\{W : \operatorname{dist}(W,W^*) \le \sigma_r^{1/2}(X^*)\right\},\\
\mathcal{R}_2 &:= \left\{W : \sigma_r(W) \le \sqrt{\frac{1}{2}}\sigma_r^{1/2}(X^*),\ \ \|WW^\top\|_F \le \frac{20}{19}\|W^*(W^*)^\top\|_F\right\},\\
\mathcal{R}_3' &:= \left\{W : \operatorname{dist}(W,W^*) > \sigma_r^{1/2}(X^*),\ \ \|W\| \le \frac{20}{19}\|W^*\|,\right.\\
&\qquad \left.\sigma_r(W) > \sqrt{\frac{1}{2}}\sigma_r^{1/2}(X^*),\ \ \|WW^\top\|_F \le \frac{20}{19}\|W^*(W^*)^\top\|_F\right\},\\
\mathcal{R}_3'' &:= \left\{W : \|W\| > \frac{20}{19}\|W^*\| = \sqrt{2}\frac{20}{19}\|X^*\|^{1/2},\ \ \|WW^\top\|_F \le \frac{10}{9}\|W^*(W^*)^\top\|_F\right\},\\
\mathcal{R}_3''' &:= \left\{W : \|WW^\top\|_F > \frac{10}{9}\|W^*(W^*)^\top\|_F = \frac{20}{9}\|X^*\|_F\right\}.
\end{aligned}$$

(Note that the definitions of $\mathcal{R}_3''$ and $\mathcal{R}_3'''$ make use of (14).) Then there exist positive constants c_α, c_β, c_γ, and c_ϵ such that $G(W)$ has the following strict saddle property.

1. *For any $W \in \mathcal{R}_1$, $G(W)$ satisfies the local regularity condition:*

$$\begin{aligned}
&\langle \nabla G(W), W - W^* R(W,W^*)\rangle\\
&\qquad \ge c_\alpha \sigma_r(X^*)\operatorname{dist}^2(W,W^*) + \frac{c_\beta}{\|X^*\|}\|\nabla G(W)\|_F^2,
\end{aligned} \tag{19}$$

where $\operatorname{dist}(W,W^*)$ is defined in (6) and $R(W,W^*)$ is defined in (7). That is, definition (8) is satisfied with $h = G$, $x = W$, $\alpha = c_\alpha\sigma_r(X^*)$, $\beta = c_\beta\|X^*\|^{-1}$, and $\delta = \sigma_r^{1/2}(X^*)$.

2. *For any $W \in \mathcal{R}_2$, $G(W)$ has a direction of large negative curvature, that is,*

$$\lambda_{\min}(\nabla^2 G(W)) \leq -c_\gamma \sigma_r(X^*). \tag{20}$$

3. *For any $W \in \mathcal{R}_3 = \mathcal{R}_3' \cup \mathcal{R}_3'' \cup \mathcal{R}_3'''$, $G(W)$ has a large gradient, that is,*

$$\|\nabla G(W)\|_F \geq c_\epsilon \sigma_r^{3/2}(X^*), \qquad \text{for all } W \in \mathcal{R}_3'; \tag{21a}$$

$$\|\nabla G(W)\|_F \geq c_\epsilon \|W\|^3, \qquad \text{for all } W \in \mathcal{R}_3''; \tag{21b}$$

$$\|\nabla G(W)\|_F \geq c_\epsilon \|WW^\top\|_F^{3/2}, \quad \text{for all } W \in \mathcal{R}_3'''. \tag{21c}$$

It follows from this theorem that the function G satisfies the robust strict saddle property of Definition 4 with

$$\alpha = c_\alpha \sigma_r(X^*), \quad \gamma = c_\gamma \sigma_r(X^*), \quad \delta = \sigma_r^{1/2}(X^*), \quad \beta = c_\beta \|X^*\|^{-1},$$

and different values of ϵ that depend on the region:

$$\begin{aligned}
\epsilon_{R_3'} &= c_\epsilon \sigma_r(X^*)^{3/2}, \\
\epsilon_{R_3''} &= c_\epsilon \|W\|^3 \geq c_\epsilon \left(\sqrt{2}\frac{20}{19}\right)^3 \|X^*\|^{3/2}, \\
\epsilon_{R_3'''} &= c_\epsilon \|WW^\top\|_F^{3/2} \geq c_\epsilon \left(\frac{20}{19}\right)^{3/2} \|X^*\|_F^{3/2}.
\end{aligned}$$

The regions defined in Theorem 5 span the space of matrices occupied by W but are not a partition, that is,

$$\mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3' \cup \mathcal{R}_3'' \cup \mathcal{R}_3''' = \mathcal{R}^{(n+m)\times r}.$$

The constants c_α, c_β, c_γ, and c_ϵ in this theorem may vary between problems in this class. Settings that work for all classes mentioned are

$$c_\alpha = \frac{1}{16}, \quad c_\beta = \frac{1}{260}, \quad c_\gamma = \frac{1}{6}, \quad c_\epsilon = \frac{1}{50}.$$

(For clarity, we use the same constant, c_ϵ, for each equation in (21) even though slightly tighter bounds are possible if each is treated individually.) Note that these constants are used in the algorithm presented below. For convenience, we define the following combination of the parameters above, which is used repeatedly in the algorithms and analysis below:

$$c_s := c_\epsilon \min\left\{1, c_\alpha^{3/2} c_\beta^{3/2}\right\}. \tag{22}$$

In addition to the strict saddle assumption, we make the following standard assumptions on G, concerning compactness of the level set defined by the initial point W^0 and smoothness.

ASSUMPTION 3. *Given an initial iterate W^0, the level set defined by $\mathcal{L}_G(W^0) = \{W | G(W) \leq G(W^0)\}$ is compact.*

ASSUMPTION 4. *The function G is twice Lipschitz continuously differentiable with respect to the Frobenius norm on an open neighborhood of $\mathcal{L}_G(W^0)$, and we denote by L_g and L_H the respective Lipschitz constants for ∇G and $\nabla^2 G$ on this set.*

Under Assumptions 3 and 4, there exist scalars G_{low}, $U_g > 0$, $U_H > 0$, and $R_{\mathcal{L}} > 0$ such that the following are satisfied for all W in an open neighborhood of $\mathcal{L}_G(W^0)$:

$$G(W) \geq G_{\text{low}}, \quad \|\nabla G(W)\|_F \leq U_g, \quad \|\nabla^2 G(W)\| \leq U_H, \quad \|W\| \leq R_{\mathcal{L}}, \tag{23}$$

where the third condition is taken on the "unrolled" Hessian of G. These assumptions also imply the following well known inequalities, for W and D such that all points in the convex hull of W and D lie in the neighborhood of the level set mentioned above:

$$G(W+D) \leq G(W) + \langle \nabla G(W), D\rangle + \frac{L_g}{2}\|D\|_F^2, \tag{24a}$$

$$G(W+D) \leq G(W) + \langle \nabla G(W), D\rangle + \frac{1}{2}\langle D, \nabla^2 G(W) D\rangle + \frac{L_H}{6}\|D\|_F^3. \tag{24b}$$

Finally, we make an assumption about knowledge of the Lipschitz constant of the gradient of $f(X)$.

ASSUMPTION 5. *The gradient $\nabla f(X)$ is Lipschitz continuous on an open neighborhood of*

$$\left\{ Z : Z = UV^\top, \begin{bmatrix} U \\ V \end{bmatrix} \in \mathcal{L}_G(W^0) \right\},$$

and the associated constant, denoted by $L_{\nabla f}$, is known or can be efficiently estimated. That is, for any X_a, X_b in the set defined above, we have

$$\|\nabla f(X_a) - \nabla f(X_b)\|_F \leq L_{\nabla f}\|X_a - X_b\|_F. \tag{25}$$

In many interesting applications, $L_{\nabla f}$ is easily discerned or can be efficiently computed. An example is the low-rank matrix completion problem where a set of observations M_{ij}, $(i,j) \in \Omega$ is made of a matrix, and the objective is $f(X) = \frac{1}{2}\sum_{(i,j)\in\Omega}(X_{ij} - M_{ij})^2$. Here, we have $L_{\nabla f} = 1$. A similar example is matrix sensing problem, in which $f(X) = \frac{1}{2}\|\mathcal{A}(X) - y\|_2^2$, where $\mathcal{A} : \mathbb{R}^{n\times m} \to \mathbb{R}^p$ is a known linear measurement operator and $y \in \mathbb{R}^p$ is the set of observations. In this case, we can write $\mathcal{A}(X) = [\langle A_i, X\rangle]_{i=1,2,\ldots,p}$ where $A_i \in \mathbb{R}^{n\times m}$ for all $i = 1, 2, \ldots, p$, so that $\nabla f(X) = \sum_{i=1}^p (\langle A_i, X\rangle - y_i)A_i$ and thus $L_{\nabla f} = \sum_{i=1}^p \|A_i\|_F^2$.

4 The Algorithm. We describe our algorithm in this section. Sections 4.1 and 4.2 give a detailed description of each element of the algorithm, along with a description of how the key parameters in the definition of strict saddle are estimated. Section 4.4 shows that the algorithm properly identifies the strict saddle regions once the parameter γ_k is a sufficiently good estimate of $\sigma_r(X^*)$.

4.1 Line-Search Algorithm for Strict Saddle Functions. Our main algorithm is defined in Algorithm 1. At each iteration, it attempts to identify the region curently occupied by W^k. A step appropriate to the region is computed. The critical parameter in identifying the regions is γ_k, which is our upper estimate of the parameter $\gamma = \sigma_r(X^*)$ (ignoring the constant c_γ), which plays a key role in the definitions of the regions in Theorem 5. When the large gradient condition is satisfied (that is, when W^k is estimated to lie in $\mathcal{R}_2$), a gradient descent step is taken. Similarly, when the condition for large negative curvature is satisfied (that is, when W^k is estimated to lie in $\mathcal{R}_1$), a direction of significant negative curvature is found by using Procedure 3, and a step is taken in a scaled version of this direction. (The approach for negative

Algorithm 1 Line-Search Algorithm For Strict Saddle Functions

Inputs: Optimality tolerances $\epsilon_g \in (0,1)$, $\epsilon_H \in (0,1)$; starting point W^0; starting guess $\gamma_0 \geq \sigma_r(X^*) > 0$; step acceptance parameter $\eta \in (0,1)$; backtracking parameter $\theta \in (0,1)$; Lipschitz constant $L_{\nabla f} \geq 0$;
Optional Inputs: Scalar $M > 0$ such that $\|\nabla^2 G(W)\| \leq M$ for all W in an open neighborhood of $\mathcal{L}_G(W^0)$;
converged $\leftarrow$ False;
for $k = 0, 1, 2, \ldots$ **do**
  if $\|\nabla G(W^k)\|_F \geq c_s \gamma_k^{3/2}$ **then** {Large Gradient, Take Steepest Descent Step}
    Compute $\nu_k = \theta^{j_k}$, where j_k is the smallest nonnegative integer such that

$$G(W^k - \nu_k \nabla G(W^k)) < G(W^k) - \eta \nu_k \|\nabla G(W^k)\|_F^2; \tag{26}$$

    $W^{k+1} \leftarrow W^k - \nu_k \nabla G(W^k)$;
    $\gamma_{k+1} \leftarrow \gamma_k$;
  else {Seek Direction of Large Negative Curvature}
    Call Procedure 3 with $H = \nabla^2 G(W^k)$, $\epsilon = c_\gamma \gamma_k$, and M (if provided);
    if Procedure 3 certifies $\lambda_{\min}(\nabla^2 G(W^k)) \geq -c_\gamma \gamma_k$ **then** {Initialize Local Phase}
      $\alpha_k \leftarrow c_\alpha \gamma_k$;
      $\delta_k \leftarrow \sqrt{2}\gamma_k^{1/2}$;
      $\beta_k \leftarrow \frac{2c_\beta}{(\delta_k + \|W^k\|_F)^2}$;
      if $\alpha_k \beta_k \leq \frac{1}{4}$ **and** $\|\nabla G(W^k)\|_F \leq \frac{\delta_k}{\beta_k}$ **and**
        $2\|\nabla f(X^k)\|_F + \frac{1}{2}\|(\hat{W}^k)^\top W^k\|_F \leq (2L_{\nabla f} + \frac{1}{2})(2\|W^k\|_F + \delta_k)\delta_k$ **then** {Try Local Phase}
        Call Algorithm 2 with $W_0^k = W^k$, ϵ_g, ϵ_H, α_k, β_k, δ_k, η, θ, and $L_{\nabla f}$ to obtain outputs W^{k+1}, T_k, converged;
        if converged = True **then** {Local Phase Found Near-Optimal Point}
          Terminate and return W^{k+1};
        end if
      else {Do not update W}
        $W^{k+1} \leftarrow W^k$;
      end if
      $\gamma_{k+1} \leftarrow \frac{1}{2}\gamma_k$;
    else {Search in Large Negative Curvature Direction s from Procedure 3}
      Set $D^k \leftarrow -\text{sgn}\left(\langle S, \nabla G(W^k)\rangle\right) |\langle S, \nabla^2 G(W^k) S\rangle| S$, where S is the $\mathbb{R}^{(n+m)\times r}$ matrix formed by reshaping the output vector s from Procedure 3;
      Compute $\nu_k = \theta^{j_k}$, where j_k is the smallest nonnegative integer such that

$$G(W^k + \nu_k D^k) < G(W^k) + \eta \frac{\nu_k^2}{2} \langle D^k, \nabla^2 G(W^k) D^k \rangle; \tag{27}$$

      $W^{k+1} \leftarrow W^k + \nu_k D^k$;
      $\gamma_{k+1} \leftarrow \gamma_k$;
    end if
  end if
end for

curvature steps is similar to that of [15].) In both cases, a backtracking line search is used to ensure sufficient decrease in G.

Algorithm 2 Local Phase for Strict Saddle Problems

Inputs: Optimality tolerances $\epsilon_g \in (0,1]$, $\epsilon_H \in (0,1]$; starting point W_0^k; strict saddle parameters α_k, β_k, $\delta_k > 0$; step acceptance parameter $\eta \in (0,1)$; backtracking parameter $\theta \in (0,1)$; Lipschitz constant $L_{\nabla f} \geq 0$;
Outputs: W^{k+1}, T_k, converged;
converged $\leftarrow$ False;
$\kappa_0 \leftarrow 1$;
$\tau_0 \leftarrow (2L_{\nabla f} + \frac{1}{2})(2\|W_0^k\|_F + \delta_k)\delta_k$;
$t \leftarrow 0$;
while $\|\nabla G(W_t^k)\|_F \leq \frac{\sqrt{\kappa_t}}{\beta_k}\delta_k$ **and** $2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \leq \tau_t$ **do**
  Compute $\nu_t = 2\beta_k\theta^{j_t}$, where j_t is the smallest nonnegative integer such that
  $$G(W_t^k - \nu_t \nabla G(W_t^k)) < G(W_t^k) - \eta\nu_t\|\nabla G(W_t^k)\|_F^2; \tag{28}$$
  $W_{t+1}^k \leftarrow W_t^k - \nu_t \nabla G(W_t^k)$;
  $\kappa_{t+1} \leftarrow (1 - 2\nu_t\alpha_k)\kappa_t$;
  $\tau_{t+1} \leftarrow (2L_{\nabla f} + \frac{1}{2})(2\|W_{t+1}^k\|_F + \sqrt{\kappa_{t+1}}\delta_k)\sqrt{\kappa_{t+1}}\delta_k$;
  $t \leftarrow t + 1$;
  if $\|\nabla G(W_t^k)\| \leq \epsilon_g$ **and** $2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \leq \epsilon_H$ **then**
    converged $\leftarrow$ True;
    break
  end if
end while
$W^{k+1} \leftarrow W_t^k$;
$T_k \leftarrow t$;

When neither of these two scenarios are satisfied, the algorithm enters a "local phase" defined by Algorithm 2. This process begins by estimating a number of the robust strict saddle parameters of Definition 4. These parameters are chosen so that the local phase will converge linearly to an approximate second-order point *provided that* γ_k is within a factor of two of the value of $\sigma_r(X^*)$, that is,

$$\gamma_k \in \Gamma(X^*) := \left[\tfrac{1}{2}\sigma_r(X^*), \sigma_r(X^*)\right). \tag{29}$$

When γ_k is in the interval $\Gamma(X^*)$, the value of δ_k defined in Algorithm 1 is an upper bound on δ, while α_k and β_k are lower bounds on α and β from Definition 4. Conditions are checked during the execution of the local phase to monitor for fast convergence. These conditions will be satisfied whenever $\gamma_k \in \Gamma(X^*)$. If the conditions are not satisfied, then (29) does not hold, so we halve the value of γ_k and proceed without taking a step in W.

The local phase, Algorithm 2, begins by initializing an inner iteration counter t as well as the scalar quantities κ_t and τ_t, which are used to check for linear convergence of W_t^k, for $t = 0, 1, 2, \ldots$ to a point satisfying 5. Each iteration of the local phase consists of a gradient descent step with a line search parameter ν_t obtained by backtracking from an initial value of $2\beta_k$. Once a stepsize ν_t is identified and the gradient descent step is taken, κ_t is updated to reflect the linear convergence rate that occurs when $\gamma_k \in \Gamma(X^*)$ and $W^k \in \mathcal{R}_1$. At each iteration, this linear convergence rate is checked, by looking at the gradient of $G(W)$ as well as the the gradient of the original function $f(X)$. Under the assumptions discussed in Section 3.3, these quantities provide estimates for (5), since the minimum eigenvalue of the Hessian of

$G(W)$ can be lower bounded using $\nabla f(X)$ (see Section 4.3 for details). These checks ensure that the local phase either converges at a linear rate to a point satisfying (5), or else exits quickly with a flag indicating that the current estimate γ_k of $\sigma_r(X^*)$ is too large.

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 λ of $\lambda_{\min}(H)$ such that $\lambda \leq -\epsilon/2$ and vector s with $\|s\| = 1$ such that $s^\top Hs = \lambda$ OR a certificate that $\lambda_{\min}(H) \geq -\epsilon$. In the latter case, the certificate is false with probability ρ for some $\rho \in [0, 1)$.

4.2 Minimum Eigenvalue Oracle. The Minimum Eigenvalue Oracle (Procedure 3) is called when the large gradient condition $\|\nabla G(W^k)\|_F \geq c_s\gamma_k^{3/2}$ does not hold. The input matrix H is the "unrolled" Hessian of $G(W)$, a symmetric matrix of dimension $N = (n+m)r$. The oracle either returns a direction along which the Hessian has curvature at most $-\epsilon/2$, or certifies that the minimum curvature is greater than $-\epsilon$. In the latter case, the certificate may be wrong with some probability $\rho \in (0, 1)$, where ρ is a user-specified parameter. When the certificate is returned, Algorithm 1 enters the local phase.

Procedure 3 can be implemented via any method that finds the smallest eigenvalue of H to an absolute precision of $\epsilon/2$ with probability at least $1 - \rho$. (A deterministic implementation based on a full eigenvalue decomposition would have $\rho = 0$.) 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 [3]). This approach does not require explicit knowledge of H, only the ability to find matrix-vector products of H with a given vector. The following result from [15, Lemma 2] verifies the effectiveness of this approach.

LEMMA 6. *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 $\rho \in [0, 1)$, this approach finds the smallest eigenvalue of H to an absolute precision of $\epsilon/2$, together with a corresponding direction s, in at most*

$$\min\left\{N, 1 + \left\lceil \frac{1}{2}\ln(2.75N/\rho^2)\sqrt{\frac{M}{\epsilon}}\right\rceil\right\} \quad \textit{iterations}, \tag{30}$$

with probability at least $1 - \rho$.

Procedure 3 can be implemented by outputting the approximate eigenvalue λ for H, determined by the randomized Lanczos process, along with the corresponding direction s, 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 $1 - \rho$.

We note here that while the second-order optimality conditions could be checked using the minimum eigenvalue oracle inside of the local phase, this procedure can be quite inefficient compared to the rest of the algorithm. From the result of Lemma 6, it is clear that attempting to verify that $\lambda_{\min}(\nabla^2 G(W)) \geq -\epsilon_H$ holds could require as many as $\min\left\{(n+m)r, \epsilon_H^{-1/2}\right\}$ gradient evaluations/Hessian-vector products. This

dependence on ϵ_H — worse than the logarithmic dependence on tolerances that is the stated goal of this work. We avoid this issue by using $\nabla f(X)$ to estimate a lower bound of the spectrum of $\nabla^2 G(W)$, as discussed in the following section. This allows us to maintain the logarithmic dependence on our optimization tolerances ϵ_g and ϵ_H while still ensuring convergence to an approximate second-order point.

4.3 Lower-Bounding the Spectrum of $\nabla^2 G(W^k)$. We now prove two technical results about quantities that lower-bound the minimum eigenvalue of Hessian of $G(W)$. These bounds motivate some unusual expressions in Algorithms 1 and 2 that allow us to check the second-order approximate optimality condition in (5) *indirectly*, and ultimately at lower cost than a direct check of $\lambda_{\min}(\nabla^2 G(W))$.

LEMMA 7. *Suppose that Assumption 2 holds, and let W and $\hat{W}$ be defined in (1) and (3), respectively, with $G(W)$ defined in (12). Then we have*

$$\lambda_{\min}(\nabla^2 G(W)) \geq -2\|\nabla f(X)\|_F - \frac{1}{2}\|\hat{W}^\top W\|_F,$$

where $X = UV^\top$.

Proof. Let D be defined as in (16), with component matrices S and Y. Since $\operatorname{rank}(X) = \operatorname{rank}(UV^\top) \leq r$ and $\operatorname{rank}(SV^\top + UY^\top) \leq 2r$, we have by Assumption 2 that

$$\langle SV^\top + UY^\top, \nabla^2 f(X)(SV^\top + UY^\top)\rangle \geq a\|D\|_F^2 \geq 0.$$

It follows from (17) and the Cauchy-Schwarz inequality that

$$\begin{aligned}\langle D, \nabla^2 G(W)D\rangle &= \langle SV^\top + UY^\top, \nabla^2 f(X)(SV^\top + UY^\top)\rangle + 2\langle \nabla f(X), SY^\top\rangle \\ &\quad + \frac{1}{2}\langle \hat{W}^\top W, \hat{D}D\rangle + \frac{1}{4}\|\hat{W}^\top D + D^\top \hat{W}\|_F^2 \\ &\geq -2\|\nabla f(X)\|_F\|S\|_F\|Y\|_F - \frac{1}{2}\|\hat{W}^\top W\|_F\|\hat{D}\|_F\|D\|_F.\end{aligned}$$

Defining D' and $\hat{D}'$ by

$$D' = \begin{bmatrix} S' \\ Y' \end{bmatrix} \in \arg\min_D \frac{\langle D, \nabla^2 G(W)D\rangle}{\|D\|_F^2}, \quad \hat{D}' = \begin{bmatrix} S' \\ -Y' \end{bmatrix}$$

we have

$$\begin{aligned}\lambda_{\min}(\nabla^2 G(W)) &\geq -2\frac{\|\nabla f(X)\|_F\|S'\|_F\|Y'\|_F}{\|D'\|_F^2} - \frac{1}{2}\frac{\|\hat{W}^\top W\|_F\|\hat{D}'\|_F\|D'\|_F}{\|D'\|_F^2} \\ &\geq -2\|\nabla f(X)\|_F - \frac{1}{2}\|\hat{W}^\top W\|_F,\end{aligned}$$

where the final inequality follows by $\|S'\|_F \leq \|D'\|_F$, $\|Y'\|_F \leq \|D'\|_F$, and $\|\hat{D}\|_F = \|\hat{D}'\|_F$. ∎

Next, we show how to relate the lower bound of Lemma 7 to the distance between W and W^*. The following result has an expression that is similar to one that appears in Algorithm 1, in the condition that determines whether to call Algorithm 2.

LEMMA 8. *Suppose that Assumptions 1, 2, and 5 hold. Let W and $\hat{W}$ be as defined as in (1) and (3), respectively, and let X^* be as in Assumption 1, with U^* and V^* (and hence W^*) defined as in (11), for some $R \in \mathcal{O}_r$. Then we have*

$$2\|\nabla f(X)\|_F + \frac{1}{2}\|\hat{W}^\top W\|_F \leq \left(2L_{\nabla f} + \frac{1}{2}\right)(2\|W\|_F + \operatorname{dist}(W, W^*))\operatorname{dist}(W, W^*).$$

Proof. We begin by bounding $\|\nabla f(X)\|_F$. Given W, and hence U and V, let $R = R(W, W^*)$ in (11) be the matrix in $\mathcal{O}_r$ that minimizes $\|WR - W^*\|_F$. (Note that the same minimizes $\|\hat{W}R - \hat{W}^*\|_F$, that is, $R(W, W^*) = R(\hat{W}, \hat{W}^*)$). By Assumption 5 and the definition of X^*, we have

$$\begin{aligned}\|\nabla f(X)\|_F &= \|\nabla f(X) - \nabla f(X^*)\|_F \\ &= \|\nabla f(UR(VR)^\top) - \nabla f(U^*(V^*)^\top)\|_F \\ &\leq L_{\nabla f}\|UR(VR)^\top - U^*(V^*)^\top\|_F.\end{aligned}$$

Further, we have

$$\begin{aligned}\|UR(VR)^\top - U^*(V^*)^\top\|_F &= \|UR(VR)^\top - UR(V^*)^\top + UR(V^*)^\top - U^*(V^*)^\top\|_F \\ &\leq \|UR(R^\top V^\top - (V^*)^\top)\|_F + \|(UR - U^*)(V^*)^\top\|_F \\ &\leq \|UR\|_F\|R^\top V^\top - (V^*)^\top\|_F + \|(V^*)^\top\|_F\|UR - U^*\|_F \\ &\leq (\|W\|_F + \|W^*\|_F)\text{dist}(W, W^*),\end{aligned}$$

so that

$$2\|\nabla f(X)\|_F \leq 2L_{\nabla f}(\|W\|_F + \|W^*\|_F)\text{dist}(W, W^*). \tag{31}$$

To bound $\|\hat{W}^\top W\|_F$, we have by $(\hat{W}^*)^\top W^* = 0$ that

$$\begin{aligned}\|\hat{W}^\top W\|_F &= \|(\hat{W}R)^\top WR\|_F \\ &= \|(\hat{W}R)^\top WR - (\hat{W}R)^\top W^* + (\hat{W}R)^\top W^* - (\hat{W}^*)^\top W^*\|_F \\ &\leq \|(\hat{W}R)^\top(WR - W^*)\|_F + \|((\hat{W}R)^\top - (\hat{W}^*)^\top)W^*\|_F \\ &\leq \|(\hat{W}R)^\top\|_F\|WR - W^*\|_F + \|W^*\|_F\|(\hat{W}R)^\top - (\hat{W}^*)^\top\|_F \\ &\leq (\|W\|_F + \|W^*\|_F)\text{dist}(W, W^*). \qquad (32)\end{aligned}$$

By combining (31) and (32), we have

$$2\|\nabla f(X)\|_F + \frac{1}{2}\|\hat{W}^\top W\|_F \leq (2L_{\nabla f} + \frac{1}{2})(\|W\|_F + \|W^*\|_F)\text{dist}(W, W^*).$$

To obtain the result, note that for $R = R(W, W^*)$, we have

$$\|W^*\|_F = \|W^* - WR + WR\|_F \leq \|W^* - WR\|_F + \|WR\|_F = \text{dist}(W, W^*) + \|W\|_F.$$

□

4.4 Behavior of Algorithm 1 under Accurate Parameter Estimates. In this section, we will show that Algorithm 1 correctly identifies the regions $\mathcal{R}_2$ and $\mathcal{R}_3$ when γ_k lies in the interval $\Gamma(X^*)$ defined by (29), that is, it is within a factor of two of $\sigma_r(X^*)$. Additionally, once the local phase is reached with $\gamma_k \in \Gamma(X^*)$ and $W^k \in \mathcal{R}_1$, the sequence W_t^k, $t = 0, 1, 2, \ldots$ generated in the local phase converges at a linear rate to a point satisfying (5).

These results are crucial building blocks for the main convergence results, as they show that once the parameter γ_k is a good estimate of $\sigma_r(X^*)$, the algorithm behaves well enough (with high probability) to not reduce γ_k any further, and converges rapidly thereafter at a rate that depends mostly on a polynomial in the inverse of $\sigma_r(X^*)$ rather than of the tolerances in (5).

We begin by showing that $\mathcal{R}_2$ is properly identified with high probability, in the sense that the algorithm is likely to take a successful negative curvature step when $W^k \in \mathcal{R}_2$.

LEMMA 9. *Let Assumptions 1 and 2 hold. Suppose that $W^k \in \mathcal{R}_2$ and that $\gamma_k \in \Gamma(X^*)$. Then a negative curvature step will be taken with probability at least $1-\rho$ by Algorithm 1 whenever $\|\nabla G(W^k)\|_F < c_s \gamma_k^{3/2}$.*

Proof. Since $\gamma_k \in \Gamma(X^*)$, we have $\gamma_k \leq \sigma_r(X^*)$. By Part 2 of Theorem 5, we have

$$\lambda_{\min}(\nabla^2 G(W^k)) \leq -c_\gamma \sigma_r(X^*) \leq -c_\gamma \gamma_k,$$

so a negative curvature step is taken whenever Procedure 3 is invoked (which occurs when $\|\nabla G(W^k)\|_F < c_s \gamma_k^{3/2}$), provided that Procedure 3 finds a direction of negative curvature, which happens with probability at least $1-\rho$. ∎

A similar result holds for $\mathcal{R}_3$.

LEMMA 10. *Let Assumptions 1 and 2 hold. Suppose that $W^k \in \mathcal{R}_3$ and that $\gamma_k \in \Gamma(X^*)$. Then $\|\nabla G(W^k)\|_F \geq c_s \gamma_k^{3/2}$, so a large-gradient step is taken by Algorithm 1.*

Proof. First, assume that $W^k \in \mathcal{R}_3'$. Then, by (21a) in Theorem 5, we have

$$\|\nabla G(W^k)\|_F \geq c_\epsilon \sigma_r(X^*)^{3/2} \geq c_s \gamma_k^{3/2},$$

where the second inequality follows from $\gamma_k \in \Gamma(X^*)$. Thus, a gradient step will be taken if $W^k \in \mathcal{R}_3'$.

Next, assume that $W^k \in \mathcal{R}_3''$. Since $\gamma_k \in \Gamma(X^*)$, we have that $\alpha = c_\alpha \sigma_r(X^*) \geq c_\alpha \gamma_k$. By Part 1 of Theorem 5, the (α, β, δ)-regularity condition of Definition 2 holds with $\alpha = c_\alpha \sigma_r(X^*)$ and $\beta = c_\beta \|X^*\|^{-1}$. Since $\alpha\beta \leq 1/4$ (as noted following Definition 2), we have

$$\frac{1}{4} \geq \alpha\beta = \frac{c_\alpha c_\beta \sigma_r(X^*)}{\|X^*\|} \geq \frac{c_\alpha c_\beta \gamma_k}{\|X^*\|},$$

so that

$$\|X^*\| \geq 4 c_\alpha c_\beta \gamma_k.$$

By the definition of $\mathcal{R}_3''$, we have

$$\|W^k\| > \frac{20}{19}\sqrt{2}\|X^*\|^{1/2} \geq \frac{40}{19}\sqrt{2} c_\alpha^{1/2} c_\beta^{1/2} \gamma_k^{1/2}.$$

Now the strict saddle property, and in particular (21b), together with (22), implies that

$$\|\nabla G(W^k)\|_F \geq c_\epsilon \|W^k\|^3 \geq c_\epsilon \left(\frac{40}{19}\sqrt{2}\right)^3 c_\alpha^{3/2} c_\beta^{3/2} \gamma_k^{3/2} \geq c_s \gamma_k^{3/2},$$

so a gradient step is taken in this case as well.

Finally, assume that $W^k \in \mathcal{R}_3'''$. By the same logic as the above, we have $\|X^*\| \geq 4 c_\alpha c_\beta \gamma_k$. From the definition of $\mathcal{R}_3'''$, we have

$$\|W^k (W^k)^\top\|_F > \frac{20}{9}\|X^*\|_F \geq \frac{20}{9}\|X^*\| \geq \frac{80}{9} c_\alpha c_\beta \gamma_k.$$

Together with (21c) and (22), we have

$$\begin{aligned}\|\nabla G(W^k)\|_F \geq c_\epsilon \|W^k (W^k)^\top\|_F^{3/2} &\geq c_\epsilon \left(\frac{80}{9}\right)^{3/2} c_\alpha^{3/2} c_\beta^{3/2} \gamma_k^{3/2} \\ &\geq c_s \gamma_k^{3/2},\end{aligned}$$

so a gradient step is taken in this case too. □

Together, Lemmas 9 and 10 imply that once $\gamma_k \in \Gamma(X^*)$, then (with high probabilty) the local phase (Algorithm 2) will be invoked only when $W^k \in \mathcal{R}_1$. With this observation in mind, we focus on the behavior of the local phase when $\gamma_k \in \Gamma(X^*)$. We show that when $W^k \in \mathcal{R}_1$ and $\gamma_k \in \Gamma(X^*)$, then W^{k+1} satisfies the approximate optimality conditions (5).

LEMMA 11. *Let Assumptions 1, 2, 3, 4, and 5 hold. Suppose that $\gamma_k \in \Gamma(X^*)$ hold and that $W_0^k \in \mathcal{R}_1$. Then, if Algorithm 2 is invoked by Algorithm 1, we have for all $t \geq 0$ that W_t^k in Algorithm 2 satisfies*

$$\|\nabla G(W_t^k)\|_F \leq \frac{\sqrt{\kappa_t}}{\beta_k}\delta_k, \tag{33a}$$

$$2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \leq \tau_t. \tag{33b}$$

In addition, Algorithm 2 terminates with the flag "converged" set to "True" and W^{k+1} satisfying

$$\|\nabla G(W^{k+1})\|_F \leq \epsilon_g, \quad \lambda_{\min}(\nabla^2 G(W^{k+1})) \geq -\epsilon_H.$$

Proof. Since $W_0^k \in \mathcal{R}_1$ and $\gamma_k \in \Gamma(X^*)$, it follows that

$$\delta_k := \sqrt{2}\gamma_k^{1/2} \geq \sigma_r^{1/2}(X^*)$$

and therefore $\operatorname{dist}(W_0^k, W^*) \leq \delta = \sigma_r^{1/2}(X^*) \leq \sqrt{2}\gamma_k^{1/2} = \delta_k$, where δ is from the (α, β, δ) regularity condition (19) and is defined to be $\sigma_r^{1/2}(X^*)$ in Theorem 5. Let $R = R(W_0^k, W^*)$ be the orthogonal matrix that minimizes $\|W^*R - W_0^k\|_F$. Then, using (14), we have

$$\begin{aligned}\sqrt{2}\|X^*\|^{1/2} = \|W^*\| \leq \|W^*\|_F = \|W^*R\|_F &\\ \leq \|W^*R - W_0^k\|_F + \|W_0^k\|_F &\\ = \operatorname{dist}(W_0^k, W^*) + \|W_0^k\|_F &\\ \leq \delta_k + \|W_0^k\|_F.&\end{aligned}$$

It follows that

$$\beta_k = \frac{2c_\beta}{(\delta_k + \|W_0^k\|_F)^2} \leq \frac{c_\beta}{\|X^*\|} = \beta. \tag{34}$$

where β is from the (α, β, δ) regularity condition (19) and defined to be $c_\beta\|X^*\|^{-1}$ in Theorem 5. Therefore, by taking a stepsize $\nu_t \leq 2\beta_k \leq 2\beta$, it follows from Lemma 3 that $W_t^k \in B(W^*, \delta)$ for all $t \geq 0$ and

$$\operatorname{dist}^2(W_t^k, W^*) \leq \operatorname{dist}^2(W_0^k, W^*)\prod_{j=0}^{t-1}(1 - 2\nu_j\alpha) \leq \delta_k^2\prod_{j=0}^{t-1}(1 - 2\nu_j\alpha_k) = \kappa_t\delta_k^2, \tag{35}$$

where we used $\alpha = c_\alpha\sigma_r(X^*) \geq c_\alpha\gamma_k = \alpha_k$, $\operatorname{dist}(W_0^k, W^*) \leq \delta_k$, and the definition of κ_t. Recalling (9) and the definition of β in our strict saddle conditions, it follows that

$$\frac{c_\beta}{\|X^*\|}\|\nabla G(W_t^k)\| \leq \operatorname{dist}(W_t^k, W^*), \quad \text{for all } t \geq 0.$$

Together with (35) this implies

$$\|\nabla G(W_t^k)\| \le \sqrt{\kappa_t}\frac{\|X^*\|}{c_\beta}\delta_k \le \frac{\sqrt{\kappa_t}}{\beta_k}\delta_k,$$

where we used $\|X^*\|/c_\beta = 1/\beta \le 1/\beta_k$ for the latter inequality, thus proving (33a). To prove (33b) (which holds for W_0^k by our local phase initialization step in Algorithm 1, by the definition of τ_0), we have from Lemma 8 and (35) that

$$\begin{aligned}
&2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \\
&\le (2L_{\nabla f} + \frac{1}{2})(2\|W_t^k\|_F + \mathrm{dist}(W_t^k, W^*))\mathrm{dist}(W_t^k, W^*) \\
&\le (2L_{\nabla f} + \frac{1}{2})(2\|W_t^k\|_F + \sqrt{\kappa_t}\delta_k)\sqrt{\kappa_t}\delta_k = \tau_t.
\end{aligned}$$

It follows from (33) that the "while" loop in Algorithm 2 terminates only when $\|\nabla G(W_t^k)\| \le \epsilon_g$ and $2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \le \epsilon_H$. Since $0 \le 1 - 2\nu_{t-1}\alpha_k < 1$, the bounds $0 \le \kappa_t \le \kappa_{t-1}$ hold for all $t = 1, 2, \ldots$, with equality holding only when $\kappa_{t-1} = 0$. Therefore, by (33), it follows by the definition of τ_t and $\|W_t^k\| \le R_{\mathcal{L}}$ for all $t \ge 0$ (see (23)) that

$$\lim_{t\to\infty} \|\nabla G(W_t^k)\| = 0, \quad \lim_{t\to\infty} 2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F = 0,$$

and thus the while loop must terminate with

$$\|\nabla G(W_{\bar{t}}^k)\| \le \epsilon_g, \quad 2\|\nabla f(X_{\bar{t}}^k)\|_F + \frac{1}{2}\|(\hat{W}_{\bar{t}}^k)^\top W_{\bar{t}}^k\|_F \le \epsilon_H,$$

for some $\bar{t} \ge 0$. Then, by Lemma 7 and $W^{k+1} = W_{\bar{t}}^k$, the final claim is proved. □

5 Complexity Analysis. This section presents our complexity results for Algorithm 1. We provide a brief "roadmap" to the sequence of results here.

We start by showing (Lemma 12) how the parameters α_k, δ_k, and β_k in the algorithm relate to the properties of the objective function and solution, in particular the key quantity $\sigma_r(X^*)$. We follow up with a result (Lemma 13) that shows that the reduction in G from a backtracking line search along the negative gradient direction is a multiple of $\|\nabla G(W)\|^2$, then apply this result to the line searches (26) and (28) (see Lemmas 14 and 15, respectively). For backtracking steps along negative curvature directions (27), we show that the reduction in G is a multiple of γ_k^3 (Lemma 16).

The next result, Lemma 17, is a bound on the number of iterations taken in Algorithm 2 when it is invoked with $\gamma_k \ge \frac{1}{2}\sigma_r(X^*)$. We then return to the main algorithm, Algorithm 1, and derive a bound on the number of non-local iterations (negative gradient or negative curvature steps) under the assumptions that $\gamma_k \ge \frac{1}{2}\sigma_r(X^*)$ and G is bounded below (Lemma 18). Lemma 19 then derives conditions under which a call to Algorithm 2 will be made that results in successful termination.

Lemma 20 is a particularly important result, showing that with high probability, we have that $\gamma_k \ge \frac{1}{2}\sigma_r(X^*)$ at all iterations, and placing a bound on the number of times that Algorithm 2 is invoked. This result leads into the main convergence results, Theorem 21 and Corollary 22, which show iteration and complexity bounds for the algorithm.

5.1 Strict Saddle Parameters. In this subsection, we present lemmas which provide bounds on the parameters α_k, β_k, δ_k, and γ_k which are generated throughout Algorithm 1 and are used to estimate the true strict saddle parameters.

LEMMA 12. *Let Assumptions 1, 2, and 3 hold. Let α_k, γ_k, δ_k, and β_k be the values of defined in Algorithm 1 (for those iterations k on which they are defined). Then, for any k such that $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds, we have*

$$\frac{1}{2}\sigma_r(X^*) \leq \gamma_k \leq \gamma_0, \tag{36a}$$

$$\frac{c_\alpha}{2}\sigma_r(X^*) \leq \alpha_k \leq c_\alpha \gamma_0, \tag{36b}$$

$$\sigma_r^{1/2}(X^*) \leq \delta_k \leq \sqrt{2}\gamma_0^{1/2}, \tag{36c}$$

$$\frac{2c_\beta}{\left(\sqrt{2}\gamma_0^{1/2} + R_{\mathcal{L}}\right)^2} \leq \beta_k \leq \frac{2c_\beta}{\sigma_r(X^*)}, \tag{36d}$$

where $R_{\mathcal{L}}$ is defined in (23).

Proof. By the definition of our algorithm, it follows that $\gamma_0 \geq \gamma_k$ for all k. In addition, by our assumption that $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds, we have proved (36a).

Noting that $\alpha_k = c_\alpha \gamma_k$, (36b) follows directly from (36a).

Now, for δ_k, we have from $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ that

$$\delta_k = \sqrt{2}\gamma_k^{1/2} \geq \sqrt{2}\left(\frac{1}{2}\sigma_r(X^*)\right)^{1/2} = \sigma_r^{1/2}(X^*),$$

while

$$\delta_k = \sqrt{2}\gamma_k^{1/2} \leq \sqrt{2}\gamma_0^{1/2},$$

proving (36c).

Recalling the definition of β_k, we have

$$\beta_k = \frac{2c_\beta}{(\delta_k + \|W^k\|_F)^2} = \frac{2c_\beta}{(\sqrt{2}\gamma_k^{1/2} + \|W^k\|_F)^2} \leq \frac{2c_\beta}{\sigma_r(X^*)}.$$

For a lower bound on β_k, note that the backtracking linesearches at each step ensure monotonicity of the iterates, so that $W^k \in \mathcal{L}_{W^0}$ for all $k \geq 0$. Thus, we have

$$\beta_k = \frac{2c_\beta}{(\delta_k + \|W^k\|_F)^2} = \frac{2c_\beta}{\left(\sqrt{2}\gamma_k^{1/2} + \|W^k\|_F\right)^2} \geq \frac{2c_\beta}{\left(\sqrt{2}\gamma_0^{1/2} + R_{\mathcal{L}}\right)^2},$$

completing our proof of (36d). ∎

5.2 Line Search Guarantees. We now provide guarantees of termination and descent for the two line searches in Algorithm 1 and the line search in Algorithm 2. We begin by providing a generic lemma for Armijo backtracking on gradient descent steps.

LEMMA 13. *Suppose that Assumptions 3 and 4 hold. Suppose that a step is computed from W using a backtracking linesearch along the negative gradient direction with a step length $\nu = \zeta\theta^l$, where l is the smallest nonnegative integer such that*

$$G(W - \nu\nabla G(W)) < G(W) - \eta\nu\|\nabla G(W)\|_F^2, \tag{37}$$

where $\eta \in (0,1)$ is the sufficient decrease parameter. Then, the backtracking line search requires at most $j \leq \tilde{j} + 1$ iterations, where

$$\tilde{j} = \left\lceil \log_\theta \left(\frac{2(1-\eta)}{L_g \zeta} \right) \right\rceil_+, \tag{38}$$

and the resulting step satisfies

$$G(W) - G(W - \nu \nabla G(W)) \geq \tilde{c} \|\nabla G(W)\|_F^2 \tag{39}$$

where ν is the step size found by the backtracking procedure and

$$\tilde{c} = \eta \min \left\{ \zeta, \frac{2\theta(1-\eta)}{L_g} \right\}. \tag{40}$$

Proof. Suppose that the maximum steplength is accepted (that is, $\nu = \zeta$). Then,

$$G(W - \zeta \nabla G(W)) < G(W) - \zeta \eta \|\nabla G(W)\|_F^2$$

so the claim holds in this case. For the remainder of the proof, we assume that $\nu < 1$. For any $l \geq 0$ such that (37) does not hold, we have from (24a) that

$$\begin{aligned} -\eta \zeta \theta^l \|\nabla G(W)\|_F^2 &\leq G(W - \zeta \theta^l \nabla G(W)) - G(W) \\ &\leq -\zeta \theta^l \langle \nabla G(W), \nabla G(W) \rangle + \frac{L_g \zeta^2 \theta^{2l}}{2} \|\nabla G(W)\|_F^2 \\ &= -\zeta \theta^l \left(1 - \frac{L_g \zeta \theta^l}{2} \right) \|\nabla G(W)\|_F^2. \end{aligned}$$

By rearranging this expression, we obtain

$$\frac{L_g}{2} \zeta \theta^l \geq 1 - \eta.$$

Therefore, for any $l \geq 0$ where (37) is not satisfied, we have

$$\theta^l \geq \frac{2(1-\eta)}{L_g \zeta} \tag{41}$$

holds. For any $l > \tilde{j}$ we have

$$\theta^l < \theta^{\tilde{j}} \leq \frac{2(1-\eta)}{L_g \zeta}$$

so (41) cannot be satisfied for any $l > \tilde{j}$ and the line search must terminate with $\nu = \zeta \theta^j$ for some $1 \leq j \leq \tilde{j} + 1$. The value of this index prior to backtracking termination satisfies (41), so we have

$$\theta^j \geq \frac{2\theta(1-\eta)}{L_g \zeta}.$$

Thus,

$$G(W) - G(W - \nu \nabla G(W)) \geq \eta \zeta \theta^j \|\nabla G(W)\|^2 \geq \frac{2\eta\theta(1-\eta)}{L_g} \|\nabla G(W)\|^2.$$

□

Lemma 13 is used directly in the next two results, which provide termination and decrease guarantees for the linesearches used in the large gradient case and in the local phase.

LEMMA 14. *Suppose that Assumptions 3 and 4 hold. Suppose that the backtracking step (26) is taken at outer iteration k. Then, the backtracking line search requires at most $j_k \leq j_{\text{grad}} + 1$ iterations, where*

$$j_{\text{grad}} := \left\lceil \log_\theta \left(\frac{2(1-\eta)}{L_g} \right) \right\rceil_+, \tag{42}$$

and the resulting step satisfies $W^{k+1} = W^k - \nu_k \nabla G(W^k)$

$$G(W^k) - G(W^{k+1}) \geq c_{\text{grad}} \|\nabla G(W^k)\|_F^2 \tag{43}$$

where

$$c_{\text{grad}} = \eta \min \left\{ 1, \frac{2\theta(1-\eta)}{L_g} \right\}. \tag{44}$$

Proof. This proof follows directly from Lemma 13 with $\zeta = 1$. ∎

LEMMA 15. *Suppose that Assumptions 1, 2, 3, and 4 hold. Suppose that the backtracking step (28) is taken at inner iteration t of outer iteration k, and that $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$. Then, the backtracking line search requires at most $j_t \leq \hat{j}_k + 1$ iterations, where*

$$\hat{j}_k := \left\lceil \log_\theta \left(\frac{1-\eta}{L_g \beta_k} \right) \right\rceil_+, \tag{45}$$

the resulting step satisfies $W_{t+1}^k = W_t^k - \nu_t \nabla G(W_t^k)$, and

$$G(W_t^k) - G(W_{t+1}^k) \geq c_{\text{local}} \|\nabla G(W_t^k)\|_F^2 \tag{46}$$

where

$$c_{\text{local}} = \eta \min \left\{ \frac{4c_\beta}{(\sqrt{2}\gamma_0^{1/2} + R_{\mathcal{L}})^2}, \frac{2\theta(1-\eta)}{L_g} \right\}.$$

Proof. This result follows from Lemma 13 with $\zeta = 2\beta_k$ and

$$\beta_k \geq \frac{2c_\beta}{(\sqrt{2}\gamma_0^{1/2} + R_{\mathcal{L}})^2},$$

which follows from Lemma 12. ∎

Next, we provide similar guarantees for the negative curvature linesearch in Algorithm 1.

LEMMA 16. *Suppose that Assumptions 3 and 4 hold. Suppose that the backtracking step (27) is taken at outer iteration k. Then, the backtracking line search requires at most $j_k \leq j_{\text{nc}} + 1$ iterations, where*

$$j_{\text{nc}} := \left\lceil \log_\theta \left(\frac{3(1-\eta)}{L_H} \right) \right\rceil_+, \tag{47}$$

and the resulting step satisfies $W^{k+1} = W^k + \nu_k D^k$

$$G(W^k) - G(W^{k+1}) \geq c_{\text{nc}} \gamma_k^3 \tag{48}$$

where

$$c_{\text{nc}} = \frac{\eta c_\gamma^3}{16} \min \left\{ 1, \left(\frac{3\theta(1-\eta)}{L_H} \right)^2 \right\}. \tag{49}$$

Proof. First, by the scaling applied to D^k in Algorithm 1, it follows that

$$-\|D^k\|_F^3 = \langle D^k, \nabla^2 G(W^k) D^k \rangle \leq -\frac{1}{8} c_\gamma^3 \gamma_k^3, \tag{50}$$

where the last inequality follows from $\langle S, \nabla^2 G(W^k) S \rangle \leq -\frac{1}{2} c_\gamma \gamma_k$ and $\|S\|_F = 1$. In addition, we have

$$\langle \nabla G(W^k), D^k \rangle \leq 0. \tag{51}$$

Suppose that the unit step is accepted (that is, $\nu_k = 1$). Then

$$G(W^k + D^k) - G(W^k) < \frac{\eta}{2} \langle D^k, \nabla^2 G(W^k) D^k \rangle \leq -\frac{\eta}{16} c_\gamma^3 \gamma_k^3 \leq -c_{\rm nc} \gamma_k^3,$$

holds so the claim holds in this case. For the remainder of the proof, we assume that $\nu_k < 1$, that is, $j_k \geq 1$. For any $j \geq 0$ such that (27) does not hold, we have from (24b), (50), and (51) that

$$\begin{aligned}
-\frac{\eta \theta^{2j}}{2} \|D^k\|_F^3 &= \frac{\eta \theta^{2j}}{2} \langle D^k, \nabla^2 G(W^k) D^k \rangle \\
&\leq G(W^k + \theta^j D^k) - G(W^k) \\
&\leq \theta^j \langle \nabla G(W^k), D^k \rangle + \frac{\theta^{2j}}{2} \langle D^k, \nabla^2 G(W^k) D^k \rangle + \frac{L_H \theta^{3j}}{6} \|D^k\|_F^3 \\
&\leq -\frac{\theta^{2j}}{2} \|D^k\|_F^3 + \frac{L_H \theta^{3j}}{6} \|D^k\|_F^3 \\
&= -\frac{\theta^{2j}}{2} \left(1 - \frac{L_H \theta^j}{3}\right) \|D^k\|_F^3
\end{aligned}$$

By rearranging this expression, we have

$$\frac{L_H}{3} \theta^j \geq 1 - \eta \quad \Rightarrow \quad \theta^j \geq \frac{3(1-\eta)}{L_H}. \tag{52}$$

For any $j > j_{\rm nc}$ we have

$$\theta^j < \theta^{j_{\rm nc}} \leq \frac{3(1-\eta)}{L_H}$$

so (52) cannot be satisfied for any $j > j_{\rm nc}$ and the line search must terminate with $\nu_k = \theta^{j_k}$ for some $1 \leq j_k \leq j_{\rm nc} + 1$. The value $j = j_k - 1$ satisfies (52), so we have

$$\theta^{j_k} \geq \frac{3\theta(1-\eta)}{L_H}.$$

Thus, by (50), we have

$$G(W^k) - G(W^{k+1}) \geq -\frac{\eta}{2} \theta^{2j_k} \langle D^k, \nabla^2 G(W^k) D^k \rangle \geq \frac{\eta}{16} \left(\frac{3\theta(1-\eta)}{L_H}\right)^2 c_\gamma^3 \gamma_k^3 \geq c_{\rm nc} \gamma_k^3.$$

Thus, the claim holds in the case of $\nu_k < 1$ also, completing the proof. ∎

5.3 Properties of Algorithm 2. This section provides a bound on the maximum number of inner iterations that may occur during the local phase, Algorithm 2.

LEMMA 17. *Let Assumptions 1, 2, 3, 4, and 5 hold and define*

$$\nu_{\min} := \frac{2\theta(1-\eta)}{L_g}. \tag{53}$$

Then, for all k such that $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds, the "while" loop in Algorithm 2 terminates in at most

$$\mathcal{T} := 2\frac{\log \hat{C} + \log(\max(\epsilon_g^{-1}, \epsilon_H^{-1}))}{\log(1/(1-\nu_{\min} c_\alpha \sigma_r(X^*)))} \tag{54}$$

iterations, where

$$\hat{C} := \max\left\{ \frac{\gamma_0^{1/2}\left(\sqrt{2}\gamma_0^{1/2} + R_{\mathcal{L}}\right)^2}{2c_\beta}, (2L_{\nabla f} + 1/2)\left(2R_{\mathcal{L}} + \sqrt{2}\gamma_0^{1/2}\right)\sqrt{2}\gamma_0^{1/2} \right\}. \tag{55}$$

Proof. By the result of Lemma 15, the backtracking line search terminates in at most $\hat{j}_k + 1$ iterations, where $\hat{j}_k$ is defined in (45). From this definition, we have

$$\nu_t \geq 2\beta_k \theta^{\hat{j}_k+1} \geq \frac{2\theta(1-\eta)}{L_g} = \nu_{\min}, \quad \text{for all } t \geq 0. \tag{56}$$

Assume for contradiction that Algorithm 2 does not terminate on or before iteration $\mathcal{T}$. Then,

$$\|\nabla G(W_{\mathcal{T}}^k)\|_F > \epsilon_g \;\text{ and/or }\; 2\|\nabla f(X_{\mathcal{T}}^k)\|_F + \frac{1}{2}\|(\hat{W}_{\mathcal{T}}^k)^\top W_{\mathcal{T}}^k\|_F > \epsilon_H$$

hold for $t = \mathcal{T}$, and for the tests at the start of the "while" loop of Algorithm 2, we have that

$$\|\nabla G(W_t^k)\|_F \leq \frac{\sqrt{\kappa_t}}{\beta_k}\delta_k, \quad \text{for all } t = 0, 1, \ldots, \mathcal{T}, \tag{57a}$$

$$2\|\nabla f(X_t^k)\|_F + \frac{1}{2}\|(\hat{W}_t^k)^\top W_t^k\|_F \leq \tau_t, \quad \text{for all } t = 0, 1, \ldots, \mathcal{T}. \tag{57b}$$

From (56) and Lemma 12, we have $1 - 2\nu_t\alpha_k \leq 1 - \nu_{\min} c_\alpha \sigma_r(X^*)$ for all t. From this observation together with $\nu_t \leq 2\beta_k$ and $\alpha_k\beta_k \leq 1/4$, we have

$$0 < \nu_{\min} c_\alpha \sigma_r(X^*) \leq 2\nu_t\alpha_k \leq 4\alpha_k\beta_k \leq 1,$$

so that $1 - \nu_{\min} c_\alpha \sigma_r(X^*) \in [0, 1)$. Thus,

$$\kappa_{\mathcal{T}} = \prod_{t=0}^{\mathcal{T}-1}(1 - 2\nu_t\alpha_k) \leq \prod_{t=0}^{\mathcal{T}-1}(1 - \nu_{\min} c_\alpha \sigma_r(X^*)) = (1 - \nu_{\min} c_\alpha \sigma_r(X^*))^{\mathcal{T}}. \tag{58}$$

Consider first the case in which termination does not occur at the "if" statement in iteration $\mathcal{T}$ because $\|\nabla G(W_{\mathcal{T}}^k)\|_F > \epsilon_g$. We then have

$$
\begin{aligned}
\epsilon_g &< \|\nabla G(W_{\mathcal{T}}^k)\|_F \\
&\leq \frac{\sqrt{\kappa_{\mathcal{T}}}}{\beta_k}\delta_k \\
&\leq (1-\nu_{\min}c_\alpha\sigma_r(X^*))^{\mathcal{T}/2}\frac{\delta_k}{\beta_k} \\
&\leq (1-\nu_{\min}c_\alpha\sigma_r(X^*))^{\mathcal{T}/2}\frac{\gamma_0^{1/2}}{2c_\beta}(\sqrt{2}\gamma_0^{1/2}+R_{\mathcal{L}})^2,
\end{aligned}
$$

where the final inequality follows from Lemma 12. Noting that $1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)) \geq 1$, we have by manipulation of this inequality that

$$
\mathcal{T} < 2\log\left(\frac{\gamma_0^{1/2}(\sqrt{2}\gamma_0^{1/2}+R_{\mathcal{L}})^2}{2c_\beta\epsilon_g}\right)/\log(1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)))
$$

which implies that

$$
\mathcal{T} < 2\frac{\log\hat{C}+\log(\epsilon_g^{-1})}{\log(1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)))},
$$

which contradicts the definition of $\mathcal{T}$.

The second possibility is that Algorithm 2 fails to terminate in the "if" statement in iteration $\mathcal{T}$ because $2\|\nabla f(X_{\mathcal{T}}^k)\|_F + \frac{1}{2}\|(\hat{W}_{\mathcal{T}}^k)^T W_{\mathcal{T}}^k\|_F > \epsilon_H$. In this case, we have

$$
\begin{aligned}
\epsilon_H &< 2\|\nabla f(X_{\mathcal{T}}^k)\|_F + \frac{1}{2}\|(\hat{W}_{\mathcal{T}}^k)^\top W_{\mathcal{T}}^k\|_F \\
&\leq \tau_{\mathcal{T}} = (2L_{\nabla f}+1/2)(2\|W_{\mathcal{T}}^k\|_F+\sqrt{\kappa_{\mathcal{T}}}\delta_k)\sqrt{\kappa_{\mathcal{T}}}\delta_k && \text{from (57b)} \\
&\leq (2L_{\nabla f}+1/2)(2\|W_{\mathcal{T}}^k\|_F+\delta_k)\sqrt{\kappa_{\mathcal{T}}}\delta_k && \text{since } \kappa_{\mathcal{T}} \leq 1 \\
&\leq (2L_{\nabla f}+1/2)(2\|W_{\mathcal{T}}^k\|_F+\delta_k)(1-\nu_{\min}c_\alpha\sigma_r(X^*))^{\mathcal{T}/2}\delta_k && \text{from (58)} \\
&\leq (1-\nu_{\min}c_\alpha\sigma_r(X^*))^{\mathcal{T}/2}(2L_{\nabla f}+1/2)(2R_{\mathcal{L}}+\sqrt{2}\gamma_0^{1/2})\sqrt{2}\gamma_0^{1/2},
\end{aligned}
$$

where the final inequality follows from $\delta_k = \sqrt{2}\gamma_k^{1/2} \leq \sqrt{2}\gamma_0^{1/2}$, Assumptions 3 and 4, and (23). By manipulating this inequality and recalling that $1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)) \geq 1$, we find that this bound implies

$$
\mathcal{T} < 2\log\left(\frac{(2L_{\nabla f}+1/2)(2R_{\mathcal{L}}+\sqrt{2}\gamma_0^{1/2})\sqrt{2}\gamma_0^{1/2}}{\epsilon_H}\right)/\log(1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)))
$$

so that (similarly to the above)

$$
\mathcal{T} < 2\frac{\log\hat{C}+\log(\epsilon_H^{-1})}{\log(1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)))},
$$

which again contradicts the definition of $\mathcal{T}$.

Since both cases lead to a contradiction, our assumption that Algorithm 2 does not terminate on or before iteration $\mathcal{T}$ cannot be true, and the result is proved. ∎

5.4 Worst Case Complexity of Algorithm 1. We now work toward our main complexity result, Theorem 21. We begin with a lemma which bounds the maximum number of large gradient and/or negative curvature iterations that can occur while $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$.

LEMMA 18. *Suppose that Assumptions 1, 2, 3, and 4 hold. Let Algorithm 1 be invoked with $\gamma_0 \geq \sigma_r(X^*)$. Then, while $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$, Algorithm 1 takes at most*

$$\mathcal{K}_{\text{large}} := \frac{8(G(W^0) - G_{\text{low}})}{\min\{c_s^2 c_{\text{grad}}, c_{\text{nc}}\}\,\sigma_r(X^*)^3} \tag{59}$$

large gradient steps and/or large negative curvature steps.

Proof. We partition the iteration indices k that are used in Algorithm 1 prior to termination as follows: K_1 contains those iteration indices for which a large gradient step is taken, K_2 contains those for which a large negative curvature step is taken, and K_3 contains those for which the local phase is initialized.

By Lemma 14 we have for all $k \in K_1$ that

$$G(W^k) - G(W^{k+1}) \geq c_{\text{grad}}\|\nabla G(W^k)\|_F^2 \geq c_s^2 c_{\text{grad}}\gamma_k^3,$$

where c_{grad} is defined in (44). Similarly, by Lemma 16, for all $k \in K_2$, we have

$$G(W^k) - G(W^{k+1}) \geq c_{\text{nc}}\gamma_k^3,$$

where c_{nc} is defined in (49).

Now, consider $k \in K_3$. On iterations where the local phase is initialized but not invoked (that is, the condition in the "if" statement immediately prior to the call to Algorithm 2 is not satisfied), then $G(W^k) - G(W^{k+1}) = 0$. On iterations where the local phase is invoked, by the definition of T_k in Algorithm 1 and the result of Lemma 15, it follows that

$$G(W^k) - G(W^{k+1}) = \sum_{t=0}^{T_k-1} G(W_t^k) - G(W_{t+1}^k) \geq \sum_{t=0}^{T_k-1} c_{\text{local}}\|\nabla G(W_t^k)\|_F^2 \geq 0.$$

Thus, $G(W^k) - G(W^{k+1}) \geq 0$ holds for all $k \in K_3$.

By defining $K = K_1 \cup K_2 \cup K_3$, we have

$$\begin{aligned} G(W^0) - G(W^{|K|}) &= \sum_{i=0}^{|K|}(G(W^i) - G(W^{i+1})) \\ &\geq \sum_{i\in K_1}(G(W^i) - G(W^{i+1})) + \sum_{j\in K_2}(G(W^j) - G(W^{j+1})) \\ &\geq \sum_{i\in K_1} c_s^2 c_{\text{grad}}\gamma_i^3 + \sum_{j\in K_2} c_{\text{nc}}\gamma_j^3 \\ &\geq \sum_{k\in K_1\cup K_2} \min\{c_s^2 c_{\text{grad}}, c_{\text{nc}}\}\gamma_k^3. \end{aligned}$$

By assumption, we have $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$, so that

$$G(W^0) - G(W^{|K|}) \geq \frac{1}{8}\min\{c_s^2 c_{\text{grad}}, c_{\text{nc}}\}\sigma_r(X^*)^3|K_1 \cup K_2|.$$

Since $G(W^{|K|}) \geq G_{\text{low}}$, we have

$$|K_1 \cup K_2| \leq \frac{8(G(W^0) - G_{\text{low}})}{\min\{c_s^2 c_{\text{grad}}, c_{\text{nc}}\}\sigma_r(X^*)^3} = \mathcal{K}_{\text{large}},$$

proving our claim. ∎

Next, we show that if γ_k is close to $\sigma_r(X^*)$ and W^k is in the region $\mathcal{R}_1$, then provided that Procedure 3 certifies a near-positive-definite Hessian, Algorithm 2 will be called and successful termination of Algorithm 1 will ensue.

LEMMA 19. *Let Assumptions 1, 2, 3, 4, and 5 hold. At iteration k, suppose that both $\gamma_k \in \Gamma(X^*)$ and $W^k \in \mathcal{R}_1$ hold and that Procedure 3 certifies that $\lambda_{\min}(\nabla^2 G(W^k)) \geq -c_\gamma \gamma_k$. Then Algorithm 1 terminates at a point W^{k+1} that satisfies approximate optimality conditions (5).*

Proof. By the definitions of $\Gamma(X^*)$, α_k, and δ_k, it follows that $\alpha_k = c_\alpha \gamma_k \leq c_\alpha \sigma_r(X^*) = \alpha$, and $\delta_k = \sqrt{2}\gamma_k^{1/2} \geq \sigma_r^{1/2}(X^*) = \delta$. Letting $R = R(W^k, W^*) \in \mathcal{O}_r$ be the orthogonal matrix that minimizes $\|W^* R - W^k\|_F$, we have from (14) and $W^k \in \mathcal{R}_1$ that

$$\begin{aligned}\sqrt{2}\|X^*\|^{1/2} = \|W^*\| \leq \|W^*\|_F = \|W^* R\|_F \\ &\leq \|W^* R - W^k\|_F + \|W^k\|_F \\ &= \text{dist}(W^k, W^*) + \|W_0^k\|_F \\ &\leq \delta_k + \|W^k\|_F\end{aligned}$$

so that

$$\beta_k = \frac{2c_\beta}{(\delta_k + \|W_0^k\|_F)^2} \leq \frac{c_\beta}{\|X^*\|} = \beta.$$

Since $\alpha\beta \leq \frac{1}{4}$ holds by definition, it follows that $\alpha_k \beta_k \leq \frac{1}{4}$ is satisfied so that the first condition of the "if" statement prior to the local phase of Algorithm 1 holds. Now, by (9) and $W^k \in \mathcal{R}_1$, $\|\nabla G(W^k)\| \leq \text{dist}(W^k, W^*)/\beta$ holds. Thus,

$$\|\nabla G(W^k)\| \leq \frac{\delta}{\beta} \leq \frac{\delta_k}{\beta_k},$$

is satisfied, so the second condition of the "if" statement prior to the local phase of Algorithm 1 also holds. Finally, by Lemma 8 and $\text{dist}(W^k, W^*) \leq \delta_k$, we have

$$2\|\nabla f(X^k)\|_F + \frac{1}{2}\|(\hat{W}^k)^\top W^k\|_F \leq (2L_{\nabla f} + 1/2)\,(2\|W^k\|_F + \delta_k)\delta_k,$$

so that the final condition of the "if" statement also holds and Algorithm 2 will be invoked at W^k. Thus, by Lemma 11, Algorithm 1 terminates at W^{k+1} that satisfies (5). ∎

Now we show that with high probability, $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds for all k.

LEMMA 20. *Suppose that Assumptions 1, 2, 3, 4, and 5 hold. Let Algorithm 1 be invoked with $\gamma_0 \geq \sigma_r(X^*)$. Then, with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$ (where $\mathcal{K}_{\text{large}}$ is defined in Lemma 18), we have that*

$$\gamma_k \geq \frac{1}{2}\sigma_r(X^*), \quad \textit{for all } k,$$

and Algorithm 2 is invoked at most

$$\mathcal{K}_{\text{local}} := \log_2\left(\frac{2\gamma_0}{\sigma_r(X^*)}\right) \quad \textit{times.} \tag{60}$$

Proof. By the definitions of Algorithm 1 and $\Gamma(X^*)$, it is clear that $\gamma_j < \frac{1}{2}\sigma_r(X^*)$ can only occur at an iteration j such that $\gamma_k \in \Gamma(X^*)$ holds for some $k < j$. Let $\hat{K}$ denote the set of (consecutive) iterations for which $\gamma_k \in \Gamma(X^*)$.

Consider any iteration $k \in \hat{K}$. Due to the structure of Algorithm 1, γ_k will be halved only on iterations k for which $\|\nabla G(W^k)\|_F < c_s\gamma_k^{3/2}$ is satisfied and Procedure 3 certifies that $\lambda_{\min}(\nabla^2 G(W^k)) \geq -c_\gamma\gamma_k$. From Lemma 10, $\|\nabla G(W^k)\| < c_s\gamma_k^{3/2}$ cannot hold for $W^k \in \mathcal{R}_3$ and $\gamma_k \in \Gamma(X^*)$, so γ_k cannot be halved on such iterations. Next, consider $k \in \hat{K}$ such that $W^k \in \mathcal{R}_1$ and Procedure 3 certifies that $\lambda_{\min}(\nabla^2 G(W^k)) \geq -c_\gamma\gamma_k$. In this case, Algorithm 1 terminates at W^{k+1}, by Lemma 19. Thus, it follows that γ_k can be reduced to a level below $\frac{1}{2}\sigma_r(X^*)$ only if there is some iteration $k \in \hat{K}$ such that $W^k \in \mathcal{R}_2$, $\|\nabla G(W^k)\|_F < c_s\gamma_k^{3/2}$ and Procedure 3 certifies that $\lambda_{\min}(\nabla^2 G(W^k)) \geq -c_\gamma\gamma_k$. Since for $\gamma_k \in \Gamma(X*)$ and $W^k \in \mathcal{R}_2$, we have $\lambda_{\min}(\nabla^2 G(W^k)) \leq -c_\gamma\sigma_r(X^*) < -c_\gamma\gamma_k$, this "certification" by Procedure 3 would be erroneous, an event that happens with probability at most ρ. Otherwise, a negative curvature backtracking step is taken.

Since the maximum number of large negative curvature steps that can occur while $\gamma_k \in \Gamma(X^*)$ is bounded by $\mathcal{K}_{\text{large}}$ (Lemma 18), there are at most $\mathcal{K}_{\text{large}}$ iterations for which both $W^k \in \mathcal{R}_2$ and $\gamma_k \in \Gamma(X^*)$ hold. It follows that with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$, Procedure 3 does not certify that $\lambda_{\min}(\nabla^2 G(W^k) \geq -c_\gamma\gamma_k$ while $W^k \in \mathcal{R}_2$ for all $k \in \hat{K}$. This further implies that with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$, $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds for all k.

The second claim follows immediately from the first claim together with the facts that Algorithm 2 is invoked at least once for each value of γ_k, and that successive values of γ_k differ by factors of 2. ∎

We are now ready to state our iteration complexity result.

THEOREM 21. *Suppose that Assumptions 1, 2, 3, 4, and 5 hold. Then, with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$, Algorithm 1 terminates in at most*

$$\mathcal{K}_{\text{outer}} := \mathcal{K}_{\text{large}} + \mathcal{K}_{\text{local}} \tag{61}$$

outer iterations (where $\mathcal{K}_{\text{large}}$ and $\mathcal{K}_{\text{local}}$ are defined in (59) and (60), resp.) and

$$\mathcal{K}_{\text{total}} := \mathcal{K}_{\text{large}} + 2\frac{\log \hat{C} + \log\max(\epsilon_g^{-1}, \epsilon_H^{-1})}{\log(1/(1-\nu_{\min}c_\alpha\sigma_r(X^*)))}\mathcal{K}_{\text{local}} \tag{62}$$

total iterations at a point satisfying (5), where $\hat{C}$ is defined in (55) and $\nu_{\min}$ is defined in (53).

Proof. By Lemma 20, with probability $(1-\rho)^{\mathcal{K}_{\text{large}}}$, $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ holds for all k and Algorithm 1 invokes Algorithm 2 at most $\mathcal{K}_{\text{local}}$ times. Thus, by Lemma 18, it follows that Algorithm 1 takes at most $\mathcal{K}_{\text{large}}$ large gradient steps and/or large negative curvature iterations with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$. Altogether, this implies that with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$, the maximum number of outer iterations in Algorithm 1 is bounded by $\mathcal{K}_{\text{large}} + \mathcal{K}_{\text{local}}$, proving (61).

The bound (62) follows by combining Lemma 17 (which bounds the number of iterations in Algorithm 2 at each invocation) with the fact that Algorithm 2 is involved at most $\mathcal{K}_{\text{local}}$ times, with probability at least $(1-\rho)^{\mathcal{K}_{\text{large}}}$. ∎

We turn our attention now to providing a complexity in terms of gradient evaluations and/or Hessian vector products. One more assumption is needed on Procedure 3.

ASSUMPTION 6. *For every iteration k at which Algorithm 1 calls Procedure 3, and for a specified failure probability ρ with $0 \leq \rho \ll 1$, Procedure 3 either certifies that $\nabla^2 G(W^k) \succeq -\epsilon I$ or finds a vector of curvature smaller than $-\epsilon/2$ in at most*

$$N_{\text{meo}} := \min\left\{N, 1 + \left\lceil \mathcal{C}_{\text{meo}} \epsilon^{-1/2} \right\rceil\right\} \tag{63}$$

Hessian-vector products (where $N = (m+n)r$ is the number of elements in W^k), with probability $1-\rho$, where $\mathcal{C}_{\text{meo}}$ depends at most logarithmically on ρ and ϵ.

Assumption 6 encompasses the strategies we mentioned in Section 4.2. Assuming the bound U_H on $\|\nabla^2 G(W)\|$ to be available, for the Lanczos method with a random starting vector, (63) holds with $\mathcal{C}_{\text{meo}} = \ln(2.75(nr+mr)/\rho^2)\sqrt{U_H}/2$. When a bound on $\|\nabla^2 G(W)\|$ 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 [15].

Under this assumption, we have the following corollary regarding the maximum number of gradient evaluations/Hessian vector products required by Algorithm 1 to find a point satisfying our approximate second-order conditions (5).

COROLLARY 22. *Suppose that the assumptions of Theorem 21 are satisfied, and that Assumption 6 is also satsified with N_{meo} defined in (63). Then, with probability $(1-\rho)^{\mathcal{K}_{\text{large}}}$, the number of gradient evaluations and/or Hessian-vector products required by Algorithm 1 to output an iterate satisfying (5) is at most*

$$N_{\text{meo}}\left(\mathcal{K}_{\text{large}} + \mathcal{K}_{\text{local}}\right) + 2\frac{\log \hat{C} + \log\max(\epsilon_g^{-1}, \epsilon_H^{-1})}{\log(1/(1-\nu_{\min} c_\alpha \sigma_r(X^*)))}\mathcal{K}_{\text{local}}, \tag{64}$$

and N_{meo} satisfies the upper bound

$$N_{\text{meo}} \leq \min\left\{(n+m)r, 1 + \left\lceil \sqrt{2}\mathcal{C}_{\text{meo}} c_\gamma^{-1/2} \sigma_r^{-1/2}(X^*) \right\rceil\right\}.$$

Proof. All large gradient steps and local iterations (iterations in Algorithm 2) require a single gradient evaluation. Thus, the number of gradient evaluations in the local phase is equal to the number of iterations in this phase.

Procedure 3 is invoked at every large negative curvature iteration and before each time the local phase is tried. With probability $(1-\rho)^{\mathcal{K}_{\text{large}}}$, the maximum number of large negative curvature iterations is bounded by $\mathcal{K}_{\text{large}}$ while the maximum number of times the local phase is entered is bounded by $\mathcal{K}_{\text{local}}$. In addition, by Assumption 6, Procedure 3 requires at most N_{meo} Hessian-vector products. Thus, the maximum number of gradient evaluations and/or Hessian-vector products required is bounded by the quantity in (64).

Since $\gamma_k \geq \frac{1}{2}\sigma_r(X^*)$ with probability $(1-\rho)^{\mathcal{K}_{\text{large}}}$, it follows that

$$\begin{aligned} N_{\text{meo}} &= \min\left\{N, 1 + \left\lceil \mathcal{C}_{\text{meo}} c_\gamma^{-1/2} \gamma_k^{-1/2} \right\rceil\right\} \\ &\leq \min\left\{(n+m)r, 1 + \left\lceil \sqrt{2}\mathcal{C}_{\text{meo}} c_\gamma^{-1/2} \sigma_r^{-1/2}(X^*) \right\rceil\right\}, \end{aligned}$$

verifying the bound on N_{meo}. ∎

6 Conclusion. We have described an algorithm that finds an approximate second-order point for robust strict saddle functions. This method does not require knowledge of the strict saddle parameters that define the optimization landscape or a specialized initialization procedure. By contrast with other methods proposed

recently for finding approximate second-order points for nonconvex smooth functions (see, for example, [3, 15]), the complexity is not related to a negative power of the optimality tolerance parameter, but depends only logarithmically on this quantity. The iteration complexity and the gradient complexity depend instead on a negative power of $\sigma_r(X^*)$, the smallest nonzero singular value of the (rank-r) minimizer of f.

One future research direction lies in investigating whether accelerated gradient methods are suitable for use in the local phase of Algorithm 1. While effective in practice [13], little is known about the convergence rate of these algorithms when the (α, β, δ)-regularity condition holds. In [22], the authors showed that under certain parameter settings, accelerated gradient methods converge at a linear rate. However, due to the techniques used, it is difficult to understand from this paper when this linear rate substantially improves over the convergence rate of gradient descent, leaving open interesting future research directions.

REFERENCES

[1] S. Bhojanapalli, B. Neyshabur, and N. Srebro, *Global optimality of local search for low rank matrix recovery*, in Advances in Neural Information Processing Systems, 2016, pp. 3873–3881.

[2] E. J. Candes, X. Li, and M. Soltanolkotabi, *Phase retrieval via Wirtinger flow: Theory and algorithms*, IEEE Transactions on Information Theory, 61 (2015), pp. 1985–2007.

[3] Y. Carmon, J. C. Duchi, O. Hinder, and A. Sidford, *Accelerated methods for non-convex optimization*, SIAM J. Optim., 28 (2018), pp. 1751–1772.

[4] Y. Chen, Y. Chi, J. Fan, and C. Ma, *Gradient descent with random initialization: Fast global convergence for nonconvex phase retrieval*, Mathematical Programming, 176 (2019), pp. 5–37.

[5] R. Ge, F. Huang, C. Jin, and Y. Yuan, *Escaping from saddle points - Online stochastic gradient for tensor decomposition*, in Volume 40: Conference on Learning Theory, 3-6 July 2015, Paris, France, PMLR, 2015, pp. 797–842.

[6] R. Ge, C. Jin, and Y. Zheng, *No spurious local minima in nonconvex low rank problems: A unified geometric analysis*, in Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 1233–1242.

[7] D. Gilboa, S. Buchanan, and J. Wright, *Efficient dictionary learning with gradient descent*, in International Conference on Machine Learning, 2019, pp. 2252–2259.

[8] A. Griewank and A. Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, Frontiers in Applied Mathematics, SIAM, Philadelphia, PA, second ed., 2008.

[9] R. H. Keshavan, A. Montanari, and S. Oh, *Matrix completion from a few entries*, IEEE transactions on information theory, 56 (2010), pp. 2980–2998.

[10] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, *Gradient descent only converges to minimizers*, in Conference on learning theory, 2016, pp. 1246–1257.

[11] X. Li, S. Ling, T. Strohmer, and K. Wei, *Rapid, robust, and reliable blind deconvolution via nonconvex optimization*, Applied and computational harmonic analysis, 47 (2019), pp. 893–934.

[12] S. Paternain, A. Mokhtari, and A. Ribeiro, *A Newton-based method for nonconvex optimization with fast evasion of saddle points*, SIAM Journal on Optimization, 29 (2019), pp. 343–368.

[13] E. J. R. Pauwels, A. Beck, Y. C. Eldar, and S. Sabach, *On fienup methods for sparse phase retrieval*, IEEE Transactions on Signal Processing, 66 (2017), pp. 982–991.

[14] B. Recht, M. Fazel, and P. A. Parrilo, *Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization*, SIAM review, 52 (2010), pp. 471–501.

[15] C. W. Royer, M. O'Neill, and S. J. Wright, *A Newton-CG algorithm with complexity guarantees for smooth unconstrained optimization*, Math. Program., (2019).

[16] J. Sun, Q. Qu, and J. Wright, *When are nonconvex problems not scary?*, arXiv preprint arXiv:1510.06096, (2015).

[17] ———, *Complete dictionary recovery over the sphere i: Overview and the geometric picture*, IEEE Transactions on Information Theory, 63 (2016), pp. 853–884.

[18] ———, *Complete dictionary recovery over the sphere i: Overview and the geometric picture*,

IEEE Transactions on Information Theory, 63 (2016), pp. 853–884.
[19] ———, *A geometric analysis of phase retrieval*, Foundations of Computational Mathematics, 18 (2018), pp. 1131–1198.
[20] R. Sun and Z.-Q. Luo, *Guaranteed matrix completion via non-convex factorization*, IEEE Transactions on Information Theory, 62 (2016), pp. 6535–6579.
[21] S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht, *Low-rank solutions of linear matrix equations via procrustes flow*, in Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML'16, JMLR.org, 2016, p. 964–973.
[22] H. Xiong, Y. Chi, B. Hu, and W. Zhang, *Analytical convergence regions of accelerated gradient descent in nonconvex optimization under regularity condition*, Automatica, 113 (2020), p. 108715.
[23] Z. Zhu, Q. Li, G. Tang, and M. B. Wakin, *The global optimization geometry of low-rank matrix optimization*, arXiv preprint arXiv:1703.01256, (2017).

Appendix A. Proof of (13). By the definition of W^* and the operator norm:

$$\|W^*\|^2 = \lambda_{\max}(W^*(W^*)^\top) = \max_z \frac{z^\top W^*(W^*)z}{\|z\|^2}$$
$$= \max_{z_\Phi, z_\Psi} \frac{\begin{bmatrix} z_\Phi \\ z_\Psi \end{bmatrix}^\top \begin{bmatrix} \Phi\Sigma\Phi^\top & \Phi\Sigma\Psi^\top \\ \Psi\Sigma\Phi^\top & \Psi\Sigma\Psi^\top \end{bmatrix} \begin{bmatrix} z_\Phi \\ z_\Psi \end{bmatrix}}{\|z_\Phi\|^2 + \|z_\Psi\|^2},$$

where we have partitioned the vector z in an obvious way. Letting ϕ_1 denote the first left singular vector of X^* and ψ_1 denote the first right singular vector of X^*. Then, it is clear that the maximum is obtained by setting $z_\Phi = \phi_1$ and $z_\Psi = \psi_1$, so that

$$\max_{z_\Phi, z_\Psi} \frac{\begin{bmatrix} z_\Phi \\ z_\Psi \end{bmatrix}^\top \begin{bmatrix} \Phi\Sigma\Phi^\top & \Phi\Sigma\Psi^\top \\ \Psi\Sigma\Phi^\top & \Psi\Sigma\Psi^\top \end{bmatrix} \begin{bmatrix} z_\Phi \\ z_\Psi \end{bmatrix}}{\|z_\Phi\|^2 + \|z_\Psi\|^2} = \frac{4\sigma_1(X^*)}{2} = 2\sigma_1(X^*) = 2\|X^*\|.$$

To prove the second result, the definition of the Frobenius norm gives

$$\|W^*(W^*)^\top\|_F = \sqrt{\sum_{i=1}^r \lambda_i \left(W^*(W^*)^\top\right)^2}.$$

Now, let ψ_i be the i-th left singular vector of X^* and ϕ_i be the i-th right singular vector of X^*. It is clear that we obtain an eigenvector for the ith eigenvalue of $W^*(W^*)^\top$ by setting $z_i = \begin{bmatrix} \phi_i \\ \psi_i \end{bmatrix}$. Similar to the calculation above for z_1, we have

$$\lambda_i\left(W^*(W^*)^\top\right) = \frac{z_i^\top W^*(W^*)^\top z_i}{\|z_i\|^2} = 2\sigma_i(X^*)$$

and thus

$$\sqrt{\sum_{i=1}^r \lambda_i \left(W^*(W^*)^\top\right)^2} = \sqrt{\sum_{i=1}^r (2\sigma_i(X^*))^2} = 2\sqrt{\sum_{i=1}^r \sigma_i^2(X^*)} = 2\|X^*\|_F.$$