Level Set Teleportation: An Optimization Perspective

Aaron Mishkin
Stanford University

Alberto Bietti
CCM, Flatiron Institute

Robert M. Gower
CCM, Flatiron Institute

Abstract

We study level set teleportation, an optimization routine which tries to accelerate gradient descent (GD) by maximizing the gradient norm over a level set of the objective. While teleportation intuitively speeds-up GD via bigger steps, current work lacks convergence theory for convex functions, guarantees for solving the teleportation operator, and even clear empirical evidence showing this acceleration. We resolve these open questions. For convex functions satisfying Hessian stability, we prove that GD with teleportation obtains a combined sub-linear/linear convergence rate which is strictly faster than GD when the optimality gap is small. This is in sharp contrast to the standard (strongly) convex setting, where teleportation neither improves nor worsens convergence. To evaluate teleportation in practice, we develop a projected-gradient method requiring only Hessian-vector products. We use this to show that gradient methods with access to a teleportation oracle out-perform their standard versions on a variety of problems. We also find that GD with teleportation is faster than truncated Newton methods, particularly for non-convex optimization.

1 INTRODUCTION

We consider minimizing a differentiable function f. When ∇f is L-Lipschitz continuous (L-smooth), the *descent lemma* (D. P. Bertsekas, 1997) implies that gradient descent (GD) with step-size η_k makes progress proportional to the squared-norm of the gradient,

$$f(w_{k+1}) \leq f(w_k) - \eta_k \left(1 - \frac{\eta_k L}{2}\right) \|\nabla f(w_k)\|_2^2. \quad (1)$$

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

If all other quantities are held constant, then increasing the gradient-norm increases the one-step progress guaranteed by smoothness. As a result, optimization trajectories which maximize the observed gradients may converge faster than their naive counterparts. This same argument has been used to incrementally grow neural networks (Evci et al., 2022).

Level set teleportation attempts to leverage the descent lemma by maximizing the gradient norm without changing the objective value. At an iteration k satisfying a pre-determined scheduling rule, level set teleportation solves the maximization problem,

$$w_k^+ \in \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) = f(w_k), \quad (2)$$

where the feasible set $\mathcal{L}_k := \{w : f(w) = f(w_k)\}$ is the level set of f at $f(w_k)$. When f is strongly convex, the Newton and gradient directions at w_k^+ coincide (Zhao, Dehmamy, et al., 2023) and the next gradient step is equivalent to a scaled Newton update (see Figure 1). For some functions, like quadratics, the gradient norm is also maximized over the level sets along the gradient flow from w_k^+ to w^* (Zhao, Gower, et al., 2023).

Although these properties suggest that level set teleportation may be an effective heuristic for improving the convergence rate of gradient methods, existing theoretical evidence is surprisingly limited. Zhao, Gower, et al. (2023) use the connection to Newton's method to show GD with teleportation obtains mixed linear/quadratic convergence, but their rate applies only to strongly convex functions and requires line-search. In contrast, teleportation for non-convex functions shows no improvement in convergence speed; rather, it strengthens standard sub-linear rates to guarantee all gradients in the level set are concentrating (Zhao, Gower, et al., 2023). To the best of our knowledge, no rates on sub-optimality have been shown for the convex setting.

Teleportation also suffers from a disconnect between theory and practice: while existing guarantees require solving Equation (2) over the full level set, empirical studies instead focus on *symmetry teleportation* (Zhao, Dehmamy, et al., 2023; Zhao, Gower, et al., 2023), which restricts optimization to group symmetries of the

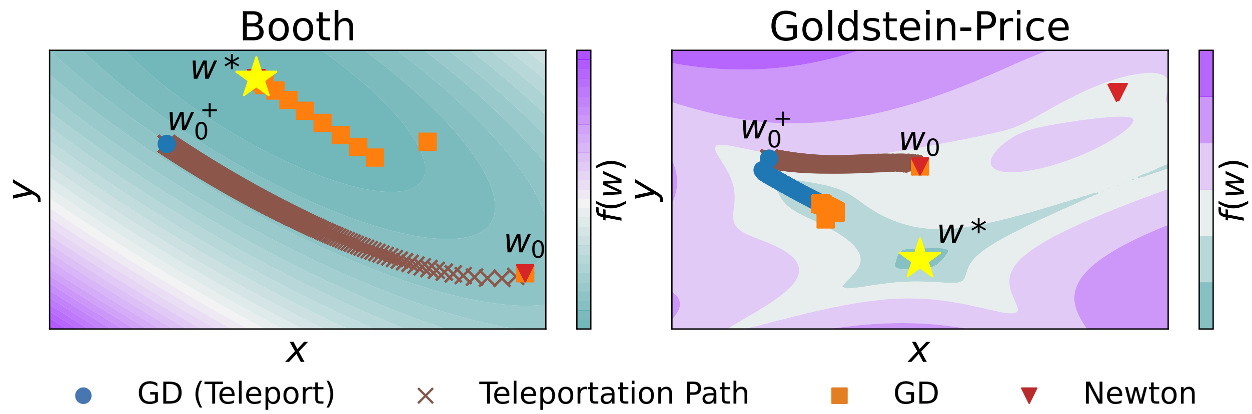

Figure 1: Initializing by level set teleportation two test functions. The Booth function is a convex quadratic and teleportation aligns w_0^+ with the maximum eigenvalue-eigenvector pair. The next iteration of GD is equivalent to a Newton update and converges in one step. The Goldstein-Price function is non-convex and teleporting pushes w_0^+ up a narrow "valley" from which convergence is slow. Newton's method diverges on the non-convex function.

objective. For example, neural networks with positively homogeneous activation functions are invariant under certain positive rescaling operators. Although these symmetries may not fully capture $\mathcal{L}_k$, optimizing over parameterized group operators can be very fast and yields an approximate form of level set teleportation (Zhao, Dehmamy, et al., 2023).

In this paper, we conduct a rigorous theoretical and empirical evaluation of level set teleportation for convex functions. We prove that while teleportation does not impair convergence, additional assumptions are necessary to obtain improved rates for both convex and strongly-convex f. Our novel proof technique, which merges the linear rate from Newton steps and the sub-linear rate from standard GD, shows that teleportation is most effective when the optimality gap is small. This means teleportation is best used when near termination in order to obtain highly accurate solutions.

In contrast to previous work, we also provide an algorithm for exactly solving the level set teleportation problem using sequential quadratic programming (SQP). Our method only requires Hessian-vector products and resembles a step of projected GD. The procedure is parameter-free—the step-size is selected automatically using a merit function—and convergence is guaranteed via connections to SQP methods. Crucially, we leverage our algorithm to provide the first rigorous empirical evaluation of level set teleportation. This is summarized along with our other contributions as follows.

- We prove that GD with teleportation cannot improve upon the convergence rate of standard GD when f is strongly convex unless adaptive step-sizes are used. In the convex setting, we show that the connection to Newton's method may fail and convergence of GD with teleportation depends on the diameter of the initial sub-level set.
- For convex functions satisfying Hessian stability, our novel proof combines the sub-linear progress from standard GD steps and the linear progress from GD steps after teleporting to obtain a convergence rate which is strictly faster than $O(1/K)$.
- We develop a tuning-free algorithm for solving teleportation problems and use it to show that stochastic and deterministic gradient methods converge faster with access to a teleportation oracle.

1.1 Additional Related Work

Level Set Teleportation: Armenta, Judge, et al. (2020) and Armenta and Jodoin (2021) first used group symmetries to randomly perturb or "teleport" the weights of neural networks during training. Zhao, Dehmamy, et al. (2023) then proposed to optimize over parametric symmetries for which teleportation is fast or even closed form, while Zhao, Ganev, et al. (2023) studied more sophisticated data-dependent symmetries. While our focus is on convex problems, Zhao, Gower, et al. (2023) analyze convergence of GD with teleportation for general non-convex functions and show that the gradient norm converges uniformly over the sub-level set, strengthening standard guarantees. They also give rates under the PL condition (Karimi et al., 2016).

Symmetries in Optimization: Teleportation is closely connected to sharp/flat minima in deep learning (Hochreiter and Schmidhuber, 1997; Keskar et al., 2017; Dinh et al., 2017). Sharpness aware minimization (Foret et al., 2021) biases optimization towards regions with low curvature, while level set teleportation actively

Algorithm 1 GD with Teleportation

Inputs: w_0; step-sizes η_k; block indices $\mathcal{B}$, sizes b_k.
$\mathcal{T} \leftarrow \bigcup_{k \in \mathcal{B}} \{k, k+1, \ldots, k+b_k-1\}$
for $k \in \{0, \ldots, K\}$ **do**
  if $k \in \mathcal{T}$ **then**
    $w_k^+ \in \arg\max \{\|\nabla f(w)\|_2 : f(w) \leq f(w_k)\}$
  else
    $w_k^+ \leftarrow w_k$
  end if
  if line-search **then**
    $g_k \leftarrow \|\nabla f(w_k^+)\|_2^2$
    $\eta_k \leftarrow \max\left\{\eta : f(w_k^+ - \eta \nabla f(w_k^+)) \leq f(w_k^+) - \frac{\eta}{2} g_k\right\}$
  end if
  $w_{k+1} \leftarrow w_k^+ - \eta_k \nabla f(w_k^+)$
end for
Output: w_K

finds large gradients. In contrast to GD with teleportation, Neyshabur et al. (2015) suggest model invariances harm optimization and propose Path-SGD, a gradient method which is invariant to rescaling symmetries in neural networks. Similarly, Bamler and Mandt (2018) leverage group operators by separating optimization into directions with symmetries and those without.

Initialization: While initialization methods for neural networks typically balance layer norms to prevent exploding or vanishing gradients (Narkhede et al., 2022), teleportation skews layers to maximize the gradient norm. Our work is related to Saxe et al. (2014) and Min et al. (2021), who study initialization in linear networks, and Tarmoun et al. (2021), who propose symmetry operators for initializing linear models with two layers. When used for initialization, teleportation can be thought of as data dependent pre-training (Lehtokangas and Saarinen, 1998; Bengio et al., 2006).

2 LEVEL SET TELEPORTATION

We begin by considering the convergence of gradient methods with intermittent teleportation. We assume that f is L-smooth, has at least one minimizer w^*, and is coercive. Coercivity holds if and only if the level sets of f are bounded and is sufficient to guarantee that the teleportation problem admits a finite solution; this property is easy to check in practice since boundedness of one level set is sufficient for all level sets to be bounded when f is convex (D. Bertsekas, 2009, Prop. 1.4.5). We show in Proposition B.2 that teleportation is ill-posed for non-coercive neural network problems.

Instead of solving Equation (2), we focus on the more general sub-level set teleportation problem,

$$w_k^+ = \arg\max_w \frac{1}{2}\|\nabla f(w)\|_2^2 \quad \text{s.t.} \quad f(w) \leq f(w_k), \tag{3}$$

where the feasible set is $\mathcal{S}_k = \{w : f(w) \leq f(w_k)\}$. For convex f, Equation (3) admits at least one solution on the boundary of $\mathcal{S}_k$; if f is strictly convex, then every solution is on the boundary and the relaxation is equivalent to level set teleportation (Lemma A.2). However, sub-level set teleportation is acceptable even for general functions since our goal is to minimize f.

Although sub-level set teleportation can be used with any iterative optimizer, we focus on (stochastic) gradient methods. To simplify the presentation, we group teleportation steps into non-overlapping blocks. Let $\mathcal{B}$ be the starting iterations of the blocks, b_k the length of the block starting at iteration k, and $\mathcal{T}$ the complete teleportation schedule (tele-schedule) so that $k \in \mathcal{B}$ implies $k, k+1, \ldots, k+b_k-1 \in \mathcal{T}$. Non-overlapping means $k \in \mathcal{B}$ implies $k-1 \notin \mathcal{T}$. See Algorithm 1.

Given initialization w_0, let w^* be the projection of w_0 onto the optimal set. The initial sub-level set $\mathcal{S}_0 = \{w : f(w) \leq f(w_0)\}$ and diameter $R = \sup\{\|w^* - w\|_2 : w \in \mathcal{S}_0\}$ play a critical role in our analysis. This is because sub-level set teleportation is not non-expansive (Definition A.1), meaning $\|w_k^+ - w^*\|_2 > \|w_k - w^*\|_2$ may occur. However, the optimality gaps $\delta_k := f(w_k) - f(w^*)$ contract if the step-sizes are sufficiently small, so the iterates remain in $\mathcal{S}_0$.

Lemma 2.1. *If f is L-smooth and $\eta_k < 2/L$, then GD with tele-schedule $\mathcal{T}$ satisfies $\delta_{k+1} \leq \delta_k$, $w_k \in \mathcal{S}_0$, and $\|w_k - w^*\|_2 \leq R$ for every $k \in \mathbb{N}$.*

If f is μ-strongly convex (μ-SC), meaning $\forall\, w, w' \in \mathbb{R}^d$,

$$f(w) \geq f(w') + \langle \nabla f(w'), w - w' \rangle + \frac{\mu}{2}\|w - w'\|_2^2, \tag{4}$$

then we also have progress in $\Delta_k = \|w_k - w^*\|_2^2$.

Lemma 2.2. *Suppose f is L-smooth and μ-SC. If $k \in \mathcal{T}$, then GD with step-size η_k satisfies*

$$\Delta_{k+1} \leq \max\left\{(1-\eta_k L)^2, (1-\eta_k \mu)^2\right\} \|w_k^+ - w^*\|_2^2. \tag{5}$$

Unfortunately, $\|w_k^+ - w^*\|_2^2$ and Δ_k differ by a factor of L/μ in the worst case since teleportation is expansive, leaving us to prove a weaker rate in the general setting.

Proposition 2.3. *If f is L-smooth and μ-SC, then GD with $\eta_k < 2/L$ and tele-schedule $\mathcal{T}$ satisfies,*

$$\delta_K \leq \prod_{i=0}^{K-1} \left[1 - 2\mu\eta_i \left(1 - \frac{\eta_i L}{2}\right)\right] \delta_0. \tag{6}$$

Moreover, if the teleportation operator is non-expansive, then for any step-sizes η_k we have,

$$\delta_K \leq \frac{L}{\mu} \prod_{i=0}^{K-1} \left[\max\left\{(1-\eta_i L)^2, (1-\eta_i \mu)^2\right\}\right] \delta_0. \tag{7}$$

Finally, there exists f for which (7) holds and is tight.

Equation (7) is the tight minimax rate for GD on a strongly-convex function (D. P. Bertsekas, 1997, Section 1.3); since it is also tight for GD with teleportation, Proposition 2.3 shows that a teleportation oracle cannot accelerate GD in the worst case. This is consistent with Zhao, Gower, et al. (2023), who require adaptive step-sizes from a Wolfe line-search (Wolfe, 1969; Wolfe, 1971) to show a linear/quadratic rate. Before giving conditions under which teleportation does strictly improve convergence, we show that the connection to Newton's method may fail for non-strongly convex f.

2.1 Convergence for Convex Functions

The link between GD with teleportation and Newton's method follows from optimality conditions for teleportation. Let w_k^+ be a local maximum of Equation (3). If the linear independence constraint qualification (LICQ) holds, then there exists $\lambda_k \geq 0$ such that w_k^+ satisfies the KKT conditions (D. P. Bertsekas, 1997),

$$\begin{aligned} \nabla^2 f(w_k^+)\nabla f(w_k^+) - \lambda_k \nabla f(w_k^+) = 0, \\ f(w_k^+) \leq f(w_k), \text{ and } \lambda_k(f(w_k^+) - f(w_k)) = 0. \end{aligned} \tag{8}$$

Equation (8) reveals that the dual parameter λ_k is also an eigenvalue of $\nabla^2 f(w_k^+)$. If $\lambda_k \neq 0$, then

$$\nabla f(w_k^+) = \lambda_k \left(\nabla^2 f(w_k^+)\right)^\dagger \nabla f(w_k^+), \tag{9}$$

where $A^\dagger$ denotes the pseudo-inverse of A. Thus, the first step of GD after teleporting is a Newton step with unknown scale λ_k. Since the teleportation problem satisfies LICQ unless $\nabla f(w_k^+) = 0$ (in which case f is stationary over $\mathcal{S}_k$), the connection to Newton may only fail when $\nabla^2 f$ is positive semi-definite and $\lambda_k = 0$. This is possible when f is non-strongly convex.

Proposition 2.4. *There exists an L-smooth, convex, and C^2 function for which the gradient direction after teleporting is not the Newton direction.*

Without the connection to Newton's method, we can only prove that GD with teleportation attains the same order of convergence as GD without teleportation.

Proposition 2.5. *If f is L-smooth and convex, then GD with $\eta < 2/L$ and tele-schedule $\mathcal{T}$ satisfies*

$$\delta_K \leq 2R^2 / \left(K\eta \left(2 - L\eta\right)\right).$$

Moreover, there exists f for which the convergence of GD with and without teleportation is identical.

In contrast, Bubeck (2015, Theorem 3.3) prove that standard GD converges at the rate,

$$\delta_k \leq 2\|w_0 - w^*\|_2^2 \,/\, \left(K\eta \left(2 - L\eta\right)\right).$$

Compared with standard results, GD with teleportation depends on the diameter of the initial sub-level set, R^2, rather than the initial distance $\|w_0 - w^*\|_2^2$. This dependence is necessary since one step of teleportation can nearly attain the diameter of the sub-level set.

Proposition 2.6. *There exists an L-smooth and convex function such that teleporting at $k = 0$ guarantees,*

$$\|w_0^+ - w^*\|_2 \geq R/4.$$

2.2 Convergence under Hessian Stability

So far we have shown that teleportation does not improve the convergence order of GD in the general convex setting *and* may blow-up the initial distance to w^*. However, in the special case where the Hessian is well-behaved, we show that teleportation leads to strictly faster optimization. We quantify the behavior of the Hessian using the notion of stability (Bach, 2010; Karimireddy et al., 2019; Gower et al., 2019).

Definition 2.7. We say f is $(\tilde{L}, \tilde{\mu})$ Hessian stable over $\mathcal{Q}$ if $\forall\, x, y \in \mathcal{Q},\ \nabla^2 f(x)(y - x) \neq 0$ and,

$$f(y) \leq f(x) + \langle \nabla f(x), y - x\rangle + \frac{\tilde{L}}{2}\|y - x\|_{\nabla^2 f(x)}^2, \tag{10}$$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x\rangle + \frac{\tilde{\mu}}{2}\|y - x\|_{\nabla^2 f(x)}^2. \tag{11}$$

Hessian stability holds for many problems, including logistic regression (Bach, 2010). Choosing $\mathcal{Q} = \mathcal{S}_0$ and using Lemma 2.1, shows that Hessian stability holds at every pair (w_{k+1}, w_k^+). As a result, $\nabla^2 f(w_k^+)\nabla f(w_k^+) \neq 0$ and Equation (9) implies the Newton and gradient directions are collinear. In particular, this means the first step of GD after teleporting obtains a linear rate.

Lemma 2.8. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then the first step of GD after teleporting with step-size $\eta < 2/L\tilde{L}$ makes linear progress as*

$$\delta_{k+1} \leq \left(1 + \tilde{\mu}\lambda_k\eta\left(\eta\lambda_k\tilde{L} - 2\right)\right)\delta_k. \tag{12}$$

Choosing the ideal step-size $\eta = 1/(\lambda_k\tilde{L})$ gives $\delta_{k+1} \leq (1 - \tilde{\mu}/\tilde{L})\delta_k$, but in general we cannot know λ_k. One way around this difficulty is to use the Armijo line-search (Armijo, 1966) after teleporting,

$$f(w_{k+1}) \leq f(w_{k+1}^+) - \frac{\eta}{2}\|\nabla f(w_k^+)\|_2^2. \tag{13}$$

Choosing η to be the largest step-size satisfying Equation (13) gives the following progress condition:

Lemma 2.9. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then the first step of GD with Armijo line-search after teleporting satisfies,*

$$\delta_{k+1} \leq \left(1 - \tilde{\mu}/\tilde{L}\right)\delta_k. \tag{14}$$

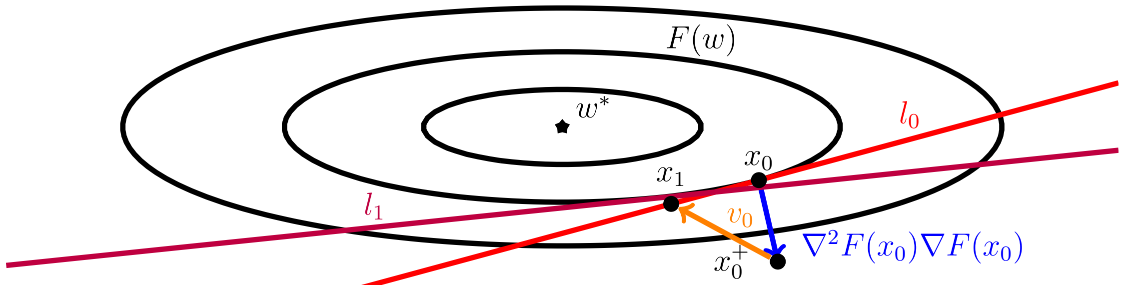

Figure 2: One iteration of our method for solving teleportation problems on a convex quadratic. The algorithm combines gradient ascent with projections onto the linearization $l_t = \{w : f(x_t) + \langle \nabla f(x_t), w - x_t\rangle = f(w_k)\}$.

Converting Lemmas 2.8 and 2.9 into a fast convergence rate requires merging the linear progress from steps after teleporting with the sub-linear progress from standard GD. Our proof technique unrolls the linear rate across adjacent iterations with teleportation and then bounds δ_k using the descent lemma to yield a combined convergence rate. In what follows, we assume the step-sizes are selected by line-search; see Theorem A.5 for an alternative result using a fixed step-size.

Theorem 2.10. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Consider any tele-schedule $\mathcal{T}$ and let $M = K - |\mathcal{T}|$ be the number of steps without teleportation. Then GD with step-size chosen by Armijo line-search converges as,*

$$\delta_K \leq \frac{2R^2 L}{M + 2R^2 L \sum_{k \in \mathcal{B}} \left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}} \right)^{b_k} - 1 \right] \frac{1}{\delta_{k-1}}}. \tag{15}$$

Assuming we teleport every-other iteration ($b_k = 1$) allows us to specialize Theorem 2.10 to obtain

$$\delta_K \leq \frac{2R^2 L}{K/2 + \frac{2R^2 L \tilde{\mu}}{\tilde{L} - \tilde{\mu}} \sum_{k \in \mathcal{T}} \frac{1}{\delta_{k-1}}}, \tag{16}$$

which shows teleporting at iteration k leads to a strictly faster rate than standard GD if $\delta_{k-1} \leq (2R^2 L \tilde{\mu})/(\tilde{L} - \tilde{\mu})$. That is, teleportation is effective when the optimality gap is small. This reflects the typical ordering between linear and sub-linear convergence, where the sub-linear rate is faster than the linear rate for a finite number of initial iterations (Bach, 2024). At the cost of hiding that relationship, weighted telescoping gives a fully explicit rate for GD with teleportation.

Theorem 2.11. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then GD with line-search and any tele-schedule $\mathcal{T}$ converges as,*

$$\delta_K \leq 2R^2 L \left[\sum_{k \notin \mathcal{T}} \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}} \right)^{n_k} \right]^{-1}, \tag{17}$$

where $n_k = |\{i \in \mathcal{T} : i > k\}|$ is the number teleportation steps after iteration k.

Compared to Theorem 2.10, this rate does not depend on the optimality gaps for progress and is an explicit convergence guarantee. As a price, the progress no longer decomposes into a sum over teleportation steps and a sum over regular GD steps. Despite this downside, it is still useful to interpret the rate in the special case of teleporting every-other iteration, for which we obtain,

$$\delta_K \leq 2R^2 L \frac{(\tilde{L} - \tilde{\mu})}{\tilde{\mu}} \left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}} \right)^{K/2} - 1 \right]^{-1}. \tag{18}$$

This surprising result shows that teleportation allows GD to obtain a linear rate without strong convexity and implies teleportation is particularly useful for computing high-accuracy solutions. However, we expect Equation (18) and Theorem 2.11 to be quite loose in general since they strongly down-weight steps without teleportation, effectively ignoring the progress made by standard GD steps. See Corollary A.4 for details and Theorem A.7 for fixed step-sizes.

Remark 2.12. Theorems 2.10 and 2.11 apply (with minor modifications to the proofs) to schemes which combine GD steps with intermittent standard Newton updates. As far as we know, these are the first analyses for alternating first-order/second-order methods which combine sub-linear and linear progress conditions to obtain a non-trivial convergence rate. In particular, using Newton steps at every iteration allows us to exactly match the rates for Newton's method (Karimireddy et al., 2019) or randomized subspace Newton (Gower et al., 2019) depending on the update scheme used.

3 EVALUATING THE TELEPORTATION OPERATOR

The results in the previous section assume access to a *teleportation oracle*, meaning a function which can solve the general non-linear programming problem needed for sub-level set teleportation. Although we could apply standard SQP methods, computing the Hessian of the teleportation objective requires third-order derivatives

Algorithm 2 Sub-level Set Teleportation

Inputs: Iterate w_k; Step-size ρ_0; Tolerances ϵ, δ.
$x_0 \leftarrow w_k$; $q_0 \leftarrow \nabla^2 f(x_0)\nabla f(x_0)$
while $\|\mathbf{P}_t q_t\|_2 > \epsilon$ or $f(x_t) - f(w_k) > \delta$ **do**
  $g_t \leftarrow \|\nabla f(x_t)\|_2^2$
  $v_t \leftarrow (\rho_t \langle \nabla f(x_t), q_t\rangle / g_t + f(x_t) - f(w_k))_+ \nabla f(x_t)$
  $x_{t+1} \leftarrow x_t + (\rho_t \cdot q_t - v_t)/g_t$
  while $\phi_{\gamma_t}(x_{t+1}) < \phi_{\gamma_t}(x_t) + \rho_t\|q_t\|_2^2/2g_t^2$ **do**
    $\rho_t \leftarrow \rho_t/2$
    $v_t \leftarrow (\rho_t \langle \nabla f(x_t), q_t\rangle / g_t + f(x_t) - f(w_k))_+ \nabla f(x_t)$
    $x_{t+1} \leftarrow x_t + (\rho_t \cdot q_t - v_t)/g_t$
  end while
  $q_{t+1} \leftarrow \nabla^2 f(x_t)\nabla f(x_t)$; $t \leftarrow t+1$
end while
Output: x_{t+1}

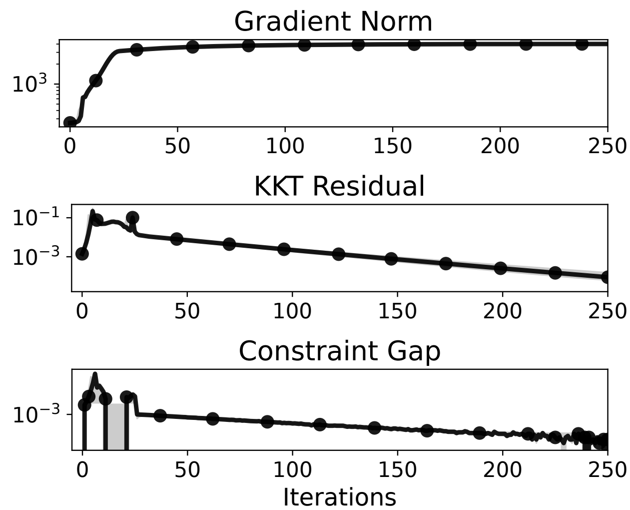

Figure 3: **Left**: full algorithm for sub-level set teleportation. **Right**: sub-level set teleportation for a two-layer MLP with 50 hidden units on MNIST. Our algorithm gives an approximate KKT point where the gradient norm is two orders of magnitude larger than that at the standard initialization.

of f, which is not feasible for large-scale problems. Instead, we develop an iterative, projected-gradient-type method that requires only Hessian-vector products. We denote by x_t the iterates of our method for solving teleportation problems, with $x_0 = w_k$.

For general f, the sub-level set $\mathcal{S}_k$ is non-convex with no closed-form projection. In contrast, the linearization of this constraint around x_t yields a half-space,

$$\tilde{\mathcal{S}}_k(x_t) := \{w : f(x_t) + \langle \nabla f(x_t), w - x_t\rangle \le f(w_k)\},$$

for which projections are simple. To obtain a tractable algorithm, we maximize a penalized linearization of the log-gradient-norm subject to this constraint,

$$x_{t+1} = \operatorname*{arg\,max}_{x \in \tilde{\mathcal{S}}_k(x_t)} \left\{ \frac{1}{2}\log(\|\nabla f(x_t)\|_2^2) - \frac{1}{2\rho_t}\|x - x_t\|_2^2 + \left\langle \frac{\nabla^2 f(x_t)\nabla f(x_t)}{\|\nabla f(x_t)\|_2^2}, x - x_t \right\rangle \right\}. \tag{19}$$

Taking the logarithm of the objective encodes positivity of the gradient norm and leads to a normalized update rule. Define $q_t = \nabla^2 f(x_t)\nabla f(x_t)$ and $g_t = \|\nabla f(x_t)\|_2^2$.

Proposition 3.1. *Equation* (19) *is solved by,*

$$\begin{aligned} v_t &= (\rho_t \langle \nabla f(x_t), q_t\rangle / g_t + f(x_t) - f(w_k))_+ \nabla f(x_t), \\ x_{t+1} &= x_t + (\rho_t q_t - v_t)/g_t. \end{aligned} \tag{20}$$

This iteration is equivalent to projected GD with a linearized sub-level set constraint (see Figure 2). It is also equivalent to an SQP method which uses $\mathbf{I}/\rho_t$ as an approximation to the Hessian of the teleportation objective (Torrisi et al., 2018). We leverage SQP to obtain the following guarantee.

Theorem 3.2. *Assume $\nabla f(w_k) \neq 0$, strict complementarity, and that Equation* (20) *is used with the relaxation step $\hat{x}_{k+1} = \alpha_t x_{t+1} + (1-\alpha_t)x_t$. Then, using appropriate (α_t, ρ_t), $\hat{x}_k$ converges asymptotically to a KKT point (x^*, λ^*) of Equation* (3)*; if (x^*, λ^*) is second-order critical, then local convergence is linear.*

Global linear convergence also holds under additional assumptions (Torrisi et al., 2018) — see Figure 3 for a real-world example. Although formally necessary, we have not found relaxation steps to be useful in practice. Finally, we remark that the requirement for strict complementarity is standard in non-linear programming (Carli Silva and Tuncel, 2019) and known to be satisfied for generic conic programs (Pataki and Tunçel, 2001).

3.1 Step-sizes and Termination

A disadvantage of our update is that it requires a step-size $\rho_t > 0$. Since teleportation is a sub-routine of a larger optimization procedure, tuning ρ_t for good performance is not acceptable. Following practice for SQP methods (see, e.g., Nocedal and Wright (1999, Theorem 18.2)), we select ρ_t using line-search on a *merit function* $\phi_\gamma(x) = \frac{1}{2}\log(\|\nabla f(x)\|_2^2) - \gamma\,(f(x) - f(w_k))_+$,

$$\phi_\gamma(x_{t+1}) \ge \phi_\gamma(x_t) + D_\phi(x_t, x_{t+1} - x_t)/2, \tag{21}$$

where $\gamma > 0$ is the penalty strength, and $D_\phi(x_t, u)$ is the directional derivative of ϕ_γ evaluated at x_t in direction u. Setting γ sufficiently large guarantees that Eq. (20) gives an ascent direction of ϕ_γ.

Proposition 3.3. *If $f(x_t) > f(w_k)$ and $\gamma_t > \langle q_t, v_t\rangle / \|\nabla f(x_t)\|_2^4 (f(x_t) - f(w_k))_+$, then $x_{t+1} - x_t$ is*

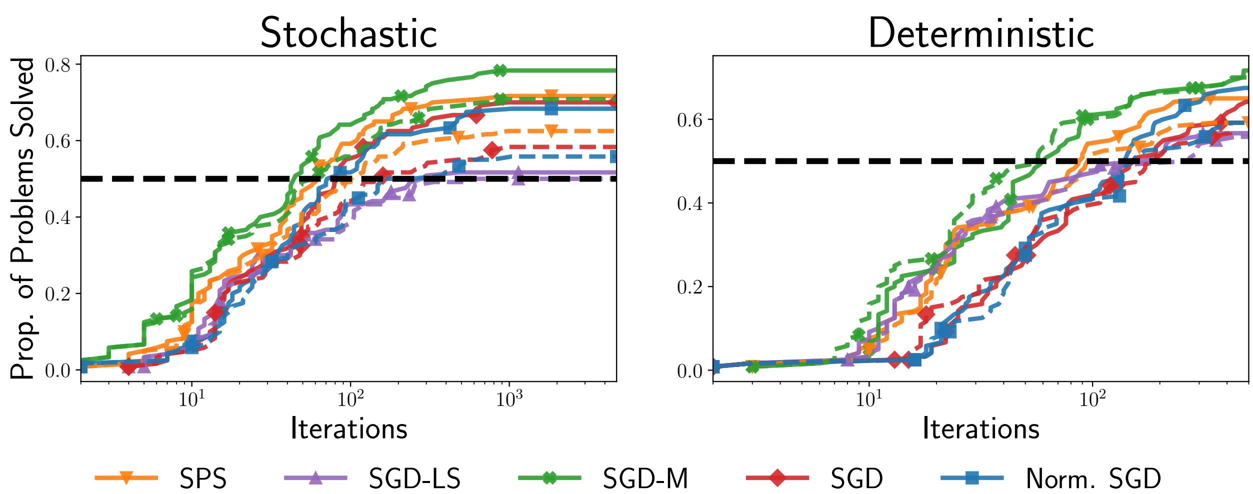

Figure 4: Performance profile comparing optimization methods with (solid lines) and without (dashed lines) sub-level set teleportation for training three-layer ReLU networks. Stochastic methods teleport once every 10 epochs starting from the epoch 5, while deterministic methods teleport once every 50 iterations starting from $k = 5$. A problem is "solved"' when $(f(w_k) - f(w^*)) / (f(w_0) - f(w^*)) \leq \tau$, where $f(w^*)$ is estimated separately and τ is a threshold. Performance is judged by comparing time to a fixed proportion of problems solved (see dashed line at 50%). Algorithms with intermittent teleportation uniformly dominate their standard counterparts.

an ascent direction of ϕ_{γ_t} and Eq. (21) *simplifies to,*

$$\phi_{\gamma_t}(x_{t+1}) \geq \phi_\gamma(x_t) + \rho_t \|q_t\|_2^2 / 2g_t^2. \tag{22}$$

Proposition 3.3 provides a recipe for computing step-sizes using backtracking line-search. Note that when $v_t = 0$ the update reduces to gradient ascent and $\gamma_t = 0$ is immediately sufficient for progress.

We also briefly discuss how to terminate our algorithm. Let $\mathbf{P}_t$ be the projection onto the orthogonal complement of $\nabla f(x_t)$. The KKT conditions (Equation (8)) imply x_t is near-optimal when $\|\mathbf{P}_t q_t\| < \epsilon$ and $f(x_t) - f(w_k) \leq \delta$. Algorithm 2 combines these conditions with line-search to give a complete solver.

3.2 Approximate Teleportation

Algorithm 2 is a practical and theoretically justified procedure for teleporting, but — like other SQP methods — it requires unbounded iterations to compute an exact solution to Equation (3). Instead, we typically obtain approximate KKT points satisfying,

$$\begin{aligned} \nabla^2 f(w_k^+) \nabla f(w_k^+) - \lambda_k \nabla f(w_k^+) &= e_k, \\ f(w_k^+) - f(w_k) &\leq r_k, \end{aligned} \tag{23}$$

where $\|e_k\|_2 \leq \epsilon_k$ is an error vector. Thus, as a corollary of Lemma 2.8, GD with approximate teleportation and step-size η makes at least as much progress as,

$$\delta_{k+1} \leq \left(1 + \tilde{\mu} \lambda_k \eta (\eta \lambda_k \tilde{L} - 2)\right) \delta_k + \frac{\tilde{L} \eta^2}{2} \|e_k^+\|_{\nabla^2 f_k}^2 + r_k.$$

While the e_k term can be managed using a decreasing step-size schedule, r_k can only be controlled by increasingly strict tolerances for the teleportation algorithm. If $r_k \to 0$ at an appropriate rate, then it is straightforward to show convergence of the overall GD scheme (D. P. Bertsekas and Tsitsiklis, 2000). Yet, this still does not lead to a combined gradient/Hessian complexity for GD with approximate teleportation because the convergence guarantee for Algorithm 2 is asymptotic. We leave complete iteration complexities to future work.

4 EXPERIMENTS

Now we present experiments on the performance of sub-level set teleportation in practice. We use Algorithm 2 to implement a teleportation oracle and evaluate the merits of teleportation for accelerating deterministic and stochastic optimization.

Solving the Teleportation Problem: Figure 1 shows the convergence path of our teleportation solver on two test functions: the Booth function and the Goldstein-Price function (Goldstein and Price, 1971). The Booth function is a convex quadratic, while Goldstein-Price is highly non-convex; our solver converges to the global optimum of the teleportation problem for the Booth function and a local maximum for the Goldstein-Price function. To demonstrate the scalability of our approach, we also solve sub-level set teleportation for a two-layer MLP with fifty hidden units and soft-plus activations on MNIST (LeCun et al., 1998). We use weight-decay regularization to ensure the objective is co-

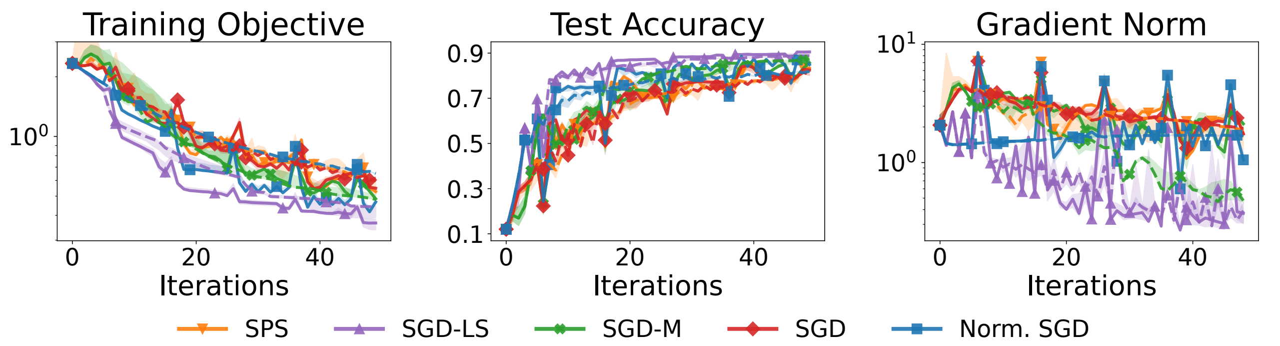

Figure 5: Performance of optimizers with (solid) and without (dashed) teleportation on MNIST. We train a MLP with the soft-plus activation and one hidden layer of size 500. All methods are run in batch mode. Teleportation significantly improves the convergence speed of all methods and does not affect generalization performance.

ercive. Figure 3 shows the norm of the network gradient (i.e. the teleportation objective), the KKT residual (8), and violation of the constraints during teleportation. Our algorithm converges to a KKT point where the gradient norm is two orders of magnitude larger than that at the initialization.

Performance Profile: Now we consider using teleportation to accelerate training of three-layer ReLU networks using gradient methods. We focus on iteration complexity since no previous paper has clearly demonstrated that teleportation improves convergence even using a cost-free teleportation oracle. We generate 120 problems by trying six regularization strengths on 20 binary classification datasets from the UCI repository (Asuncion and Newman, 2007). Figure 4 presents a performance profile (Dolan and Moré, 2002) comparing GD (SGD), GD with the Polyak step-size (SPS) (Polyak, 1987; Loizou et al., 2021), GD with line-search (GD-LS), and normalized GD (Norm. SGD) with (solid lines) and without (dashed lines) teleportation.

We find that access to a teleportation oracle almost uniformly improves iteration complexity of both stochastic and deterministic optimizers (nearly every solid line is above the corresponding dashed line). Teleportation seems particularly useful for computing high-accuracy solutions, as the gaps between methods with and without teleportation widen — meaning teleportation methods solve more problems to a high accuracy — when many iterations are used. This matches our theory, which suggests that teleportation is most useful when the optimality gap is small (see Theorem 2.10). We refer to Figure 8 for similar trends on convex problems satisfying Hessian stability and Figures 17 and 18 for convergence curves on specific problems.

Time Complexity: Now we compare the wall-clock time of gradient methods with and without teleportation. Since the accuracy of the teleportation solver affects both the time complexity of teleporting and the effectiveness of the teleportation steps, we leverage this experiment to perform an ablation on the maximum number of iterations used by Algorithm 2. Figures 11 and 12 show performance profiles on our 120 problems from the UCI repository with total time on the x-axis.

Although solving the teleportation problem to high accuracy introduces overhead, stochastic gradient methods using $t = 50$ iterations for the teleportation subprocedure solve significantly more problems than their standard counterparts with the same computational budget. This gap is particularly noticeable when training non-convex ReLU networks. In comparison, teleportation methods with insufficiently many iterations ($t \in \{1, 10\}$) are often slower than methods without teleportation without noticeable improvements in accuracy. We conclude that solving the teleportation problem accurately is critical and can be computationally worthwhile when highly accurate solutions are desired.

Image Classification: To confirm our observations, we train a two-layer network with 500 neurons and soft-plus activations on MNIST. We use full-batch gradients and compare teleporting once every five iterations, starting at $k = 5$, against the same methods without teleportation. Figure 5 shows that, as on the UCI datasets, teleporting leads to faster convergence for every optimization method considered. This gap is particularly pronounced for normalized GD and GD with line-search. Although teleportation steps significantly increase the gradient norm, it decreases quickly and generalization performance is unaffected. See Figures 13 and 14 for results with stochastic gradients and non-smooth activations.

Comparison with Newton: Equation (9) implies that GD with teleportation can be viewed as a type of Hessian-free Newton method. We study this connection and compare GD with teleportation at every step

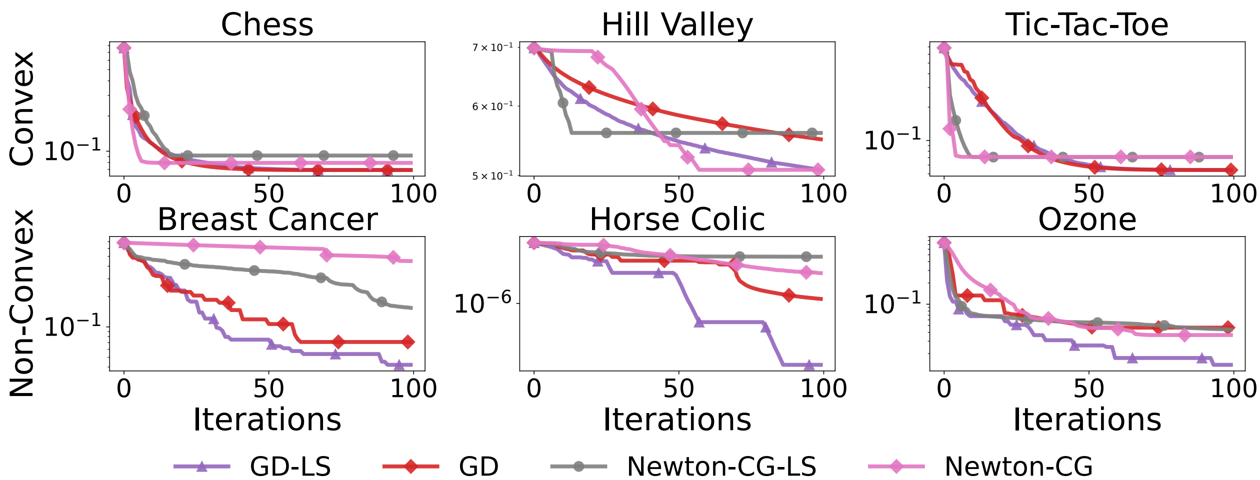

Figure 6: GD with teleportation compared to truncated Newton (Newton-CG), with and without line-search. Newton-CG is somewhat competitive on convex problems, but completely fails for non-convex optimization.

against Newton-CG (Dembo and Steihaug, 1983), an inexact Newton update where the search direction is computed using the conjugate gradient (CG) method. Because both methods use only Hessian-vector products, we can ensure an equal computational burden by fixing the number of CG iterations to be the same as the number of iterations used by Algorithm 2.

Figure 6 reports training performance on three convex logistic regression problems and three non-convex ReLU network training problems. We plot the minimum training objective attained so far at each iteration since GD with teleportation at every iteration produces noisy iterates. While Newton-CG is initially faster for 2/3 convex problems, it stalls due to poor quality of the approximate search direction. Moreover, Newton-CG fails on all three non-convex problems, likely because it is attracted to any critical point, including saddles and local maxima. This replicates well-known failure modes of Newton-type methods for non-convex optimization (Xu et al., 2020). In contrast, GD with teleportation still obtains a fast linear rate of convergence. See Figures 15 and 16 for figures with additional metrics.

Additional Experiments: In Appendix C.1 we present additional experiments on (i) using sub-level set teleportation to initialize optimization and (ii) the effects of regularization on the success of teleportation for non-coercive ReLU networks. Figures 9 and 10 show that initializing using teleportation has mixed performance compared to the standard Kaiming He initialization (He et al., 2015) and is particularly poor for logistic regression problems. We speculate that this is because R^2 may be very large at the start of optimization. Surprisingly, Figure 7 shows that there is no clear connection between regularization and improved success rates for methods with teleportation.

5 CONCLUSION

Despite work advocating for level set teleportation, little has been done to solve teleportation problems or evaluate their practical utility. We rectify this by studying teleportation in detail; we prove new theoretical guarantees for GD with teleportation, showing it can obtain a combined linear/sub-linear rate, which improves significantly over standard GD, when Hessian stability holds. We also derive a novel algorithm for solving teleportation problems and evaluate the performance of teleportation on a large suite of problems. Our results reveal that access to a teleportation oracle implemented by our algorithm speeds-up both convex and non-convex optimization can even outperform approximate Newton methods. Going beyond this oracle model to obtain full iteration complexities for GD with an approximate teleportation method is an open problem that we leave to future work.

Acknowledgements

Aaron Mishkin was supported by NSF Grant DGE-1656518, by NSERC Grant PGSD3-547242-2020, and by an internship at the Center for Computational Mathematics, Flatiron Institute. We thank Si Yi (Cathy) Meng and Michael Eickenberg for helpful conversations while preparing this work and David Martínez-Rubio and Gauthier Gidel for insightful comments on an earlier version. We also thank the Scientific Computing Core, Flatiron Institute, for supporting the numerical experiments in this work and Fabian Schaipp for use of the `step-back` code. We credit Ryan D'Orazio for drawing our attention to sufficient conditions for coercivity and the anonymous reviewers for their invaluable comments on earlier versions of this paper.

References

Armenta, Marco and Pierre-Marc Jodoin (2021). “The representation theory of neural networks”. In: *Mathematics* 9.24, p. 3216.

Armenta, Marco, Thierry Judge, et al. (2020). “Neural Teleportation”. In: *CoRR* abs/2012.01118.

Armijo, Larry (1966). “Minimization of functions having Lipschitz continuous first partial derivatives”. In: *Pacific Journal of mathematics* 16.1, pp. 1–3.

Asuncion, Arthur and David Newman (2007). *UCI machine learning repository.*

Bach, Francis (2010). “Self-concordant analysis for logistic regression”. In: *Electronic Journal of Statistics* 4, pp. 384–414.

– (2024). “Scaling laws of optimization”. In.

Bamler, Robert and Stephan Mandt (2018). “Improving optimization for models with continuous symmetry breaking”. In: *International Conference on Machine Learning*. PMLR, pp. 423–432.

Bengio, Yoshua et al. (2006). “Greedy Layer-Wise Training of Deep Networks”. In: *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, pp. 153–160.

Bertsekas, Dimitri (2009). *Convex optimization theory*. Vol. 1. Athena Scientific.

Bertsekas, Dimitri P (1997). “Nonlinear programming”. In: *Journal of the Operational Research Society* 48.3, pp. 334–334.

Bertsekas, Dimitri P and John N Tsitsiklis (2000). “Gradient convergence in gradient methods with errors”. In: *SIAM Journal on Optimization* 10.3, pp. 627–642.

Bubeck, Sébastien (2015). “Convex Optimization: Algorithms and Complexity”. In: *Found. Trends Mach. Learn.* 8.3-4, pp. 231–357.

Carli Silva, Marcel K de and Levent Tuncel (2019). “Strict complementarity in semidefinite optimization with elliptopes including the maxcut SDP”. In: *SIAM Journal on Optimization* 29.4, pp. 2650–2676.

Dembo, Ron S and Trond Steihaug (1983). “Truncated-Newton algorithms for large-scale unconstrained optimization”. In: *Mathematical Programming* 26.2, pp. 190–212.

Dinh, Laurent et al. (2017). “Sharp Minima Can Generalize For Deep Nets”. In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Vol. 70. Proceedings of Machine Learning Research, pp. 1019–1028.

Dolan, Elizabeth D and Jorge J Moré (2002). “Benchmarking optimization software with performance profiles”. In: *Mathematical programming* 91, pp. 201–213.

Evci, Utku et al. (2022). “GradMax: Growing Neural Networks using Gradient Information”. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=qjN4h_wwUO`.

Feinman, Reuben (2021). *Pytorch-minimize: a library for numerical optimization with autograd*. URL: `https://github.com/rfeinman/pytorch-minimize`.

Fernández-Delgado, Manuel et al. (2014). “Do we need hundreds of classifiers to solve real world classification problems?” In: *The journal of machine learning research* 15.1, pp. 3133–3181.

Foret, Pierre et al. (2021). “Sharpness-aware Minimization for Efficiently Improving Generalization”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Goldstein, Allen A. and J. F. Price (1971). “On descent from local minima”. In: *Mathematics of Computation* 25, pp. 569–574.

Gower, Robert M. et al. (2019). “RSN: Randomized Subspace Newton”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 614–623.

He, Kaiming et al. (2015). “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 1026–1034.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). “Flat Minima”. In: *Neural Comput.* 9.1, pp. 1–42.

Karimi, Hamed, Julie Nutini, and Mark Schmidt (2016). “Linear Convergence of Gradient and Proximal-Gradient Methods Under the Polyak-Łojasiewicz Condition”. In: *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I*. Vol. 9851. Lecture Notes in Computer Science, pp. 795–811.

Karimireddy, Sai Praneeth, Sebastian U. Stich, and Martin Jaggi (2019). “Global linear convergence of Newton’s method without strong-convexity or Lipschitz gradients”. In: *OPT 2019: Optimization for Machine Learning (NeurIPS 2019 Workshop)*.

Keskar, Nitish Shirish et al. (2017). “On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima”. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

LeCun, Yann et al. (1998). “Gradient-based learning applied to document recognition”. In: *Proc. IEEE* 86.11, pp. 2278–2324.

Lehtokangas, Mikko and Jukka Saarinen (1998). “Weight initialization with reference patterns”. In: *Neurocomputing* 20.1-3, pp. 265–278.

Loizou, Nicolas et al. (2021). “Stochastic Polyak Stepsize for SGD: An Adaptive Learning Rate for Fast Convergence”. In: *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*. Vol. 130. Proceedings of Machine Learning Research, pp. 1306–1314.

Min, Hancheng et al. (2021). “On the Explicit Role of Initialization on the Convergence and Implicit Bias of Overparametrized Linear Networks”. In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Vol. 139. Proceedings of Machine Learning Research, pp. 7760–7768.

Narkhede, Meenal V, Prashant P Bartakke, and Mukul S Sutaone (2022). “A review on weight initialization strategies for neural networks”. In: *Artificial intelligence review* 55.1, pp. 291–322.

Nesterov, Yurii E. (2004). *Introductory Lectures on Convex Optimization - A Basic Course*. Vol. 87. Applied Optimization. Springer.

Neyshabur, Behnam, Ruslan Salakhutdinov, and Nathan Srebro (2015). “Path-SGD: Path-Normalized Optimization in Deep Neural Networks”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2422–2430.

Nocedal, Jorge and Stephen J Wright (1999). *Numerical optimization*. Springer.

Paszke, Adam et al. (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 8024–8035.

Pataki, Gabor and Levent Tunçel (2001). “On the generic properties of convex optimization problems in conic form”. In: *Mathematical Programming* 89, pp. 449–457.

Polyak, Boris T (1987). “Introduction to optimization”. In.

Saxe, Andrew M., James L. McClelland, and Surya Ganguli (2014). “Exact solutions to the nonlinear dynamics of learning in deep linear neural networks”. In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Tarmoun, Salma et al. (2021). “Understanding the Dynamics of Gradient Flow in Overparameterized Linear models”. In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Vol. 139. Proceedings of Machine Learning Research, pp. 10153–10161.

Torrisi, Giampaolo et al. (2018). “A Projected Gradient and Constraint Linearization Method for Nonlinear Model Predictive Control”. In: *SIAM J. Control. Optim.* 56.3, pp. 1968–1999.

Vaswani, Sharan et al. (2019). “Painless stochastic gradient: Interpolation, line-search, and convergence rates”. In: *Advances in neural information processing systems* 32.

Wolfe, Philip (1969). “Convergence conditions for ascent methods”. In: *SIAM review* 11.2, pp. 226–235.

– (1971). “Convergence conditions for ascent methods. II: Some corrections”. In: *SIAM review* 13.2, pp. 185–188.

Xu, Peng, Fred Roosta, and Michael W Mahoney (2020). “Newton-type methods for non-convex optimization under inexact Hessian information”. In: *Mathematical Programming* 184.1, pp. 35–70.

Zhao, Bo, Nima Dehmamy, et al. (2023). *Symmetry Teleportation for Accelerated Optimization*. arXiv: `2205.10637 [cs.LG]`.

Zhao, Bo, Iordan Ganev, et al. (2023). “Symmetries, Flat Minima, and the Conserved Quantities of Gradient Flow”. In: *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*.

Zhao, Bo, Robert M. Gower, et al. (2023). “Improving Convergence and Generalization Using Parameter Symmetries”. In: *CoRR* abs/2305.13404.

Zhu, Zeyuan Allen and Lorenzo Orecchia (2017). “Linear Coupling: An Ultimate Unification of Gradient and Mirror Descent”. In: *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA*. Vol. 67. LIPIcs, 3:1–3:22.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. **Yes**
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. **Yes**
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. **No**

   Justification: Source code will be released after acceptance. All algorithms are carefully described and illustrated with pseudocode. We prove iteration complexities for GD with teleportation and state when no iteration complexity is known (i.e. convergence is only known asymptotically).

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. **Yes**
   (b) Complete proofs of all theoretical results. **Yes**
   (c) Clear explanations of any assumptions. **Yes**

   Justification: Every theoretical result is presented in a formal statement with the complete set of necessary assumptions. All proofs are provided in the appendix.

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). **No**
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). **Yes**
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). **Yes**
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). **Yes**

   Justification: Appendix C.2 contains all hyperparameter settings and algorithmic details necessary to reproduce our experiments. Algorithms are exactly described using pseudo-code. Moreover, we will release the code to exactly reproduce our experiments upon publication.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. **Yes**
   (b) The license information of the assets, if applicable. **Yes**
   (c) New assets either in the supplemental material or as a URL, if applicable. **Not Applicable**
   (d) Information about consent from data providers/curators. **Not Applicable**
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. **Not Applicable**

   Justification: We do not create or release new assets. All existing datasets and models have been carefully referenced in the text.

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. **Not Applicable**
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. **Not Applicable**
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. **Not Applicable**

   Justification: We did not use crowdsourcing.

A LEVEL SET TELEPORTATION: PROOFS

Definition A.1. An operator $M : \mathbb{R}^d \to \mathbb{R}^d$ is called non-expansive if for every $x, y \in \mathbb{R}^d$,

$$\|M(x) - M(y)\|_2 \leq \|x - y\|_2.$$

Lemma A.2. *Suppose f is (strictly) convex. Then at least one (every) solution to the sub-level set teleportation problem (Equation (3)) is a solution to the level set teleportation problem (Equation (2)).*

Proof. Let $w(t) = w + t\nabla f(w)$. From convexity,

$$\begin{aligned} f(w(t)) - f(w) &\geq t\|\nabla f(w)\|_2^2 \\ f(w) - f(w(t)) &\geq -t \left\langle \nabla f(w(t)), \nabla f(w) \right\rangle. \end{aligned}$$

Adding these inequalities and using Cauchy-Schwarz,

$$\implies \|\nabla f(w(t))\|_2 \geq \|\nabla f(w)\|_2.$$

That is, the gradient norm is monotone non-decreasing along the gradient direction when f is convex. Thus, at least one solution to the maximization problem must occur on the boundary of the sub-level set, which completes the first part of the proof. For the second, simply note that the inequalities hold strictly if f is strictly convex, implying that every solution must be on the boundary. □

Lemma 2.1. *If f is L-smooth and $\eta_k < 2/L$, then GD with tele-schedule $\mathcal{T}$ satisfies $\delta_{k+1} \leq \delta_k$, $w_k \in \mathcal{S}_0$, and $\|w_k - w^*\|_2 \leq R$ for every $k \in \mathbb{N}$.*

Proof. Starting from the descent lemma (Equation (1)), we obtain

$$\begin{aligned} f(w_{k+1}) &\leq f(w_k^+) + \eta_k \left(\frac{L\eta_k}{2} - 1\right) \|\nabla f(w_k^+)\|_2^2 \\ &\leq f(w_k) + \eta_k \left(\frac{L\eta_k}{2} - 1\right) \|\nabla f(w_k)\|_2^2 \\ &\leq f(w_k), \end{aligned}$$

since $\eta_k < 2/L$. Subtracting $f(w^*)$ from both sides implies $\delta_{k+1} \leq \delta_k$. Thus, $w_{k+1} \in \mathcal{S}_0$ and $\|w_{k+1} - w^*\|_2 \leq R$ by definition. Arguing by induction now completes the proof. □

Lemma 2.2. *Suppose f is L-smooth and μ-SC. If $k \in \mathcal{T}$, then GD with step-size η_k satisfies*

$$\Delta_{k+1} \leq \max\left\{(1-\eta_k L)^2, (1-\eta_k\mu)^2\right\} \|w_k^+ - w^*\|_2^2. \tag{5}$$

Proof. Since f is L-smooth and μ strongly convex, ∇f satisfies the following inequality:

$$\langle \nabla f(x) - \nabla f(y), x - y\rangle \geq \frac{\mu L}{\mu + L}\|x - y\|_2^2 + \frac{1}{L + \mu}\|\nabla f(x) - \nabla f(y)\|_2^2. \tag{24}$$

This is sometimes called coercivity of the gradient; see, for example, Nesterov (2004, Theorem 2.1.12). Suppose

$k \in \mathcal{T}$. Then,

$$\begin{aligned}
\|w_{k+1} - w^*\|_2^2 &= \|w_k^+ - \eta_k \nabla f(w_k^+) - w^*\|_2^2 \\
&= \|w_k^+ - w^*\|_2^2 - 2\eta_k \left\langle \nabla f(w_k^+), w_k^+ - w^* \right\rangle + \eta_k^2 \|\nabla f(w_k^+)\|_2^2 \\
&\leq \|w_k^+ - w^*\|_2^2 - 2\eta_k \left(\frac{\mu L}{\mu + L} \|w_k^+ - w^*\|_2^2 + \frac{1}{L+\mu} \|\nabla f(w_k^+)\|_2^2 \right) \\
&\quad + \eta_k^2 \|\nabla f(w_k^+)\|_2^2 \\
&= \left(1 - \frac{2\eta_k \mu L}{\mu + L}\right) \|w_k^+ - w^*\|_2^2 + \eta_k \left(\eta_k - \frac{2}{\mu + L}\right) \|\nabla f(w_k^+)\|_2^2 \\
&\leq \left(1 - \frac{2\eta_k \mu L}{\mu + L}\right) \|w_k^+ - w^*\|_2^2 \\
&\quad + \eta_k \max \left\{ L^2 \left(\eta_k - \frac{2}{\mu + L}\right), \mu^2 \left(\eta_k - \frac{2}{\mu + L}\right) \right\} \|w_k^+ - w^*\|_2^2 \\
&= \max \left\{ (1 - \eta_k L)^2, (1 - \eta_k \mu)^2 \right\} \|w_k^+ - w^*\|_2^2,
\end{aligned}$$

where we have used μ strong convexity and L-smoothness to bound the gradient norm depending on the step-size η_k. This completes the proof. □

Proposition 2.3. *If f is L-smooth and μ-SC, then GD with $\eta_k < 2/L$ and tele-schedule $\mathcal{T}$ satisfies,*

$$\delta_K \leq \prod_{i=0}^{K-1} \left[1 - 2\mu\eta_i \left(1 - \frac{\eta_i L}{2}\right) \right] \delta_0. \tag{6}$$

Moreover, if the teleportation operator is non-expansive, then for any step-sizes η_k we have,

$$\delta_K \leq \frac{L}{\mu} \prod_{i=0}^{K-1} \left[\max \left\{ (1 - \eta_i L)^2, (1 - \eta_i \mu)^2 \right\} \right] \delta_0. \tag{7}$$

Finally, there exists f for which (7) holds and is tight.

Proof. First we show the upper bound when teleportation is not non-expansive. Starting from the descent lemma Equation (1), we have

$$\begin{aligned}
f(w_{k+1}) - f(w^*) &\leq f(w_k^+) - f(w^*) - \eta_k \left(1 - \frac{\eta_k L}{2}\right) \|\nabla f(w_k^+)\|_2^2 \\
&\leq f(w_k) - f(w^*) - \eta_k \left(1 - \frac{\eta_k L}{2}\right) \|\nabla f(w_k)\|_2^2 \\
&\leq f(w_k) - f(w^*) - 2\mu\eta_k \left(1 - \frac{\eta_k L}{2}\right) (f(w_k) - f(w^*)) \\
&= \left(1 - 2\mu\eta_k \left(1 - \frac{\eta_k L}{2}\right)\right) (f(w_k) - f(w^*)),
\end{aligned}$$

where in the last inequality we used $\eta_k < \frac{2}{L}$ and the PL-condition,

$$\frac{1}{2\mu} \|\nabla f(w)\|_2^2 \geq f(w) - f(w^*),$$

which is implied by strong convexity. Recursing on the final inequality implies

$$\delta_K \leq \prod_{i=0}^{K-1} \left(1 - 2\mu\eta_i \left(1 - \frac{\eta_i L}{2}\right)\right) \delta_0,$$

as claimed.

Now we consider the setting where teleportation is assumed to be non-expansive. Starting from Lemma 2.2, we obtain,

$$\begin{aligned}\|w_{k+1} - w^*\|_2^2 &\le \max\left\{(1-\eta_k L)^2, (1-\eta_k \mu)^2\right\} \|w_k^+ - w^*\|_2^2 \\ &\le \max\left\{(1-\eta_k L)^2, (1-\eta_k \mu)^2\right\} \|w_k - w^*\|_2^2,\end{aligned}$$

by non-expansivity of teleportation and the fact that w^* is a fixed point of the teleportation operator. Recursing on this inequality implies,

$$\begin{aligned}\|w_K - w^*\|_2^2 &\le \prod_{i=0}^{K-1} \max\left\{(1-\eta_i L)^2, (1-\eta_i \mu)^2\right\} \|w_0 - w^*\|_2^2 \\ \implies f(w_K) - f(w^*) &\le \frac{L}{\mu}\left[\prod_{i=0}^{K-1} \max\left\{(1-\eta_i L)^2, (1-\eta_i \mu)^2\right\}\right] \left(f(w_0) - f(w^*)\right),\end{aligned}$$

where the last inequality uses both smoothness and strong convexity to upper and lower bound $\|w - w^*\|_2^2$ in terms of $f(w) - f(w^*)$.

For the lower bound, suppose $f(w) = \frac{1}{2}\|w\|_2^2$. Clearly f is 1-smooth and strongly convex with $\mu = 1$. The unique minimizer is simply $w^* = 0$. Starting from an arbitrary w_0, each iteration of GD with teleportation has the following recursion:

$$w_{k+1} = w_k^+ - \eta_k w_k^+ = (1 - \eta_k) w_k^+,$$

and the objective evolves trivially as

$$\begin{aligned}\frac{1}{2}\|w_{k+1}\|_2^2 &= (1-\eta_k)^2 \frac{1}{2}\|w_k^+\|_2^2 \\ &= (1-\eta_k)^2 \frac{1}{2}\|w_k\|_2^2,\end{aligned}$$

where the second equality uses Lemma A.2 to guarantee that the solution to sub-level set teleportation lies on the level set $\mathcal{L}_k$ and the fact that every point $x \in \mathcal{L}_k$ satisfies $\|x\|_2 = \|w_k\|$. Thus, each step of gradient descent makes progress exactly matching the convergence rate for general smooth, strongly convex functions regardless of teleportation. This completes the lower-bound. □

A.1 Convergence for Convex Functions: Proofs

Proposition 2.4. *There exists an L-smooth, convex, and C^2 function for which the gradient direction after teleporting is not the Newton direction.*

Proof. Define the second-order Huber function,

$$h_2(x) = \begin{cases} -\frac{1}{8}x^4 + \frac{3}{4}x^2 + \frac{3}{8} & \text{if } |x| < 1, \\ |x| & \text{Otherwise.} \end{cases} \tag{25}$$

It is straightforward to verify that h_2 is C^2 and convex with gradient function

$$\nabla h_2(x) = \begin{cases} -\frac{1}{2}x^3 + \frac{3}{2}x & \text{if } |x| < 1, \\ \text{sign}(x) & \text{otherwise.} \end{cases}$$

Consider the family of target functions $f_d : \mathbb{R}^d \to \mathbb{R}$ given by

$$f_d(x) = \sum_{i=1}^{d} h_2(x_i),$$

where d is an integer. Given level $b \ge 1$, define $q = \lfloor b \rfloor$ and consider the teleportation problem,

$$x^+ \in \arg\max_x \left\{\|\nabla f_q(x)\|_2^2 : f_q(x) = b\right\}.$$

Since f_q is symmetric, we restrict ourselves to maximization over the non-negative orthant. At any point x, the squared-norm of the gradient is given by

$$\|\nabla f_q(x)\|_2^2 = \sum_{i=1}^{q} (\nabla h_2(x_i))^2,$$

which is maximized when $x_i \geq 1$ for each i. Since $q < b$, the solution set of the teleportation problem is the following:

$$\mathcal{X}^* = \left\{ x : x_i \geq 1, \sum_{i=1}^{q} x_i = b \right\}.$$

At any of these points, the gradient is simply $\nabla f_n(x) = \mathbf{1}$. However, the Hessian is $\nabla^2 f(x) = 0$, meaning the gradient does not coincide with the Newton direction, which is not defined.

Although this argument requires $b \geq 1$, in practice any $b > 0$ can be handled by adjusting the point at which h_2 transitions from a linear to a quartic function. □

Proposition 2.5. *If f is L-smooth and convex, then GD with $\eta < 2/L$ and tele-schedule $\mathcal{T}$ satisfies*

$$\delta_K \leq 2R^2 / \left(K\eta \left(2 - L\eta\right)\right).$$

Moreover, there exists f for which the convergence of GD with and without teleportation is identical.

Proof. The proof follows Zhu and Orecchia (2017, Fact B.1) and Bubeck (2015, Theorem 3.3). Starting from L-smoothness of f, we obtain

$$\begin{aligned} f(w_{k+1}) &\leq f(w_k^+) + \eta \left(\frac{L\eta}{2} - 1 \right) \|\nabla f(w_k^+)\|_2^2 \\ &\leq f(w_k) + \eta \left(\frac{L\eta}{2} - 1 \right) \|\nabla f(w_k)\|_2^2, \end{aligned} \tag{26}$$

which implies that $\{f(w_k)\}$ are monotone decreasing regardless of the teleportation schedule. Thus, $\|w_{k+1} - w^*\|_2 \leq R$ for all k and we use convexity of f and the Cauchy-Schwarz inequality to obtain,

$$\begin{aligned} f(w_k) - f(w^*) &\leq \langle \nabla f(w_k), w_k - w^* \rangle \\ &\leq R \|\nabla f(w_k)\|_2, \end{aligned}$$

which holds for all k. Let $\delta_k = f(w_k) - f(w^*)$. Using Equation (26) from above and then dividing on both sides by δ_k and δ_{k+1} gives the following:

$$\begin{aligned} \delta_k^2 &\leq \frac{2R^2}{\eta \left(2 - L\eta\right)} \left(\delta_k - \delta_{k+1}\right) \\ \implies 1 \leq \frac{\delta_k}{\delta_{k+1}} &\leq \frac{2R^2}{\eta \left(2 - L\eta\right)} \left(\frac{1}{\delta_{k+1}} - \frac{1}{\delta_k} \right) \\ \implies \frac{\eta \left(2 - L\eta\right)}{2R^2} &\leq \left(\frac{1}{\delta_{k+1}} - \frac{1}{\delta_k} \right), \end{aligned}$$

so that summing from $k = 0$ to $K - 1$ gives

$$\begin{aligned} \frac{K\eta \left(2 - L\eta\right)}{2R^2} &\leq \left(\frac{1}{\delta_K} - \frac{1}{\delta_0} \right) \\ &\leq \frac{1}{\delta_K}, \end{aligned}$$

which completes the first part of the proof.

For the second part of the proof, consider the second-order Huber function,

$$h_2(x) = \begin{cases} -\frac{1}{8}x^4 + \frac{3}{4}x^2 + \frac{3}{8} & \text{if } |x| < 1, \\ |x| & \text{Otherwise.} \end{cases} \tag{27}$$

It is straightforward to verify that h_2 is C^2 and convex. Let $f(w) = h_2(w_1)$, meaning f is constant in every dimension but the first. This implies that the teleportation operator

$$w_k^+ = \arg\max\left\{\frac{1}{2}\|\nabla f(w)\|_2^2 : f(w) \leq f(w_k)\right\},$$

is simply the identity operator, i.e. $w_k^+ = w_k$. Thus, the optimization paths of vanilla GD and GD with teleportation are identical and the two methods have the same convergence rate. □

Proposition 2.6. *There exists an L-smooth and convex function such that teleporting at $k = 0$ guarantees,*

$$\|w_0^+ - w^*\|_2 \geq R/4.$$

Proof. We assume for convenience that $w = (x, y)$, $x > 1$ and $0 < y < \alpha$. These assumptions can be relaxed by modifying our construction. Let $\epsilon > 0$ such that $\alpha, y > \epsilon$ and $x + y\epsilon > \alpha\epsilon + 1$ hold, where both conditions can be satisfied by taking ϵ sufficiently small.

Let g_δ be the Huber function defined by

$$g_\delta(x) = \begin{cases} \frac{1}{2}x^2 & \text{if } x \leq \delta \\ \delta\left(|x| - \delta/2\right) & \text{otherwise.} \end{cases} \tag{28}$$

and consider the objective function

$$f_{(\epsilon,\alpha)}(x, y) = g_1(x) + g_\epsilon(y) + \frac{1}{2}\mathbb{1}_{y \geq \alpha}\left(y - \alpha\right)^2. \tag{29}$$

We first show that there exists $x' = 1$ and some $y' > \alpha$ satisfying

$$f_{(\epsilon,\alpha)}(x', y') = f_{(\epsilon,\alpha)}(x, y).$$

For this to hold, we must have

$$\begin{aligned} & f_{(\epsilon,\alpha)}(x', y') = \frac{1}{2} + \epsilon y' - \frac{\epsilon^2}{2} + \frac{1}{2}\left(y' - \alpha\right)^2 = x - \frac{1}{2} + \epsilon y - \frac{\epsilon^2}{2} = f_{(\epsilon,\alpha)}(x, y) \\ \iff & (y')^2 + 2\left(\epsilon - \alpha\right)y' + \left(\alpha^2 - 2x - 2\epsilon y + 2\right) = 0 \\ \iff & y' = \alpha - \epsilon \pm \sqrt{\left(\epsilon - \alpha\right)^2 - \alpha^2 + 2x + 2\epsilon y - 2}. \end{aligned}$$

In particular, for $y' > \alpha$, it must hold that

$$\begin{aligned} & \left(\epsilon - \alpha\right)^2 - \alpha^2 + 2x + 2\epsilon y - 2 > \epsilon^2 \\ \iff & x + \epsilon y > \alpha\epsilon + 1, \end{aligned}$$

where this last condition is guaranteed by assumption on ϵ. We conclude that (x', y') is on the level set as desired.

The gradient of $f_{(\epsilon,\alpha)}$ is easy to calculate as

$$\nabla f_{(\epsilon,\alpha)}(x, y) = \begin{cases} \operatorname{sign}(x) + \epsilon \cdot \operatorname{sign}(y) + \mathbb{1}_{y \geq \alpha}\left(y - \alpha\right) & \text{if } |x| \geq 1, |y| \geq \epsilon \\ x + \epsilon \cdot \operatorname{sign}(y) + \mathbb{1}_{y \geq \alpha}\left(y - \alpha\right) & \text{if } |x| \leq 1, |y| \geq \epsilon \\ \operatorname{sign}(x) + \epsilon y + \mathbb{1}_{y \geq \alpha}\left(y - \alpha\right) & \text{if } |x| \geq 1, |y| \leq \epsilon \\ x + \epsilon y + \mathbb{1}_{y \geq \alpha}\left(y - \alpha\right) & \text{if } |x| \leq 1, |y| \leq \epsilon. \end{cases}$$

In particular, the gradient norm at (x', y') is given by

$$\|\nabla f_{(\epsilon,\alpha)}(x', y')\|_2^2 = 1 + \epsilon^2 + (y' - \alpha)^2.$$

A straightforward case analysis reveals that for every $\bar{x}$, $\bar{y}$ such that $\bar{y} < \alpha$,

$$\|\nabla f_{(\epsilon,\alpha)}(\bar{x}, \bar{y})\|_2^2 < \|\nabla f_{(\epsilon,\alpha)}(x', y')\|_2^2.$$

That is, the maximizer of the gradient norm on the level set must satisfy $y^+ \geq \alpha$. We conclude that

$$\|w^+ - w^*\|_2^2 = \|(x^+, y^+)\|_2^2 \geq \alpha^2.$$

Now we upper-bound the diameter of the sub-level set as

$$\begin{aligned} R^2 &= \max\left\{\|(x', y')\|_2 : f_{(\epsilon,\alpha)}((x', y')) = f_{(\epsilon,\alpha)}((x, y))\right\} \\ &\leq \max\left\{|x'| + |y'| : \exists \tilde{x}, \tilde{y} \text{ s.t. } f_{(\epsilon,\alpha)}((x', \tilde{y})) = f_{(\epsilon,\alpha)}((x, y)), f_{(\epsilon,\alpha)}((\tilde{x}, y')) = f_{(\epsilon,\alpha)}((x, y))\right\}. \end{aligned}$$

Taking $\tilde{x} = \tilde{y} = 0$ allow us to maximize x', y'. As a result, the constraint on x' implies

$$x' - \frac{1}{2} = x - \frac{1}{2} + \epsilon(y - \epsilon/2),$$

from which we deduce

$$x' = x + \epsilon y - \epsilon^2/2 \leq x + \epsilon y.$$

Repeating a similar calculation for y', we obtain the constraint

$$\begin{aligned} \epsilon(y' - \epsilon/2) + \frac{1}{2}(y' - \alpha)^2 &= x + \epsilon(y - \epsilon/2) \\ \implies y' &= \alpha - \epsilon \pm \sqrt{\epsilon^2 - 2\alpha\epsilon + 2x + 2\epsilon y} \\ \implies (y' - \alpha + \epsilon)^2 &= \epsilon^2 - 2\alpha\epsilon + 2x + 2\epsilon y \\ \implies (y')^2 &= 2\alpha y' - 2(y' - y)\epsilon + 2x - \alpha^2 \\ \implies y' &\leq 2\alpha + \frac{2x}{\alpha}, \end{aligned}$$

where we have used $y' \geq \alpha > y$. Combining these results, we have

$$R \leq x + \epsilon y + 2\alpha + \frac{2x}{\alpha}.$$

Taking $\alpha \gg x$, x arbitrarily close to 1, and ϵ sufficiently small implies there exists an absolute constant $C < 2$ such that

$$R \leq 2\alpha + C < 4\alpha.$$

Recalling that $\|(x^+, y^+)\|_2 \geq \alpha$ shows that teleportation comes within a small constant factor of the diameter of the sub-level set. □

A.2 Convergence under Hessian Stability: Proofs

Lemma A.3. *If f satisfies $(\tilde{L}, \tilde{\mu})$-Hessian stability, then minimizing on both sides of Equation (11) implies*

$$f(x^*) \geq f(x) - \frac{c}{2}\|\nabla f(x)\|_{\nabla^2 f(x)^{-1}}. \tag{30}$$

Proof. See Karimireddy et al. (2019, Lemma 2) for proof. □

Lemma 2.8. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then the first step of GD after teleporting with step-size $\eta < 2/L\tilde{L}$ makes linear progress as*

$$\delta_{k+1} \leq \left(1 + \tilde{\mu}\lambda_k\eta\left(\eta\lambda_k\tilde{L} - 2\right)\right)\delta_k. \tag{12}$$

Proof. Lemma 2.1 implies that Hessian stability holds since the iterates remain inside the initial sub-level set. Starting from Equation (10), we obtain

$$
\begin{aligned}
f(w_{k+1}) &\leq f(w_k^+) + \left\langle \nabla f(w_k^+), w_{k+1} - w_k^+ \right\rangle + \frac{\tilde{L}}{2} \|w_{k+1} - w_k^+\|^2_{\nabla^2 f(w_k^+)} \\
&\leq f(w_k) - \lambda_k \eta \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} + \frac{\tilde{L}\lambda_k^2 \eta^2}{2} \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} \\
&= f(w_k) + \lambda_k \eta \left(\frac{\tilde{L}\lambda_k \eta}{2} - 1 \right) \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} \\
&\leq f(w_k) + 2\lambda_k \eta \tilde{\mu} \left(\frac{\tilde{L}\lambda_k \eta}{2} - 1 \right) \left[f(w_k^+) - f(w^*) \right] \\
&= f(w_k) + 2\lambda_k \eta \tilde{\mu} \left(\frac{\tilde{L}\lambda_k \eta}{2} - 1 \right) \left[f(w_k) - f(w^*) \right],
\end{aligned}
$$

where we have used $\lambda_k \leq L$ to guarantee $\frac{\tilde{L}\lambda_k\eta}{2} < 1$ and then applied Equation (30). The final equality follows since Hessian stability implies f is strictly convex, at which point Lemma A.2 guarantees $f(w_k^+) = f(w_k)$. Adding and subtracting $f(w^*)$ on both sides yields the final result,

$$
f(w_{k+1}) - f(w^*) \leq \left(1 + 2\lambda_k \eta \tilde{\mu} \left(\frac{\tilde{L}\lambda_k \eta}{2} - 1 \right) \right) \left(f(w_k) - f(w^*) \right).
$$

□

Lemma 2.9. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then the first step of GD with Armijo line-search after teleporting satisfies,*

$$
\delta_{k+1} \leq \left(1 - \tilde{\mu}/\tilde{L} \right) \delta_k. \tag{14}
$$

Proof. Satisfaction of the line-search criterion in Equation (13) implies $f(w_{k+1}) \leq f(w_k^+) = f(w_k)$, meaning the first GD step after teleporting can only decrease the function value and thus $w_{k+1} \in \mathcal{S}_0$. So, w_k, w_k^+, and w_{k+1} satisfy stability of the Hessian.

Now we use stability to control the step-size η. Using $\nabla f(w_k^+) = \lambda_k \nabla^2 f(w_k^+)^{-1} \nabla f(w_k^+)$ from teleportation optimality conditions, the line-search criterion is equivalent to,

$$
f(w_{k+1}) \leq f(w_k^+) - \frac{\lambda_k \eta}{2} \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}}.
$$

At the same time, Hessian stability implies

$$
\begin{aligned}
f(w_{k+1}) &\leq f(w_k^+) + \left\langle \nabla f(w_k^+), w_{k+1} - w_k^+ \right\rangle + \frac{\tilde{L}}{2} \|w_{k+1} - w_k^+\|^2_{\nabla^2 f(w_k^+)} \\
&\leq f(w_k) - \lambda_k \eta \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} + \frac{\tilde{L}\lambda_k^2 \eta^2}{2} \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} \\
&= f(w_k) + \lambda_k \eta \left(\frac{\tilde{L}\lambda_k \eta}{2} - 1 \right) \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}}.
\end{aligned}
$$

Since η is the largest step-size for which the line-search criterion holds, we can deduce the following bound on the step-size:

$$
\begin{aligned}
& \frac{\lambda_k \eta}{2} \geq \lambda_k \eta \left(1 - \frac{\tilde{L}\lambda_k \eta}{2} \right) \\
\implies & \eta \geq \frac{1}{\tilde{L}\lambda_k}.
\end{aligned}
$$

Substituting this equation back into the line-search criterion and using the lower bound from Hessian stability (Equation (30)) yields

$$
\begin{aligned}
f(w_{k+1}) &\leq f(w_k^+) - \frac{1}{2\tilde{L}} \|\nabla f(w_k^+)\|^2_{\nabla^2 f(w_k^+)^{-1}} \\
&\leq f(w_k^+) - \frac{\tilde{\mu}}{\tilde{L}} \left(f(w_k^+) - f(w^*)\right) \\
&= f(w_k) - \frac{\tilde{\mu}}{\tilde{L}} \left(f(w_k) - f(w^*)\right),
\end{aligned}
$$

where we used the fact that Hessian stability implies f is strictly convex, at which point Lemma A.2 guarantees $f(w_k^+) = f(w_k)$. Adding and subtracting $f(w^*)$ on both sides finishes the proof,

$$
f(w_{k+1}) - f(w^*) \leq \left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right) \left(f(w_k) - f(w^*)\right).
$$

□

Theorem 2.10. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Consider any tele-schedule $\mathcal{T}$ and let $M = K - |\mathcal{T}|$ be the number of steps without teleportation. Then GD with step-size chosen by Armijo line-search converges as,*

$$
\delta_K \leq \frac{2R^2 L}{M + 2R^2 L \sum_{k \in \mathcal{B}} \left[\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k} - 1\right] \frac{1}{\delta_{k-1}}}. \tag{15}
$$

Proof. Satisfaction of the line-search criterion in Equation (13) implies $f(w_{k+1}) \leq f(w_k)$ regardless of teleportation. As a result, $\{f(w_k)\}$ are non-increasing and Hessian stability holds for all iterates.

Assume $k \in \mathcal{B}$. Using Lemma 2.9 and unrolling Equation (14) for the b_k teleportation steps, we obtain

$$
\begin{aligned}
\delta_{k+b_k} &\leq \left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right)^{b_k} \delta_k \\
&\leq \left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right)^{b_k} \left[\delta_{k-1} - \frac{\eta_{k-1}}{2} \|\nabla f(w_{k-1})\|_2^2\right],
\end{aligned}
$$

where the second inequality follows from the line-search. Since η is the largest step-size for which Equation (13) holds and f is L-smooth, the descent lemma (Equation (1)) yields

$$
\frac{\eta_{k-1}}{2} \geq \eta_{k-1} \left(1 - \frac{\eta_{k-1} L}{2}\right) \implies \eta_{k-1} \geq \frac{1}{L}.
$$

Combining control of η_{k-1} with our bound on δ_{k+b_k} yields,

$$
\delta_{k+b_k} \leq \left(1 - \frac{\tilde{\mu}}{\tilde{L}}\right)^{b_k} \left[\delta_{k-1} - \frac{L}{2} \|\nabla f(w_k - 1)\|_2^2\right].
$$

Re-arranging this inequality allows us to control the gradient norm as follows:

$$
\|\nabla f(w_{k-1})\|_2^2 \leq 2L \left[\delta_{k-1} - \delta_{k+b_k}\right] + 2L \left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right] \delta_{k+b_k} \tag{31}
$$

Starting now from the optimality gap and using convexity as well the Cauchy-Schwarz inequality, we obtain

$$
\begin{aligned}
\delta_{k-1} &\leq \langle \nabla f(w_{k-1}), w_{k-1} - w^* \rangle \\
&\leq R \|\nabla f(w_{k-1})\|_2.
\end{aligned}
$$

Substituting in Equation (31) from above and then dividing on both sides by δ_{k-1} and δ_{k+b_k} gives the following:

$$\begin{aligned}
\delta_{k-1}^2 &\leq 2R^2L\left(\delta_{k-1} - \delta_{k+b_k}\right) + 2R^2L\left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right]\delta_{k+b_k} \\
\implies 1 \leq \frac{\delta_{k-1}}{\delta_{k+b_k}} &\leq 2R^2L\left(\frac{1}{\delta_{k+b_k}} - \frac{1}{\delta_{k-1}}\right) + 2R^2L\left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right]\frac{1}{\delta_{k-1}} \\
\implies \frac{1}{2R^2L} &\leq \frac{1}{\delta_{k+b}} - \frac{1}{\delta_{k-1}} + \left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right]\frac{1}{\delta_{k-1}}. \qquad (32)
\end{aligned}$$

Since $k \in \mathcal{B}$, it must be that $k + b_k \notin \mathcal{B}$. Then, either we have $k + b_k + 1 \in \mathcal{B}$ and Equation (32) holds with the choice of $k' = k + b_k + 1$, that is,

$$\frac{1}{2R^2L} \leq \frac{1}{\delta_{k+b_{k'}+b_k+1}} - \frac{1}{\delta_{k+b_k}} + \left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b}\right]\frac{1}{\delta_{k+b_k}},$$

or $k+b_k+1$ is also not a teleportation step and we obtain the following the simpler inequality (see Proposition 2.5):

$$\frac{1}{2R^2L} \leq \frac{1}{\delta_{k+b_k+1}} - \frac{1}{\delta_{k+b_k}}.$$

In either case, these two inequalities form a telescoping series with Equation (32).

Note that this analysis only holds for $k \in \mathcal{T}$ such that $k-1$ was a gradient step. In this special case where $k = 0 \in \mathcal{T}$, we must treat the update at k as a regular gradient step for Equation (32) to hold. However, this causes no problems since every GD step after teleporting can also be analyzed as a vanilla GD step using the descent lemma.

Summing over this telescoping series allows us to obtain the following:

$$\begin{aligned}
\frac{K - |\mathcal{T}|}{2R^2L} &\leq \left(\frac{1}{\delta_K} - \frac{1}{\delta_0}\right) + \sum_{k \in \mathcal{B}}\left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right]\frac{1}{\delta_{k-1}} \\
&\leq \frac{1}{\delta_K} + \sum_{k \in \mathcal{B}}\left[1 - \left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{b_k}\right]\frac{1}{\delta_{k-1}}.
\end{aligned}$$

Finally, re-arranging this expression yields

$$\delta_K \leq \frac{2R^2L}{K - |\mathcal{T}| + 2R^2L\sum_{k\in\mathcal{B}}\left[\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{b_k} - 1\right]\frac{1}{\delta_{k-1}}}.$$

$\square$

Theorem 2.11. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Then GD with line-search and any tele-schedule $\mathcal{T}$ converges as,*

$$\delta_K \leq 2R^2L\Big[\sum_{k\notin\mathcal{T}}\left(\frac{\tilde{L}}{\tilde{L} - \tilde{\mu}}\right)^{n_k}\Big]^{-1}, \qquad (17)$$

where $n_k = |\{i \in \mathcal{T} : i > k\}|$ is the number teleportation steps after iteration k.

Proof. Define the constant,

$$\zeta_{k-1} = \begin{cases} \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{b_k} & \text{if } k \in \mathcal{B} \\ 1 & \text{otherwise.}\end{cases}$$

Using this with Equation (32) implies

$$\begin{aligned}\frac{1}{2R^2L} &\leq \frac{1}{\delta_{k+b_k}} - \frac{1}{\delta_{k-1}} + \left[1 - \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{b_k}\right]\frac{1}{\delta_{k-1}} \\ &= \frac{1}{\delta_{k+b_k}} - \frac{\zeta_{k-1}}{\delta_{k-1}}.\end{aligned}$$

Since $k \in \mathcal{B}$, it must be that $k + b_k \notin \mathcal{B}$. If $k + b_k + 1 \in \mathcal{B}$, then Equation (32) also holds with the choice of $k' = k + b_k + 1$ and

$$\frac{1}{2R^2L} \leq \frac{1}{\delta_{k+b_{k'}+b_k+1}} - \frac{\zeta_{k+b_k}}{\delta_{k+b_k}}.$$

On the other hand, if $k + b_k + 1$ is also not a teleportation step, then we have (see Proposition 2.5):

$$\frac{1}{2R^2L} \leq \frac{1}{\delta_{k+b_k+1}} - \frac{\zeta_{k+b_k}}{\delta_{k+b_k}}.$$

Let $\theta_0 = 1$ and $\theta_{k+b_k} = \theta_{k-1}/\zeta_{k-1}$. We must choose the next θ to do a weighted telescoping of these equations. If $k + b_k + 1 \in \mathcal{T}$, then set $\theta_{k+b_{k'}+b_k+1} = \theta_{k+b_k}/\zeta_{k+b_k}$. On the other hand, if $k + b_k + 1 \notin \mathcal{B}$, then we must set $\theta_{k+b_k+1} = \theta_{k+b_k}/\zeta_{k+b_k}$. In either case, weighting the telescoping series with the θ_i's and summing gives the following:

$$\begin{aligned}\frac{\theta_{k+b_k}}{2R^2L} &\leq \frac{\theta_{k+b_k}}{\delta_{k+b_k}} - \frac{\theta_{k-1}}{\delta_{k-1}} \\ \implies \frac{\sum_{k\notin\mathcal{T}}\theta_k}{2R^2L} &\leq \frac{\theta_K}{\delta_K} - \frac{\theta_0}{\delta_0} \\ &\leq \frac{\theta_K}{\delta_K}.\end{aligned}$$

Re-arranging this equation gives

$$\delta_K \leq \frac{2R^2L}{\sum_{k\notin\mathcal{T},k>0}\theta_k/\theta_K}.$$

To determine the rate of convergence, we must compute θ_k/θ_K for each $k \notin \mathcal{T}$.

By construction, the weight θ_k satisfies,

$$\theta_k = \prod_{i\notin\mathcal{T},i\leq k}\zeta_{i-1} = \prod_{i\in\mathcal{B},i\leq k}\left(\frac{\tilde{L}-\tilde{\mu}}{\tilde{L}}\right)^{b_k} = \left(\frac{\tilde{L}-\tilde{\mu}}{\tilde{L}}\right)^{|\mathcal{T}_i|},$$

where $\mathcal{T}_i = \{i \in \mathcal{T} : i \leq k\}$. Let $n_k = (|\mathcal{T}| - |\mathcal{T}_k|)$, which is the number of teleport steps after iteration k. In this notation, we have $\theta_k/\theta_K = \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{n_k}$, meaning our final convergence rate is given by,

$$\delta_K \leq \frac{2R^2L}{\sum_{k\notin\mathcal{T},k>0}\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{n_k}}.$$

□

Corollary A.4. *Consider the setting of Theorem 2.11. Suppose K is even and that we teleport every-other iteration starting with $k = 1$. Then, GD with teleportation converges as*

$$\delta_K \leq \frac{2R^2L(\tilde{L}-\tilde{\mu})}{\tilde{\mu}\left[\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{K/2} - 1\right]}. \tag{33}$$

Moreover, this is strictly faster than GD without teleportation if

$$K > \log\left(\frac{(\tilde{L}-\tilde{\mu})K + \tilde{\mu}}{\tilde{\mu}}\right) / \log\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right). \tag{34}$$

Proof. Using the fact that $\mathcal{T} = \{1, 3, \ldots, K-1\}$ implies

$$\begin{aligned}
\delta_K &\leq \frac{2R^2 L}{\sum_{i=2; i \text{ even}}^{K} \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{(K-i)/2}} \\
&= \frac{2R^2 L}{\sum_{j=0}^{K/2-1} \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{j}} \\
&= \frac{2R^2 L(\tilde{L}-\tilde{\mu})}{\tilde{\mu}\left[\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{K/2} - 1\right]}.
\end{aligned}$$

This rate is faster than standard GD when

$$\begin{aligned}
&\frac{\tilde{L}-\tilde{\mu}}{\tilde{\mu}\left[\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{K/2} - 1\right]} < \frac{1}{K} \\
\iff &\frac{(\tilde{L}-\tilde{\mu})K + \tilde{\mu}}{\tilde{\mu}} < \left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right)^{K/2} \\
\iff 2\log &\left(\frac{(\tilde{L}-\tilde{\mu})K + \tilde{\mu}}{\tilde{\mu}}\right) / \log\left(\frac{\tilde{L}}{\tilde{L}-\tilde{\mu}}\right) < K.
\end{aligned}$$

□

Theorem A.5. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Consider any teleportation schedule $\mathcal{T}$ and let $M = K - |\mathcal{T}|$ be the number of steps without teleportation. Then GD with step-size $\eta < \frac{2}{L\tilde{L}}$ converges as,*

$$\delta_K \leq \frac{2R^2}{\xi M + 2R^2 \sum_{k \in \mathcal{B}} \frac{\psi_{k-1}}{\delta_{k-1}}}, \tag{35}$$

where $\xi = \eta(2 - L\eta)$, $\psi_{k-1} = \left[1 - \left[\prod_{i=k}^{k+b_k-1} \beta_i\right]\right]$, and $\beta_i = \left(1 + \lambda_i \eta \tilde{\mu}\left(\tilde{L}\lambda_i \eta - 2\right)\right)^{-1}$.

Proof. Since $\eta < \frac{2}{L}$, Lemma 2.1 implies that the iterates — with or without teleportation steps — remain within the initial sub-level set. As a result, Hessian stability holds for all iterates.

Assume $k \in \mathcal{B}$ and let

$$\beta_k^{-1} = \left(1 + \lambda_k \eta \tilde{\mu}\left(\tilde{L}\lambda_k \eta - 2\right)\right).$$

Since $\lambda_k \leq L$ and $\eta < \frac{2}{L\tilde{L}}$, we know that $\beta_k \in (0, 1)$. Using Lemma 2.8 and unrolling Equation (12) for the b_k teleportation steps, we obtain

$$\begin{aligned}
\delta_{k+b_k} &\leq \left[\prod_{i=k}^{k+b_k-1} \beta_i^{-1}\right] \delta_k \\
&\leq \left[\prod_{i=k}^{k+b_k-1} \beta_i^{-1}\right] \left[\delta_{k-1} - \eta\left(1 - \frac{L\eta}{2}\right)\|\nabla f(w_{k-1})\|_2^2\right] \\
&= \left[\prod_{i=k}^{k+b_k-1} \beta_i^{-1}\right] \left[\delta_{k-1} - \frac{\xi}{2}\|\nabla f(w_{k-1})\|_2^2\right],
\end{aligned}$$

where the second inequality follows from the descent lemma (Equation (1)) and $\xi = \eta(2 - L\eta)$. Re-arranging this inequality allows us to control the gradient norm as follows:

$$\|\nabla f(w_{k-1})\|_2^2 \leq \frac{2}{\xi}\left[\delta_{k-1} - \delta_{k+b_k}\right] + \frac{2}{\xi}\left[1 - \left[\prod_{i=k}^{k+b_k-1} \beta_i\right]\right] \delta_{k+b_k} \tag{36}$$

Starting now from the optimality gap and using convexity as well the Cauchy-Schwarz inequality, we obtain

$$\begin{aligned}\delta_{k-1} &\leq \langle \nabla f(w_{k-1}), w_{k-1} - w^* \rangle \\ &\leq R \|\nabla f(w_{k-1})\|_2.\end{aligned}$$

To simplify the notation, it will be very helpful to define the scalar,

$$\psi_{k-1} = \left[1 - \left[\prod_{i=k}^{k+b_k-1} \beta_i\right]\right].$$

Substituting in Equation (36) from above and then dividing on both sides by δ_{k-1} and δ_{k+b_k} gives the following:

$$\begin{aligned}\delta_{k-1}^2 &\leq \frac{2R^2}{\xi}\left(\delta_{k-1} - \delta_{k+b_k}\right) + \frac{2R^2}{\xi}\left[1 - \left[\prod_{i=k}^{k+b_k-1} \beta_i\right]\right]\delta_{k+b_k} \\ \implies 1 \leq \frac{\delta_{k-1}}{\delta_{k+b_k}} &\leq \frac{2R^2}{\xi}\left(\frac{1}{\delta_{k+b_k}} - \frac{1}{\delta_{k-1}}\right) + \frac{2R^2}{\xi}\psi_{k-1}\frac{1}{\delta_{k-1}} \\ \implies \frac{\xi}{2R^2} &\leq \frac{1}{\delta_{k+b}} - \frac{1}{\delta_{k-1}} + \psi_{k-1}\frac{1}{\delta_{k-1}}.\end{aligned} \tag{37}$$

Since $k \in \mathcal{B}$, it must be that $k + b_k \notin \mathcal{B}$. Then, either we have $k + b_k + 1 \in \mathcal{B}$ and Equation (37) holds with the choice of $k' = k + b_k + 1$, that is,

$$\frac{\xi}{2R^2} \leq \frac{1}{\delta_{k+b_{k'}+b_k+1}} - \frac{1}{\delta_{k+b_k}} + \psi_{k+b_k}\frac{1}{\delta_{k+b_k}},$$

or $k+b_k+1$ is also not a teleportation step and we obtain the following the simpler inequality (see Proposition 2.5):

$$\frac{\xi}{2R^2 L} \leq \frac{1}{\delta_{k+b_k+1}} - \frac{1}{\delta_{k+b_k}}.$$

In either case, these two inequalities form a telescoping series with Equation (32). Note that this analysis only holds for $k \in \mathcal{T}$ such that $k-1$ was a gradient step. In this special case where $k = 0 \in \mathcal{T}$, we must treat the update at k as a regular gradient step for Equation (32) to hold. However, this causes no problems since every GD step after teleporting can also be analyzed as a vanilla GD step using the descent lemma.

Summing over this telescoping series allows us to obtain the following:

$$\begin{aligned}\frac{\xi(K - |\mathcal{T}|)}{2R^2} &\leq \left(\frac{1}{\delta_K} - \frac{1}{\delta_0}\right) + \sum_{k \in \mathcal{B}} \psi_{k-1}\frac{1}{\delta_{k-1}} \\ &\leq \frac{1}{\delta_K} + \sum_{k \in \mathcal{B}} \psi_{k-1}\frac{1}{\delta_{k-1}}.\end{aligned}$$

Finally, re-arranging this expression yields

$$\delta_K \leq \frac{2R^2}{\xi(K - |\mathcal{T}|) + 2R^2 \sum_{k \in \mathcal{B}} \psi_{k-1}\frac{1}{\delta_{k-1}}}.$$

□

Corollary A.6. *In the setting of Theorem A.5, if* $\eta = \frac{1}{L\tilde{L}}$*, then*

$$\frac{1}{\xi} \leq L\tilde{L}, \qquad \beta_i \leq \frac{L\tilde{L}}{L\tilde{L} - \lambda_i \tilde{\mu}}, \qquad \psi_{k-1} \leq \left[1 - \left[\prod_{i=k}^{k+b_k-1} \frac{L\tilde{L}}{L\tilde{L} - \lambda_i \tilde{\mu}}\right]\right],$$

and GD with teleportation converges at the following rate:

$$\delta_K \leq \frac{2R^2 L\tilde{L}}{M + 2R^2 L\tilde{L} \sum_{k \in \mathcal{B}} \left[1 - \left[\prod_{i=k}^{k+b_k-1} \frac{L\tilde{L}}{L\tilde{L} - \lambda_i \tilde{\mu}}\right]\right] \frac{1}{\delta_{k-1}}}. \tag{38}$$

Moreover, the dependence on $L\tilde{L}$ can be improved to L if the step-size $\eta = \frac{1}{L}$ is used whenever $k \notin \mathcal{T}$. Finally, if teleportation steps are used every-other iteration, then GD with teleportation satisfies

$$\delta_K \leq \frac{2R^2 L\tilde{L}}{K/2 + 2R^2 L\tilde{L} \sum_{k \in \mathcal{B}} \left[\frac{\lambda_i \tilde{\mu}}{L\tilde{L} - \lambda_i \tilde{\mu}} \frac{1}{\delta_{k-1}} \right]}. \tag{39}$$

Theorem A.7. *Suppose f is L-smooth, convex, and satisfies Hessian stability on $\mathcal{S}_0$. Consider any teleportation schedule $\mathcal{T}$ and let $M = K - |\mathcal{T}|$ be the number of steps without teleportation. Then GD with step-size $\eta < \frac{2}{L\tilde{L}}$ converges as,*

$$\delta_K \leq \frac{2R^2}{\xi \sum_{k \notin \mathcal{T}, k > 0} \left[\prod_{i \notin \mathcal{T}, i > k} \zeta_{i-1} \right]}, \tag{40}$$

where $\xi = \eta\left(2 - L\eta\right)$, $\beta_i = \left(1 + \lambda_i \eta \tilde{\mu} \left(\tilde{L}\lambda_i \eta - 2\right)\right)^{-1}$ and

$$\zeta_{k-1} = \begin{cases} \left[\prod_{i=k}^{k+b_k-1} \beta_i\right] & \textit{if } k \in \mathcal{B} \\ 1 & \textit{otherwise.} \end{cases}$$

Proof. Recall from the proof of Theorem A.5 that $\xi = \eta\left(2 - L\eta\right)$ and

$$\beta_k^{-1} = \left(1 + \lambda_k \eta \tilde{\mu} \left(\tilde{L}\lambda_k \eta - 2\right)\right).$$

Using this notation, define the constant

$$\zeta_{k-1} = \begin{cases} \left[\prod_{i=k}^{k+b_k-1} \beta_i\right] & \text{if } k \in \mathcal{B} \\ 1 & \text{otherwise.} \end{cases}$$

Combing this with Equation (37) implies

$$\begin{aligned} \frac{\xi}{2R^2} &\leq \frac{1}{\delta_{k+b}} - \frac{1}{\delta_{k-1}} + \psi_{k-1}\frac{1}{\delta_{k-1}} \\ &= \frac{1}{\delta_{k+b_k}} - \frac{\zeta_{k-1}}{\delta_{k-1}}. \end{aligned}$$

Since $k \in \mathcal{B}$, it must be that $k + b_k \notin \mathcal{B}$. If $k + b_k + 1 \in \mathcal{B}$, then Equation (32) also holds with the choice of $k' = k + b_k + 1$ and

$$\frac{\xi}{2R^2 L} \leq \frac{1}{\delta_{k+b_{k'}+b_k+1}} - \frac{\zeta_{k+b_k}}{\delta_{k+b_k}}.$$

On the other hand, if $k + b_k + 1$ is also not a teleportation step, then we have (see Proposition 2.5):

$$\frac{\xi}{2R^2} \leq \frac{1}{\delta_{k+b_k+1}} - \frac{\zeta_{k+b_k}}{\delta_{k+b_k}}.$$

Let $\theta_0 = 1$ and $\theta_{k+b_k} = \theta_{k-1}/\zeta_{k-1}$. We must choose the next θ to do a weighted telescoping of these equations. If $k + b_k + 1 \in \mathcal{T}$, then set $\theta_{k+b_{k'}+b_k+1} = \theta_{k+b_k}/\zeta_{k+b_k}$. On the other hand, if $k + b_k + 1 \notin \mathcal{B}$, the we must set $\theta_{k+b_k+1} = \theta_{k+b_k}/\zeta_{k+b_k}$. In either case, weighting the telescoping series with the θ_i's and summing gives the following:

$$\begin{aligned} \frac{\xi\theta_{k+b_k}}{2R^2} &\leq \frac{\theta_{k+b_k}}{\delta_{k+b_k}} - \frac{\theta_{k-1}}{\delta_{k-1}} \\ \implies \frac{\sum_{k \notin \mathcal{T}} \theta_k}{2R^2 L} &\leq \frac{\theta_K}{\delta_K} - \frac{\theta_0}{\delta_0} \\ &\leq \frac{\theta_K}{\delta_K}. \end{aligned}$$

Re-arranging this equation gives

$$\delta_K \leq \frac{2R^2}{\xi \sum_{k \notin \mathcal{T}, k>0} \theta_k / \theta_K}. \tag{41}$$

To determine the rate of convergence, we must compute θ_k/θ_K for each $k \notin \mathcal{T}$.

By construction, the ratio satisfies

$$\theta_k = \prod_{i \notin \mathcal{T}, i \leq k} \zeta_{i-1} \left[\prod_{i \notin \mathcal{T}, i \leq k} \zeta_{i-1} \right]^{-1} = \prod_{i \notin \mathcal{T}, i > k} \zeta_{i-1},$$

meaning our final convergence rate is given by,

$$\delta_K \leq \frac{2R^2}{\xi \sum_{k \notin \mathcal{T}, k>0} \left[\prod_{i \notin \mathcal{T}, i>k} \zeta_{i-1} \right]}.$$

□

Proposition A.8. *In the setting of Theorem A.5, if $\eta = \frac{1}{L\tilde{L}}$, teleportation steps are used every-other iteration, and $K \geq 4\tilde{L}/\tilde{\mu}$, then GD with teleportation satisfies*

$$\delta_K \leq \frac{4R^2 L \tilde{L}^2}{\tilde{\mu}} \left[\prod_{i=2; i \text{ even}}^{K} \left(1 - \frac{\lambda_k \tilde{\mu}}{L\tilde{L}} \right) \right]. \tag{42}$$

Moreover, this rate can be tightened by a factor of $\tilde{L}$ if the step-size $\eta = \frac{1}{L}$ is used at iterations $k \notin \mathcal{T}$.

Proof. Our choice of step-size implies

$$\frac{1}{\xi} \leq L\tilde{L}, \qquad \beta_i^{-1} = 1 + \frac{\lambda_k \tilde{\mu}}{L\tilde{L}} \left(\frac{\lambda_k}{L} - 2 \right), \qquad \zeta_{k-1} = \beta_i^{-1},$$

Similarly, bounding $0 \leq \lambda_k \leq L$ and $\eta = \frac{1}{L\tilde{L}}$ gives

$$1 - \frac{2\tilde{\mu}}{\tilde{L}} \leq 1 - \frac{2\lambda_k \tilde{\mu}}{L\tilde{L}} \leq \quad \zeta_{k-1}^{-1} = 1 + \frac{\lambda_k \tilde{\mu}}{\tilde{L}L} \left(\frac{\lambda_k}{L} - 2 \right) \quad \leq 1 - \frac{\lambda_k \tilde{\mu}}{L\tilde{L}}. \tag{43}$$

Using the geometric series we have that

$$\begin{aligned}
\sum_{k=2; k \text{ even}}^{K} \theta_k &= \sum_{k=2; k \text{ even}}^{K} \left[\prod_{i=k+2; i \text{ even}}^{K} \right] \gamma_i^{-1} \\
&\geq \sum_{j=0}^{K/2-1} \left(1 - \frac{2\tilde{\mu}}{\tilde{L}} \right)^j \\
&= \frac{1 - \left(1 - \frac{2\tilde{\mu}}{\tilde{L}} \right)^{K/2}}{2\tilde{\mu}/\tilde{L}}.
\end{aligned}$$

Let $\kappa = \frac{\tilde{L}}{\tilde{\mu}}$ and choose K so that

$$\left(1 - \frac{2}{\kappa^{-1}} \right)^{K/2} \leq \frac{1}{2} \quad \iff \quad \frac{\log(2)}{\log\left(1 + \frac{2}{\kappa^{-1} - 2} \right)} \leq K/2. \tag{44}$$

To simplify the above expression we can use,

$$\frac{x}{1+x} \leq \log(1+x) \leq x, \quad \text{for } x > -1,$$

since $\kappa^{-1} \leq 1$, with strictness or else one step of GD after teleporting solves the optimization problem. Using this inequality implies

$$\frac{\log(2)}{\log\left(1 + \frac{2}{\kappa^{-1} - 2}\right)} \leq 2\kappa \leq \frac{K}{2} \quad \iff \quad \frac{4\tilde{L}}{\tilde{\mu}} \geq K.$$

Using the above to control the geometric series gives,

$$\sum_{k=2;k \text{ even}}^{K} \left[\prod_{i=k+2;i \text{ even}}^{K}\right] \gamma_i^{-1} \geq \frac{\tilde{L}}{4\tilde{\mu}}, \tag{45}$$

if $K \geq 4\tilde{L}/\tilde{\mu}$. Using this together with Equation (41) gives that the convergence rate is at least

$$\begin{aligned}
\delta_K &\leq \frac{2R^2}{\xi \sum_{k \notin \mathcal{T}, k>0} \theta_k / \theta_K} \\
&\leq \frac{4R^2 L \tilde{L}^2}{\tilde{\mu}} \left[\prod_{i=2;i \text{ even}}^{K} \left(1 - \frac{\lambda_k \tilde{\mu}}{L\tilde{L}}\right)\right],
\end{aligned}$$

if $K \geq 4\tilde{L}/\tilde{\mu}$.

To see that the rate can be tightened, observe that using $\eta_{k-1} = 1/L$ when $k - 1 \notin \mathcal{T}$ tightens the value of ξ to $\frac{1}{\xi} = L$ without affecting the proof. □

B EVALUATING THE TELEPORTATION OPERATOR: PROOFS

Proposition 3.1. *Equation* (19) *is solved by,*

$$\begin{aligned}
v_t &= (\rho_t \langle \nabla f(x_t), q_t \rangle / g_t + f(x_t) - f(w_k))_+ \nabla f(x_t), \\
x_{t+1} &= x_t + (\rho_t q_t - v_t) / g_t.
\end{aligned} \tag{20}$$

Proof. We proceed by case analysis. Let

$$\bar{x} = \arg\max_{x \in \mathbb{R}^d} \left\{ \frac{1}{2} \log(\|\nabla f(x_t)\|_2^2) + \left\langle \frac{\nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2}, x - x_t \right\rangle - \frac{1}{2\rho_t} \|x - x_t\|_2^2 \right\}.$$

Taking derivatives shows that $\bar{x}$ is a gradient ascent step with step-size ρ_t,

$$\bar{x} = x_t + \rho_t \frac{\nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2}.$$

Case 1: $\bar{x} \in \tilde{\mathcal{S}}_k(x_t)$. Then $\bar{x}$ satisfies the linearized constraint and $x_{t+1} = \bar{x}$ must hold. We conclude that,

$$x_{t+1} = x_t + \rho_t \frac{\nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2}.$$

Substituting this value into the linearization of the half-space and using $x_{t+1} \in \tilde{\mathcal{S}}_k(x_t)$, we find

$$\rho_t \frac{\nabla f(x_t)^\top \nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2} + f(x_t) - f(w_k) \leq 0.$$

Case 2: $\bar{x} \notin \tilde{\mathcal{S}}_k(x_t)$. Then the solution lies on the boundary of the half-space constraint and is given by projecting $\bar{x}$ onto

$$\tilde{\mathcal{L}}_k(x_t) = \{x : f(x_t) + \langle \nabla f(x_t), x - x_t \rangle = f(w_k)\}.$$

The Lagrangian of this problem is

$$L(x, \lambda) = \frac{1}{2}\|x - \bar{x}\|_2^2 + \lambda \left(f(x_t) + \langle \nabla f(x_t), x - x_t \rangle - f(w_k)\right).$$

Minimizing over x yields

$$x_{t+1} = \bar{x} - \lambda \nabla f(x_t),$$

which shows that the dual function is given by,

$$\begin{aligned} d(\lambda) &= -\frac{1}{2}\lambda^2 \|\nabla f(x_t)\|_2^2 + \lambda \left(f(x_t) + \langle \nabla f(x_t), \bar{x} - x_t \rangle - f(w_k)\right) \\ &= -\frac{1}{2}\lambda^2 \|\nabla f(x_t)\|_2^2 + \lambda \left(f(x_t) + \frac{\rho_t}{\|\nabla f(x_t)\|_2^2} \left\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t)^2 \right\rangle - f(w_k) \right) \end{aligned}$$

This is a concave quadratic; maximizing over λ gives the following dual solution:

$$\lambda^* = \frac{f(x_t) - f(w_k) + \rho_t \left\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t) \right\rangle / \|\nabla f(x_t)\|_2^2}{\|\nabla f(x_t)\|_2^2}.$$

Note that

$$f(x_t) - f(w_k) + \rho_t \left\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t) \right\rangle / \|\nabla f(x_t)\|_2^2 > 0,$$

since the half-space constraint is violated at $\bar{x}$. Plugging this value back into the expression for x_{t+1},

$$\begin{aligned} x_{t+1} = x_t + \rho_t & \frac{\nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2} \\ & - \frac{\rho_t \left\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t) \right\rangle / \|\nabla f(x_t)\|_2^2 + f(x_t) - f(w_k)}{\|\nabla f(x_t)\|_2^2} \nabla f(x_t). \end{aligned}$$

This completes the second case. Putting the analysis together, we obtain the desired result:

$$\begin{aligned} x_{t+1} = x_t + \rho_t & \frac{\nabla^2 f(x_t) \nabla f(x_t)}{\|\nabla f(x_t)\|_2^2} \\ & - \left(\frac{\rho_t \left\langle \nabla f(x_t), \nabla^2 f(x_t) \nabla f(x_t) \right\rangle / \|\nabla f(x_t)\|_2^2 + f(x_t) - f(w_k)}{\|\nabla f(x_t)\|_2^2} \right)_+ \nabla f(x_t). \end{aligned}$$

□

Theorem 3.2. *Assume $\nabla f(w_k) \neq 0$, strict complementarity, and that Equation (20) is used with the relaxation step $\hat{x}_{k+1} = \alpha_t x_{t+1} + (1 - \alpha_t) x_t$. Then, using appropriate (α_t, ρ_t), $\hat{x}_k$ converges asymptotically to a KKT point (x^*, λ^*) of Equation (3); if (x^*, λ^*) is second-order critical, then local convergence is linear.*

Proof. The convergence guarantees follow from the connection between Equation (19) and SQP as developed by Torrisi et al. (2018). Equation (20) gives an exact solution to the SQP update in Torrisi et al. (2018, Equation 2.3) with the setting $H^{(i)} = \frac{1}{\rho_t}$ and the parameterization $d_z = x_{t+1} - x_t$. To obtain their guarantee, we must also explicitly form the dual variables λ_t associated with the linearized constraint, but these are only needed to select the step-size α_t used in the relaxation step. Note that the values for the dual variables can be computed in closed form from the KKT conditions for the linearized constraint.

Now, we do require (α_t, ρ_t) to be "appropriately selected" for convergence guarantees to hold. Torrisi et al. (2018) choose $\alpha_t \leq 1$ using the strong Wolfe conditions (Wolfe, 1969; Wolfe, 1971) on the augmented Lagrangian function (this explains the need to maintain explicit values for the dual parameter λ_t). Note that the augmented Lagrangian function also requires a penalty parameter δ_t. Torrisi et al. (2018) set this parameter to ensure descent in a similar fashion to our choice for the merit function in Proposition 3.3.

No conditions on $\rho_t > 0$ are required for global convergence of the algorithm except boundedness. For local linear convergence, the step-size ρ_t must satisfy $\rho_t \leq 1/\lambda_{\max}\left(\nabla^2 \mathcal{L}(x^*, \lambda^*)\right)$. This is a typical assumption for fast local convergence of SQP methods. The requirement that $\nabla f(w_k) \neq 0$ is sufficient to guarantee that implies LICQ holds, as discussed in Section 2. □

Lemma B.1. *The directional derivative of the merit function satisfies*

$$D_\phi(x_t; d_t) \geq \left(\rho_t \|\nabla^2 f(x_t) \nabla f(x)\|_2^2 - \left\langle \nabla^2 f(x_t) \nabla f(x_t), v_t \right\rangle\right) / \|\nabla f(x_t)\|_2^4 + \gamma \left(f(x_t) - f(w_k)\right)_+ . \tag{46}$$

Moreover, if $\gamma_t > \langle q_t, v_t \rangle / \|\nabla f(x_t)\|_2^4 (f(x_t) - f(w_k))_+$*, then* d_t *is a ascent direction for* ϕ_γ*.*

Proof. Let $d_t = x_{t+1} - x_t$. Define $\Delta_t(\alpha) = \phi_\gamma(w^t + \alpha d_t) - \phi_\gamma(x_t)$. Using first-order Taylor expansions, we have

$$\begin{aligned}
\Delta_t(\alpha) &= \frac{1}{2} \log(\|\nabla f(x_t + \alpha d_t)\|_2^2) - \gamma \left(f(x_t + \alpha d_t) - f(w_k)\right)_+ - \frac{1}{2} \log(\|\nabla f(x_t)\|_2^2) + \gamma \left(f(x_t) - f(w_k)\right)_+ \\
&= \frac{\alpha}{\|\nabla f(x_t)\|_2^2} \left\langle \nabla^2 f(x_t) \nabla f(x_t), d_t \right\rangle - \gamma \left(f(x_t) + \alpha \left\langle \nabla f(x_t), d_t \right\rangle - f(w_k)\right)_+ + \gamma \left(f(x_t) - f(w_k)\right)_+ \\
&\qquad + O(\alpha^2) \\
&\geq \frac{\alpha}{\|\nabla f(x_t)\|_2^2} \left\langle \nabla^2 f(x_t) \nabla f(x_t), d_t \right\rangle - \gamma (1 - \alpha) \left(f(x_t) - f(w_k)\right)_+ + \gamma \left(f(x_t) - f(w_k)\right)_+ + O(\alpha^2),
\end{aligned}$$

where we have used $\langle \nabla f(x_t), d_t \rangle \leq f(w_k) - f(x_t)$ since x_{t+1} satisfies the linearized sub-level set constraint. Simplifying, we obtain,

$$\Delta_t(\alpha) \geq \frac{\alpha}{\|\nabla f(x_t)\|_2^2} \left\langle \nabla^2 f(x_t) \nabla f(x_t), d_t \right\rangle + \gamma \alpha \left(f(x_t) - f(w_k)\right)_+ + O(\alpha^2).$$

Dividing both sides by α and taking the limit as $\alpha \to 0$ shows that

$$\begin{aligned}
D_\phi(x_t; d_t) &\geq \frac{1}{\|\nabla f(x_t)\|_2^2} \left\langle \nabla^2 f(x_t) \nabla f(x_t), d_t \right\rangle + \gamma \left(f(x_t) - f(w_k)\right)_+ \\
&= \left(\rho_t \|\nabla^2 f(x_t) \nabla f(x)\|_2^2 - \left\langle \nabla^2 f(x_t) \nabla f(x_t), v_t \right\rangle\right) / \|\nabla f(x_t)\|_2^4 + \gamma \left(f(x_t) - f(w_k)\right)_+ .
\end{aligned}$$

Substituting the choice of γ_t into the directional derivative yields,

$$\begin{aligned}
D_\phi(x_t; d_t) &\geq \left(\rho_t \|\nabla^2 f(x_t) \nabla f(x)\|_2^2 - \left\langle \nabla^2 f(x_t) \nabla f(x_t), v_t \right\rangle\right) / \|\nabla f(x_t)\|_2^4 + \gamma \left(f(x_t) - f(w_k)\right)_+ \\
&> \rho_t \|\nabla^2 f(x_t) \nabla f(x)\|_2^2 / \|\nabla f(x_t)\|_2^4 \geq 0.
\end{aligned}$$

Since this quantity is strictly positive, d_t is an ascent direction for ϕ_γ. □

Proposition 3.3. *If* $f(x_t) > f(w_k)$ *and* $\gamma_t > \langle q_t, v_t \rangle / \|\nabla f(x_t)\|_2^4 (f(x_t) - f(w_k))_+$*, then* $x_{t+1} - x_t$ *is an ascent direction of* ϕ_{γ_t} *and Eq.* (21) *simplifies to,*

$$\phi_{\gamma_t}(x_{t+1}) \geq \phi_\gamma(x_t) + \rho_t \|q_t\|_2^2 / 2 g_t^2 . \tag{22}$$

Proof. The first part of the proof follows immediately from Lemma B.1. Substituting the choice of γ_t into the line search condition yields,

$$\begin{aligned}
\phi_\gamma(x_{t+1}) &\geq \phi_\gamma(x_t) + D_\phi(x_t, x_{t+1} - x_t)/2 \\
&> \phi_\gamma(x_t) + \rho_t \|\nabla^2 f(x_t) \nabla f(x)\|_2^2 / (2 \|\nabla f(x_t)\|_2^4)
\end{aligned}$$

which is straightforward to check in practice. □

Proposition B.2. *Suppose* g *is a loss function and* $f(w) = g(h_w(X), y)$*, where*

$$h_w(X) = \phi(W_l \phi(\ldots W_2(\phi(W_1 X)))$$

is the prediction function of a neural network with weights $w = (W_l, \ldots, W_1)$*,* $l \geq 2$*. If the activation function* ϕ *is positively homogeneous such that* $\lim_{\beta \to \infty} \phi(\beta) = \infty$ *and* $\lim_{\beta \to \infty} \phi'(\beta) < \infty$*, then optimal value of the sub-level set teleportation problem is unbounded and Equation* (3) *does not admit a finite solution.*

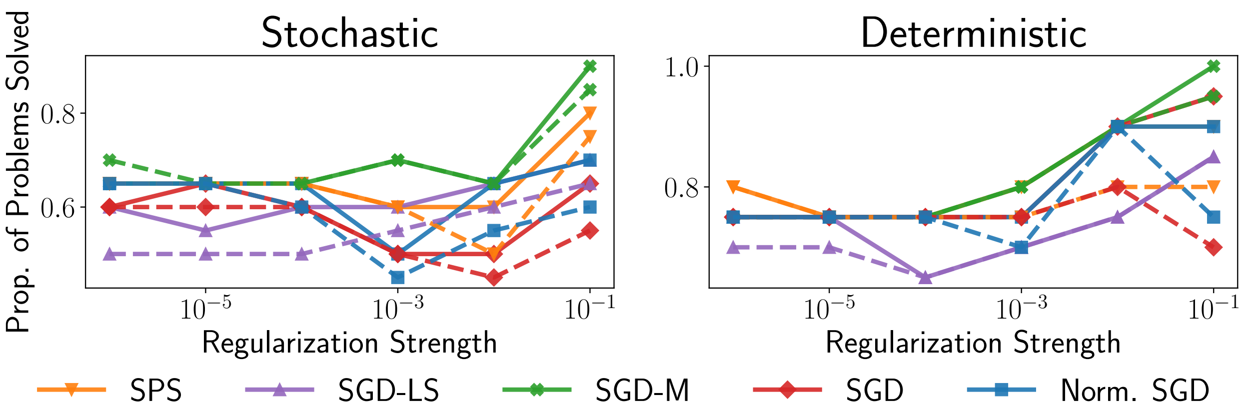

Figure 7: Effect of regularization strength on optimization when training three-layer ReLU networks on 20 datasets from the UCI repository. Success ratios are computed using the same methodology as in Figure 4. Methods using teleportation steps (solid lines) outperform methods without (dashed) when regularization is moderate.

Proof. For simplicity, we prove the result in the scalar case for $l = 2$, although it immediately generalizes. Since ϕ is positively homogeneous, we have

$$w_2\phi(w_1 x) = \alpha w_2 \phi((w_1/\alpha)x).$$

Let $\tilde{w}_2 = \alpha w_2$ and $\tilde{w}_1 = w_1/\alpha$. Define $v = w_2\phi(w_1 x) = \tilde{w}_2\phi(\tilde{w}_1 x)$. Then gradients with respect to the first and second layers are given by

$$\begin{aligned}
\frac{\partial}{\partial \tilde{w}_1} f(w) &= \frac{\partial}{\partial v} g(v) w_2 \phi'(\tilde{w}_1 x) x \\
&= \alpha \frac{\partial}{\partial v} g(v) w_2 \phi'(w_1 x/\alpha) x \\
\frac{\partial}{\partial \tilde{w}_2} f(w) &= \frac{\partial}{\partial v} g(v) \phi(\tilde{w}_1 x) \\
&= \frac{\partial}{\partial v} g(v) \phi(w_1 x/\alpha).
\end{aligned}$$

Taking the limit as $\alpha \to 0$, we see that

$$\begin{aligned}
\frac{\partial}{\partial \tilde{w}_1} f(w) &\to 0 \\
\frac{\partial}{\partial \tilde{w}_2} f(w) &\to \infty,
\end{aligned}$$

by assumption on ϕ. Thus, there exists a diverging sequence of points on the level set whose gradient norm is also diverging. Since the level sets are unbounded, the objective is not coercive and the problem is ill-posed. This completes the proof. $\square$

C EXPERIMENTS

C.1 Additional Experiments

Now we provide further experimental results which could not be included in the main paper due to space constraints.

Effects of Regularization: The teleportation problem may not admit a solution when training neural networks with homogeneous activations (e.g. ReLU) since the objective is not coercive and the level sets are not compact (see Proposition B.2). In practice, we "compactify" the level sets using regularization. Figure 7 compares the strength of weight decay regularization against the success rates of methods with and without teleportation. As expected, all methods solve more problems when the regularization is large. However, while our intuition

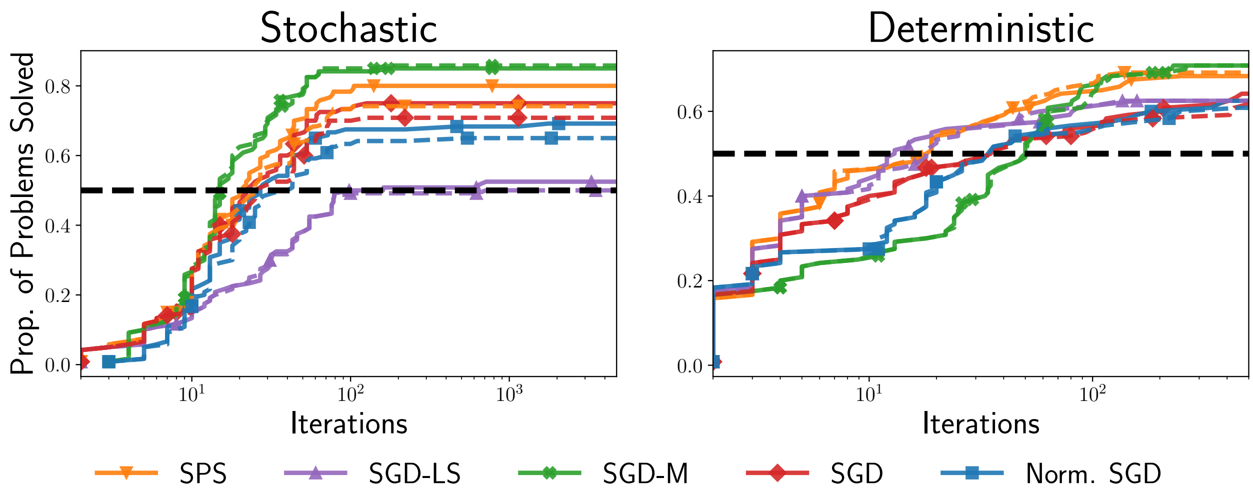

Figure 8: Performance profile comparing stochastic and deterministic optimization methods with (solid lines) and without (dashed lines) sub-level set teleportation for logistic regression models. Stochastic methods teleport once every 10 epochs starting from the fifth epoch, while deterministic methods teleport once every 50 iterations starting from iteration five. A problem is solved when $(f(w_k) - f(w^*))/(f(w_0) - f(w^*)) \leq \tau$, where $f(w^*)$ is estimated separately and τ is a threshold.

suggests that methods with teleportation should solve more problems than standard gradient methods when the regularization is large, this is not obviously true. For example, standard SGD-M outperforms SGD-M with teleportation when regularization is both very large and very small. This indicates that teleportation may work best with moderate regularization.

Logistic Regression Performance Profile: We generate an alternative version of the performance profile from the main paper (Figure 4) using the logistic regression model class instead of two-layer neural networks. Logistic regression is non-strongly convex, but satisfies the Hessian stability condition we use in Section 2.2 to establish fast convergence rates for GD with teleportation. Thus, we expect gradient methods with teleportation to perform better than their counterparts without.

Figure 8 shows that including intermittent teleportation steps improves the performance of every method. This is particularly true for SGD with momentum (SGD-M), which solves nearly every problem when augmented with teleportation steps in both the stochastic and deterministic settings. The poor performance of SGD with line-search (SGD-LS) in the stochastic setting is due to the unreliability of step-sizes chosen by searching along stochastic gradients, which may not be descent directions with respect to the true gradient. Typically interpolation is required for SGD-LS to converge when used with stochastic gradients (Vaswani et al., 2019). Interpolation does not hold in our setting since we consider regularized problems.

Initialization by Teleportation: One notable aspect of our theory is that it suggests not teleporting at the first iteration. This is because our proof technique requires that each block of iterations with teleportation be unrolled back to a single GD step without teleportation; this GD step then allows us to control the norm of the gradient using the descent lemma (Equation (1)). We overcome this limitation by simply treating the GD step at w_0 as a standard gradient update regardless of teleportation, but the result is that we do not benefit theoretically from teleporting at w_0. Thus, it is interesting to examine whether very early teleportation steps are useful for optimization.

Figures 9 and 10 present performance profiles for gradient methods with teleportation at w_0 only (i.e. initialization by teleportation) and methods without teleportation. Figure 9 shows results for logistic regression, while Figure 10 concerns training three-layer ReLU networks. Aside from teleportation schedule, the experimental setup is identical to that for Figures 8 and 4.

We find that initialization by teleportation is broadly detrimental for logistic regression problems, but useful for optimization methods which use adaptive step-sizes (SPS, Norm. SGD, SGD-LS) when training ReLU networks.

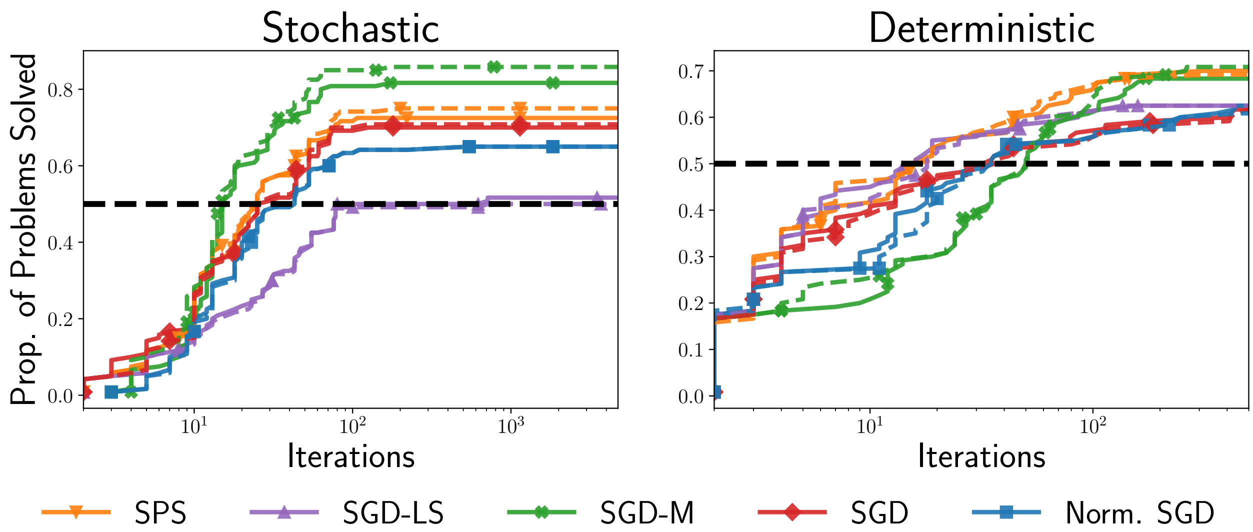

Figure 9: Performance profile comparing stochastic and deterministic optimization methods with (solid lines) and without (dashed lines) initialization by teleportation for training logistic regression models. The experimental configuration is the same as in Figure 8. Initialization by teleportation harms the performance of all methods.

These conclusions are particularly true for training neural networks in the stochastic setting. We conjecture this behavior is because of the strong curvature of optimization problems near the initialization, which is exacerbated by teleportation. Optimizers with adaptive step-sizes can leverage this curvature, while SGD-M and SGD cannot.

Timed Performance Profiles: In practice, evaluating the teleportation oracle introduces a computational overhead. The amount of overhead and the degree of accuracy to which the teleportation sub-problem is solved is determined the number of iterations k used by Algorithm 2. To measure this overhead, we now compare the performance of optimization methods with and without teleportation with respect to total compute time as k is increased from 1 to 50. Figure 11 shows a version of Figure 8 (logistic regression) with time along the x-axis, while Figure 12 gives a version of Figure 4 (three-layer ReLU network).

We see similar trends for both the convex logistic regression problems and non-convex neural network training problems. When $k = 1$, teleportation has minimal overhead but performance is essentially unchanged from the default gradient methods. Increasing k to 10 or 25 increases the time cost, but with only marginal gains. However, for $k = 50$, teleportation methods exceed or match the performance of methods without teleportation while using the same time budget. This trend is particularly noticeable for three-layer ReLU networks in the stochastic setting. We conclude that (i) solving the teleportation sub-problem to sufficient accuracy is critical for good performance and (ii) methods with teleportation are more effective than standard methods given sufficient time budget.

Stochastic and Non-Smooth Image Classification: We replicate our experiments on MNIST with stochastic gradients using a batch-size of 128 in Figure 13. The experimental setup is otherwise identical to that for Figure 5. In this stochastic setting, we find almost no difference between methods with or without teleportation. Indeed, all methods appear to converge (noisily) to similar performing models.

We also consider training non-smooth ReLU neural networks on MNIST. Although we did not find a meaningful difference between non-smooth and smooth activations in our experiments on the UCI datasets[1], smooth activation functions appear more important for teleportation on MNIST. This is consistent with our theory, which requires f to be differentiable and L-smooth.

Figure 14 shows the results of training a two-layer ReLU neural network with 500 hidden units. While SGD and the Polyak step-size (SPS) still perform better with teleportation, SGD-LS stalls and SGD-M is quite unstable when used with teleportation.

[1]Methods with teleportation performed very slightly worse compared to those without teleportation when using the soft-plus activation on the UCI datasets, but the overall trend in performance profiles was identical.

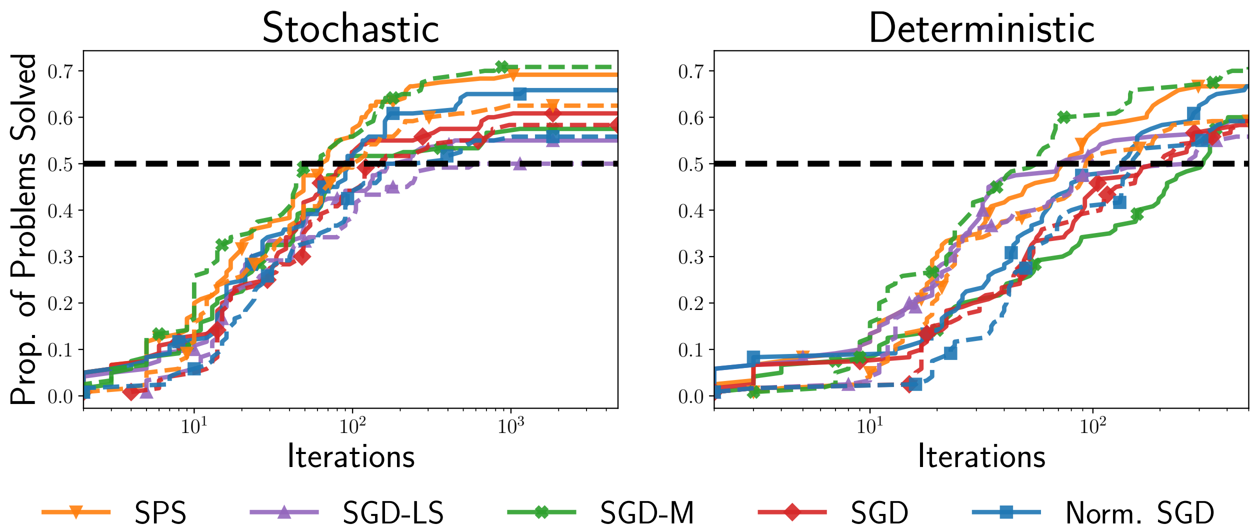

Figure 10: Performance profile comparing stochastic and deterministic optimization methods with (solid lines) and without (dashed lines) initialization by teleportation for training three-layer ReLU networks. The experimental configuration is the same as in Figure 8. Initialization by teleportation slightly improves the performance of methods with adaptive step-sizes (SPS, Norm. SGD), but greatly worsens that of SGD-M.

Full Comparison with Truncated Newton Method: Here we include additional details for the experimental comparison between GD with teleportation at every step and Newton-CG. In particular, we training curves for the test accuracy and gradient norm which were omitted from the main paper. These results further confirm our theoretical findings.

Figure 15 presents full results for three convex logistic regression problems. While Newton-CG enjoys fast convergence initially, it stalls on the Hill Valley and Chess datasets. This is consistent with existing results on truncated Newton methods. For example, the linear rate proved by Karimireddy et al. (2019) only holds under a hard-to-verify assumption on the quality of the Hessian approximation. Since Hill Valley has $d = 100 \gg 25$ features, this assumption likely fails and Newton-CG performs poorly. In contrast, GD with teleportation shows robust, linear convergence. This linear convergence rate confirms our theoretical results in Theorems 2.10 and 2.11.

Figure 16 presents full results for the non-convex neural network training problem. We find that Newton-CG fails on all three problems, likely because it is attracted to any critical point, including saddles and local maxima. This replicates well-known failure modes of Newton-type methods for non-convex optimization (Xu et al., 2020). In comparison, GD with teleportation still obtains a linear rate. The robustness of teleportation for non-convex optimization is one of our primary motivations for this paper.

Convergence Curves for UCI Datasets: Finally, we provide convergence plots for a subset of problems from our performance profiles on the UCI datasets. We select these problems to illustrate various points about the effectiveness of teleportation as an optimization sub-routine. Figure 17 shows convergence for three logistic regression problems on the `Chess`, `Ionosphere`, and `TicTacToe` datasets. Note that we use weight decay regularization with strength $\lambda = 10^{-6}$ and consider deterministic optimization (the plots for stochastic optimization were very noisy).

We also provide convergence plots for training ReLU networks on the `Credit Approval`, `Pima`, and `Ringnorm` datasets (Figure 18). We use weight decay regularization with strength $\lambda = 0.01$ and again consider only deterministic optimization. All three training problems show a very interesting phenomena where several methods with teleportation — notably SGD and Normalized SGD — actually take bad steps after teleporting for which the objective increases. Despite this, they converge to a similar or better final objective value than methods without teleportation. On `Pima`, they appear to convergence to a different local optimum, implying that teleportation can affect the implicit regularization of optimization methods.

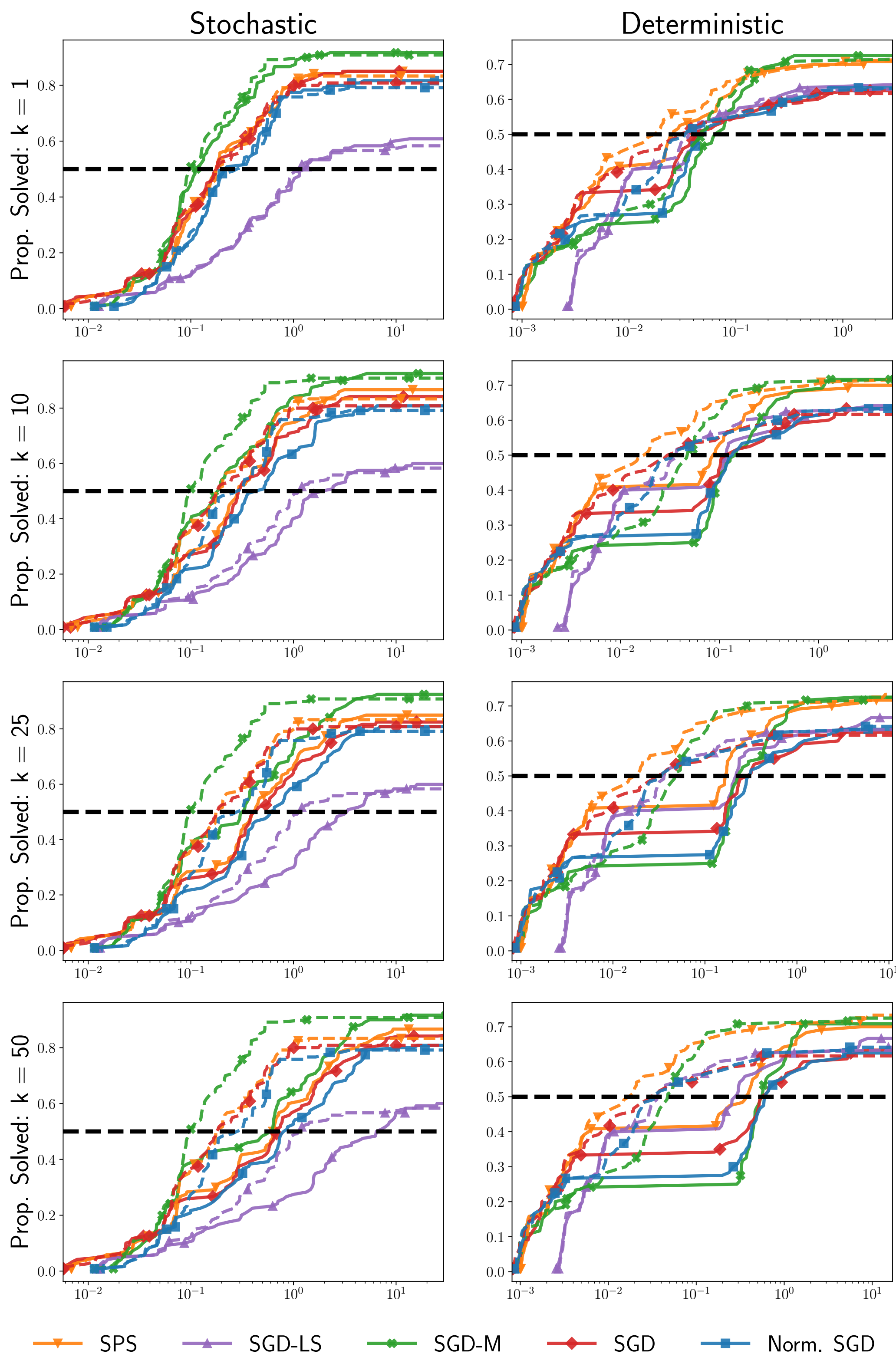

Figure 11: Performance profile comparing stochastic and deterministic optimization methods with (solid lines) and without (dashed lines) teleportation on 120 logistic regression problems. We restrict teleportation methods to k steps of Algorithm 2 to control the time cost and accuracy of the teleportation sub-solver. Unlike Figure 8, the x-axis shows cumulative time, including the time required to teleport, against the proportion of problems solved. Although teleportation introduces a significant computational overhead, methods with teleportation eventually catch-up to those without.

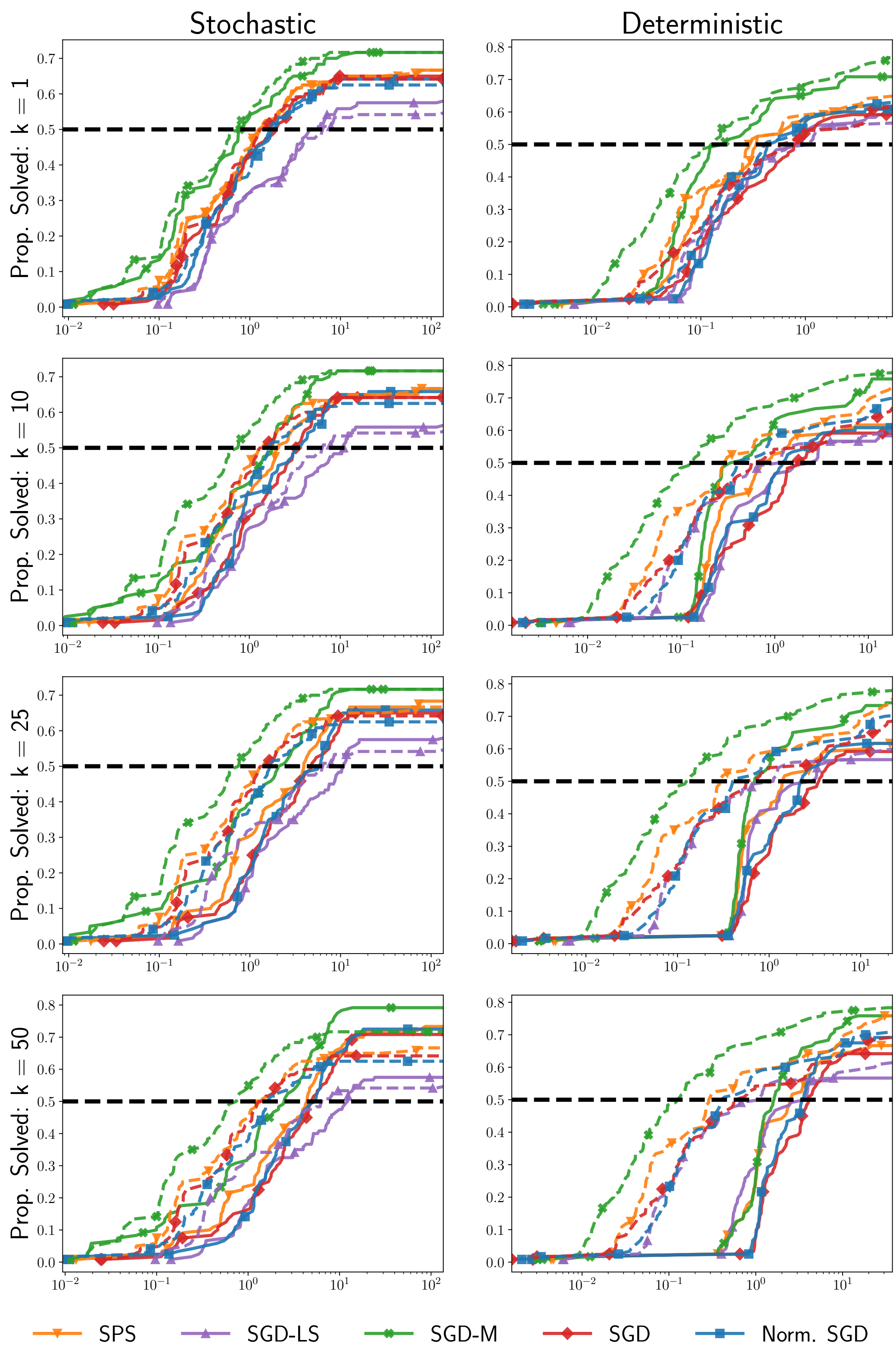

Figure 12: Performance profile comparing stochastic and deterministic optimization methods with (solid lines) and without (dashed lines) teleportation for training three-layer ReLU networks. We restrict teleportation methods to k steps of Algorithm 2 to control the time cost and accuracy of the teleportation sub-solver. The x-axis shows cumulative time as in Figure 11. We find solving the teleportation sub-problem to high accuracy is critical to good performance of gradient methods with teleportation.

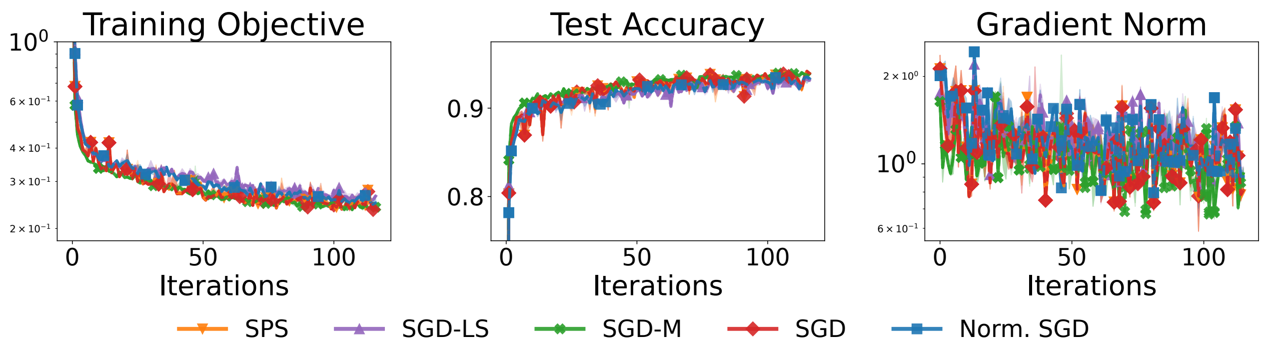

Figure 13: Performance of optimizers with (solid) and without (dashed) teleportation on MNIST. We train a MLP with the soft-plus activation and one hidden layer of size 500. All methods are run with batch-size of 128. Teleportation does not appear to help or harm optimization.

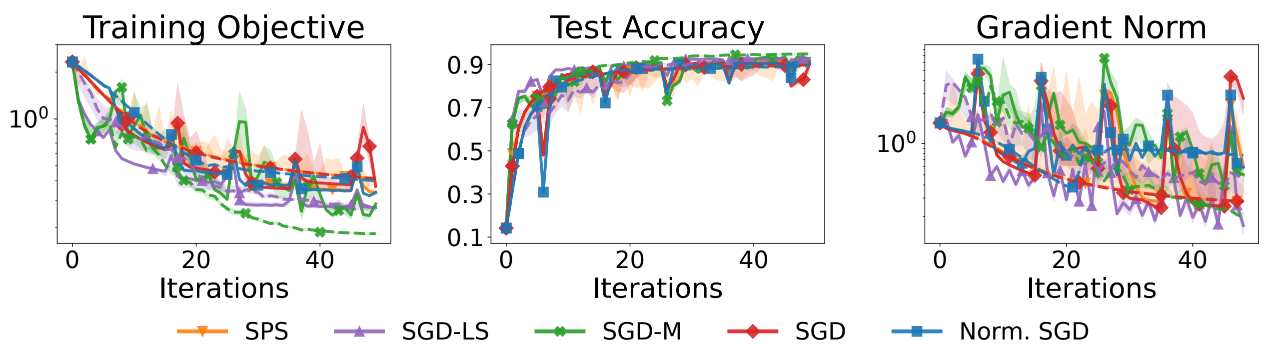

Figure 14: Convergence of deterministic gradient methods for training a two-layer ReLU network with 500 hidden units on the MNIST dataset. Solid lines show methods with teleportation, while dashed lines show the same algorithms without teleportation. Unlike when using smooth activations, teleportation has mixed effectiveness. While SGD and the Polyak step-size (SPS) still perform better with teleportation, SGD-LS stalls and SGD-M is quite unstable when used with teleportation.

C.2 Experimental Details

In this section we include additional details necessary to replicate our experiments. The code used to run the experiments in this paper is publicly available on GitHub at `https://github.com/aaronpmishkin/teleport`. We run our experiments using PyTorch (Paszke et al., 2019). When selecting step-sizes for optimization methods, we perform a grid-search using the grid $\{1000, 100, 10, 5, 2, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$. All experimental results are shown for three random restarts excepting the deterministic Newton experiments and the performance profiles, where averaging is not straightforward. We plot the median and first/third quartiles. Step-sizes are selected by minimizing the training loss at the end of the last epoch.

For our teleportation method, we initialize the step-size at $\rho = 0.1$ and use the tolerances $\epsilon = \delta = 10^{-6}$. In practice, we scale γ_t by 0.1 to bias the teleportation solver towards more aggressive maximization of the gradient norm. Unless otherwise stated, we limit the teleportation method to a maximum of 50 iterations, although it can terminate earlier based on the tolerances. We allow the teleportation method to backtrack at most 25 iterations before returning a default step-size of $\rho_t = 10^{-16}$.

For SGD with momentum, we use the momentum parameter $\beta = 0.9$ and dampening parameter $\mu = 0.9$. For SGD-LS, we optimize over relaxation parameters for the Armijo line-search and choose the best from the set $\{0.001, 0.01, 0.1, 0.5\}$. To ensure the step-size returned by line-search is always close to the maximum attainable, at each iteration we increase the step-size until the Armijo condition fails before backtracking. We use forward-tracking and back-tracking parameters of 1.25 and 0.8, respectively. We estimate f^* with zero for SPS. All neural networks are initialized using the standard Kaiming initialization (He et al., 2015).

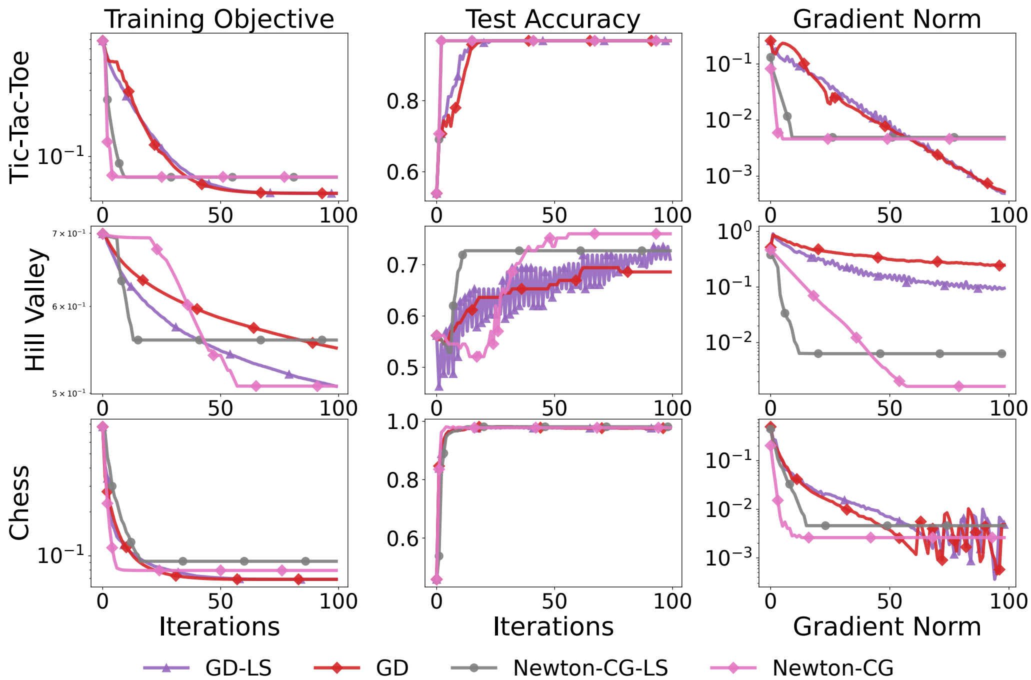

Figure 15: Comparison of inexact Newton and GD with teleportation using tuned step-sizes and using line-search (LS) for three logistic regression problems satisfying Hessian stability. The inexact Newton step is computed using conjugate gradients. For fairness, Newton-CG and GD with teleportation are limited to 25 Hessian-vector products at each step. While inexact Newton initially converges quickly, it is sensitive to the eigenvalue spectrum of the Hessian and convergence stalls on two out of three datasets.

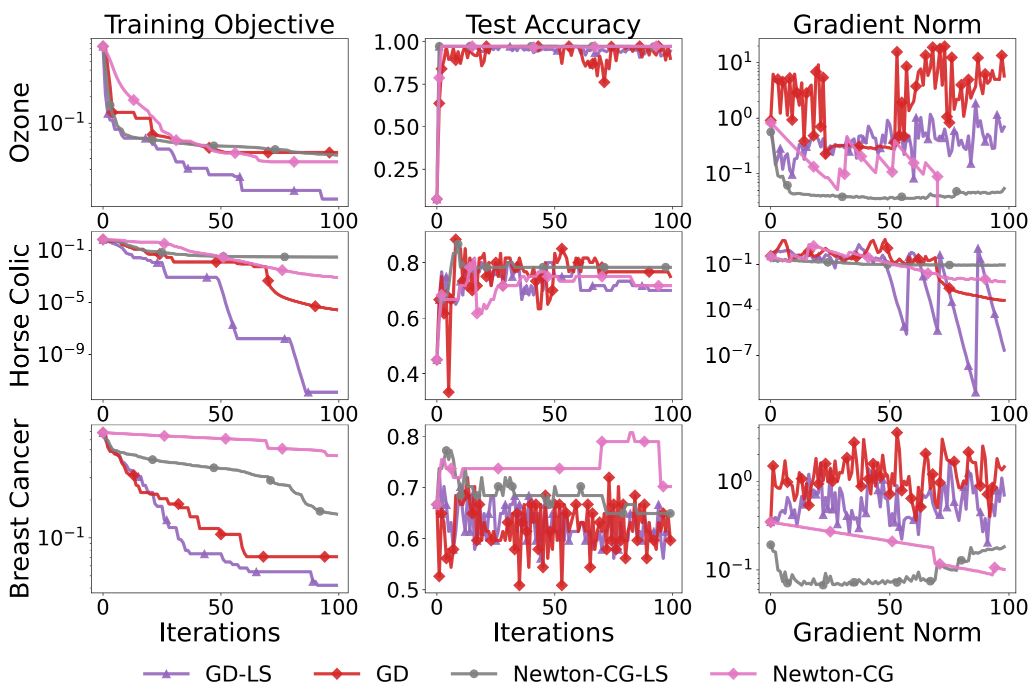

Figure 16: Comparison of inexact Newton and GD with teleportation for training two-layer ReLU networks. Both methods are limited to 25 Hessian-vector products. We find that GD with teleportation converges at a fast rate while Newton-CG fails on every problem due to non-convexity.

Test Functions: We use open-source implementations of the Booth and Goldstein-Price functions [2]. Gradient descent with and without teleportation are run with an Armijo line-search starting from step-size $\eta = 1$. Newton's method is run with a fixed step-size $\eta = 0.8$.

Solving the Teleportation Problem: We use a two-layer ReLU network with fifty hidden units and soft-plus activation function. The strength of weight decay regularization is set at $\lambda = 1.8$. We initialize the teleportation method with step-size $\rho = 1$, resulting in slightly more aggressive steps.

UCI Performance Profiles: We run on the following 20 binary classification datasets selected from the UCI repository: `blood`, `breast-cancer`, `chess-krvkp`, `congressional-voting`, `conn-bench-sonar`, `credit-approval`, `cylinder-bands`, `hill-valley`, `horse-colic`, `ilpd-indian-liver`, `ionosphere`, `magic`, `mammographic`, `musk-1`, `ozone`, `pima`, `tic-tac-toe`, `titanic`, `ringnorm`, `spambase`. We use the pre-processed datasets provided by Fernández-Delgado et al. (2014), although we do not use their splits since these are known to have test set contamination. To obtain 120 distinct problems, we also consider regularization parameters from the grid $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. For each problem, we estimate $f(w^*)$ by running SGD-M with teleportation to obtain a very accurate solution. For fairness, we estimate $f(w^*)$ separately for deterministic and stochastic problems; in the deterministic setting, we run SGD-M for 2500 iterations with teleportation every fifty iterations starting at $k = 5$. In the stochastic setting, we run SGD-M for 250 epochs with teleportation steps every 10 epochs starting at epoch 5. In both cases, we select step-sizes independently for each problem using the grid-search described above.

For the stochastic setting, we use batch-sizes of 64 and train for 100 epochs. The teleportation schedule is $\mathcal{T} = \{5, 15, 25, 35, 45, 55, 65, 75, 85, 95\}$ and we record metrics every five iterations. In the deterministic setting, we run for 500 iterations and use teleportation schedule $\mathcal{T} = \{5, 55, 105, 155, 205, 255, 305, 355\}$. We record metrics at every iteration.

We use different tolerances to generate our performance profiles. All profiles in the stochastic setting use $\tau = 0.05$, while profiles in the deterministic setting use $\tau = 0.15$ for logistic regression and $\tau = 0.1$ for neural networks. These tolerances were chosen so that the worst-performing method solved approximately 50% of problems. As discussed, we estimate $f(w^*)$ separately in the stochastic and deterministic settings using SGD with momentum and teleportation. Thus, the estimate of $f(w^*)$ is not the same between Figure 4 and Figure 10.

Initialization by Teleportation: All experimental details are as described for the other performance profiles except that the teleportation schedule is $\mathcal{T} = \{0\}$ for methods using teleportation.

Timed Performance Profiles: All experimental details are as described for the other performance profiles except that we run all methods for longer. We run deterministic methods for 1000 iterations with teleportation schedule $\mathcal{T} = \{5 + 50k : k \in \mathbb{N}, k < 20\}$ iterations. For stochastic methods, we run for 200 epochs with teleportation schedule $\mathcal{T} = \{5 + 10k : k \in \mathbb{N}, k < 20\}$.

Image Classification: We use a fixed strength of $\lambda = 10^{-2}$ for the weight decay regularization. All other settings are as described above. Depending on the experiment, we use either the soft-plus or ReLU activations. All models are two-layer networks with 500 hidden units. We plot the median and interquartile range out of three random restarts.

Effects of Regularization: The data for Figure 7 comes directly from the performance profiles in Figure 4. All results are shown for ReLU networks with two hidden layers of 100 neurons each.

Additional UCI Plots: The data for Figure 17 comes directly from the performance profile in Figure 8 and the data for Figure 18 comes from the performance profile in Figure 4. All neural network results are shown for regularization parameter $\lambda = 0.01$, while logistic regression results use $\lambda = 10^{-6}$.

Comparison with Newton: We use the open source implementation of Newton-CG in PyTorch by Feinman (2021). The number CG steps for each step of Newton-CG is fixed at 25. Similarly, for GD with teleportation we limit the number of iterations used by Algorithm 2 to be 25. This ensures both methods use the same number of Hessian-vector products. For Newton-CG-LS, we set the step-size using line-search on the strong Wolfe conditions. For methods with fixed step-sizes, we pick the step-size by grid-search on the training loss over the set

$$\{\{1000, 100, 10, 5, 2, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}\}.$$

[2] `https://github.com/AxelThevenot/Python_Benchmark_Test_Optimization_Function_Single_Objective`

We set the regularization parameter to 0 (unregularized) for both the convex and non-convex problems. The non-convex model is a three-layer ReLU network with 100 hidden units in each hidden layer.

C.3 Computational Details

The experiments in Figure 1 were run on a MacBook Pro (16 inch, 2019) with a 2.6 GHz 6-Core Intel i7 CPU and 16GB of memory. All other experiments were run on a Slurm cluster with several different node configurations. For experiments requiring accurate timing, we used nodes with Nvidia A100 GPUs with either 80GB or 40GB memory (their throughput is rated as the same) and Icelake CPUs. For experiments without accurate timing, we also allowed nodes with Nvidia H100-80GB GPUs, Nvidia V100-32GB, or V100-16GB GPUs, where nodes with the last two cards use Skylake CPUs instead of Icelake. All jobs were allocated a single GPU and 24GB of RAM. We thank the Scientific Computing Core, Flatiron Institute for support running our experiments.

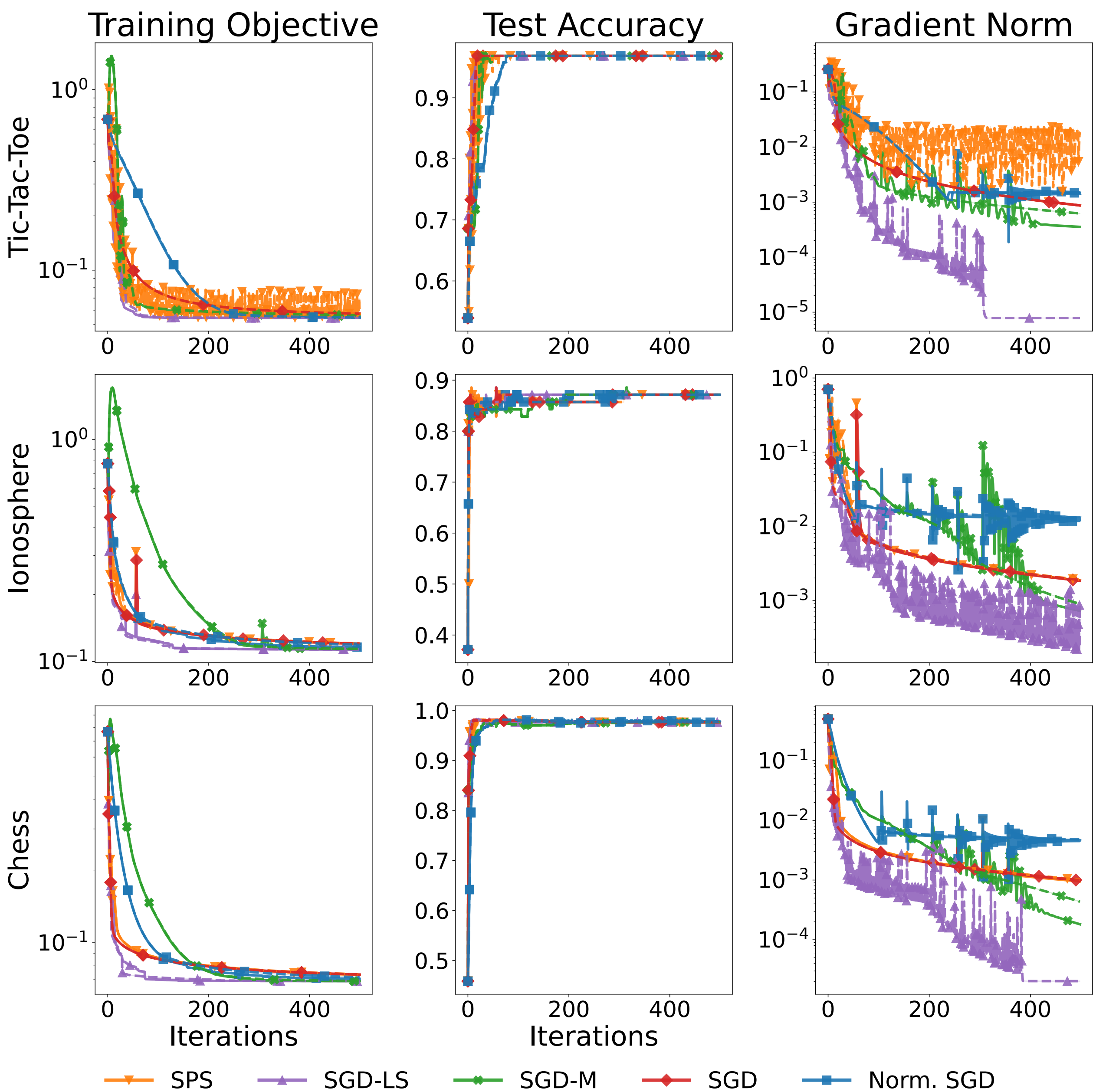

Figure 17: Convergence of deterministic gradient methods for training logistic regression models on three datasets from the UCI repository. Solid lines show methods with teleportation, while dashed lines show the same algorithms without teleportation. This figure reproduces exact training curves from Figure 8.

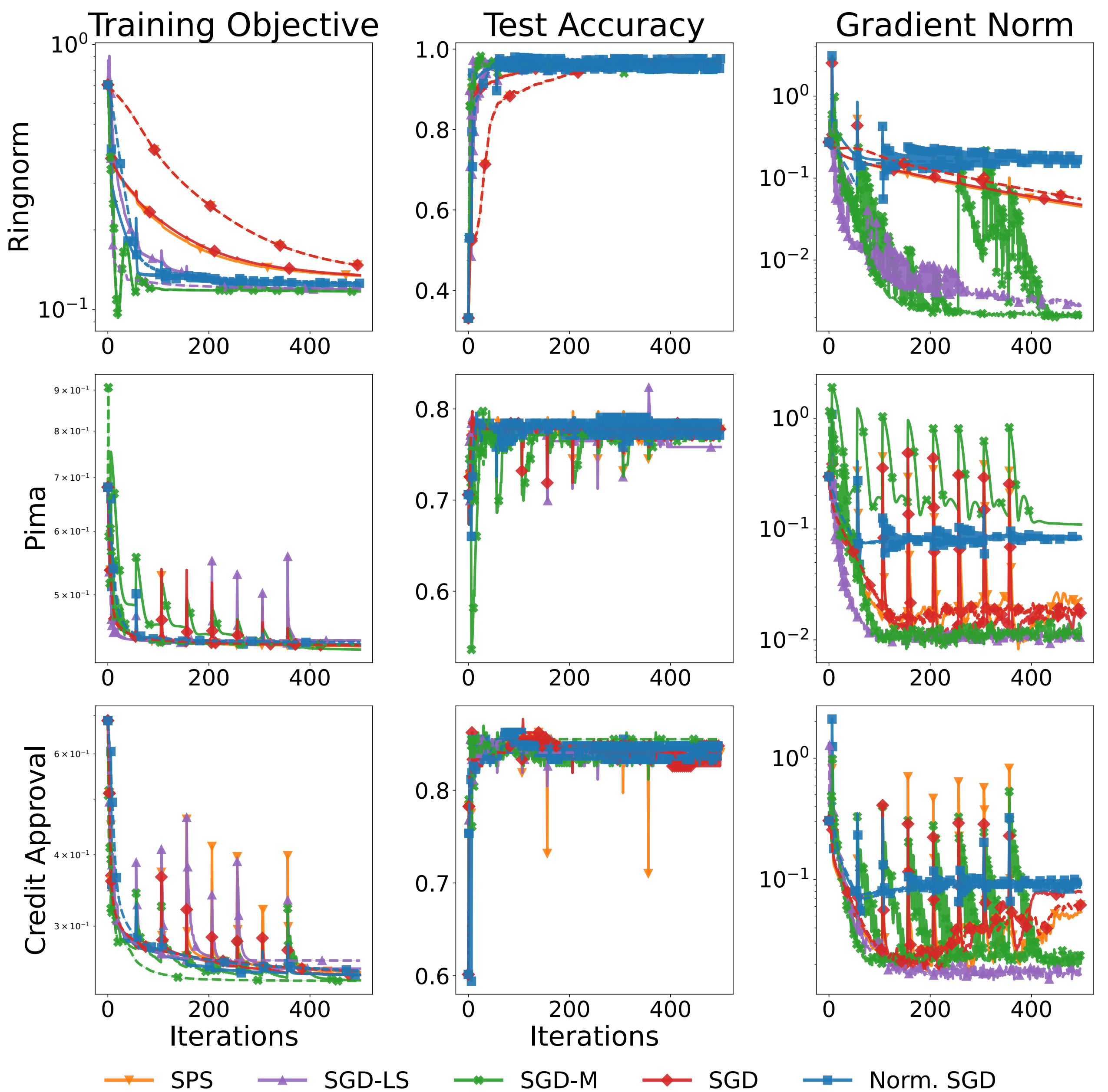

Figure 18: Convergence of deterministic gradient methods for three-layer ReLU networks with 100 units each on three datasets from the UCI repository. Solid lines show methods with teleportation, while dashed lines show the same algorithms without teleportation. This figure reproduces exact training curves from Figure 4.