Adaptive Quasi-Newton and Anderson Acceleration Framework with Explicit Global Convergence Rates

Anonymous Author(s)
Affiliation
Address
email

Abstract

Despite the impressive numerical performance of quasi-Newton and Anderson/nonlinear acceleration methods, their global convergence rates have remained elusive for over 50 years. This paper addresses this long-standing question by introducing a framework that derives novel and adaptive quasi-Newton or nonlinear/Anderson acceleration schemes. Under mild assumptions, the proposed iterative methods exhibit explicit, non-asymptotic convergence rates that blend those of gradient descent and Cubic Regularized Newton's method. Notably, these rates are achieved adaptively, as the method autonomously determines the optimal step size using a simple backtracking strategy. The proposed approach also includes an accelerated version that improves the convergence rate on convex functions. Numerical experiments demonstrate the efficiency of the proposed framework, even compared to a fine-tuned BFGS algorithm with line search.

1 Introduction

Consider the problem of finding the minimizer $x^\star$ of the unconstrained minimization problem

$$f(x^\star) = f^\star = \min_{x \in \mathbb{R}^d} f(x),$$

where d is the problem's dimension, and the function f has a Lipschitz continuous Hessian.

Assumption 1. *The function $f(x)$ has a Lipschitz continuous Hessian with a constant L,*

$$\forall \ y, z \in \mathbb{R}^d, \quad \|\nabla^2 f(z) - \nabla^2 f(y)\| \leq L\|z - y\|. \tag{1}$$

In this paper, $\|.\|$ stands for the maximal singular value of a matrix and for the ℓ_2 norm for a vector. Many twice-differentiable problems like logistic or least-squares regression satisfy Assumption 1.

The Lipschitz continuity of the Hessian is crucial when analyzing second-order algorithms, as it extends the concept of smoothness to the second order. The groundbreaking work by Nesterov et al. [46] has sparked a renewed interest in second-order methods, revealing the remarkable convergence rate improvement of Newton's method on problems satisfying Assumption 1 when augmented with cubic regularization. For instance, if the problem is also convex, accelerated gradient descent typically achieves $O(\frac{1}{t^2})$, while accelerated second-order methods achieve $O(\frac{1}{t^3})$. Recent advancements have further pushed the boundaries, achieving even faster convergence rates of up to $\mathcal{O}(\frac{1}{t^{7/2}})$ through the utilization of hybrid methods [43, 14] or direct acceleration of second-order methods [44, 27, 40].

Unfortunately, second-order methods may not always be feasible, particularly in high-dimensional problems common in machine learning. The limitation is that exact second-order methods require solving a linear system that involves the Hessian of the function f. This main limitation motivated alternative approaches that balance the efficiency of second-order methods and the scalability of first-order methods, such as *inexact/subspace/stochastic techniques*, *nonlinear/Anderson acceleration*, and *quasi-Newton* methods.

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

1.1 Contributions

Despite the impressive numerical performance of quasi-Newton methods and nonlinear acceleration schemes, there is currently no knowledge about their global explicit convergence rates. In fact, global convergence cannot be guaranteed without using either exact or Wolfe-line search techniques. This raises the following long-standing question **that has remained unanswered for over 50 years**:

What are the non-asymptotic global convergence rates of quasi-Newton and Anderson/nonlinear acceleration methods?

This paper provides a partial answer by introducing generic updates (see algorithms 1 to 3) that can be viewed as cubic-regularized quasi-Newton methods or regularized nonlinear acceleration schemes.

Under mild assumptions, the iterative methods constructed within the proposed framework (see algorithms 3 and 6) exhibit *explicit, global and non-asymptotic* convergence rates that interpolate the one of first order and second order methods (more details in appendix A):

- Convergence rate on non-convex problems (Theorem 4): $\min_i \|\nabla f(x_i)\| \leq O(t^{-\frac{2}{3}} + t^{-\frac{1}{3}})$,
- Convergence rate on (star-)convex problems (Theorems 5 and 6): $f(x_t) - f^\star \leq O(t^{-2} + t^{-1})$,
- Accelerated rate on convex problems (Theorem 7): $f(x_t) - f^\star \leq O(t^{-3} + t^{-2})$.

1.2 Related work

Inexact, subspace, and stochastic methods. Instead of explicitly computing the Hessian matrix and Newton's step, these methods compute an approximation using sampling [2], inexact Hessian computation [29, 19], or random subspaces [20, 31, 35]. By adopting a low-rank approximation for the Hessian, these approaches substantially reduce per-iteration costs without significantly compromising the convergence rate. The convergence speed in such cases often represents an interpolation between the rates observed in gradient descent methods and (cubic) Newton's method.

Nonlinear/Anderson acceleration. Nonlinear acceleration techniques, including Anderson acceleration [1], have a long standing history [3, 4, 28]. Driven by their promising empirical performance, they recently gained interest in their convergence analysis [64, 26, 63, 38, 69, 67, 72, 71, 56, 65, 66, 6, 60, 8, 57]. In essence, Anderson acceleration is an optimization technique that enhances convergence by extrapolating a sequence of iterates using a combination of previous gradients and corresponding iterates. Comprehensive reviews and analyses of these techniques can be found in notable sources such as [38, 7, 37, 36, 5, 17]. However, these methods do not generalize well outside quadratic minimization and their convergence rate can only be guaranteed asymptotically when using a line-search or regularization techniques [62, 68, 56].

Quasi-Newton methods. Quasi-Newton schemes are renowned for their exceptional efficiency in continuous optimization. These methods replace the exact Hessian matrix (or its inverse) in Newton's step with an approximation that is updated iteratively during the method's execution. The most widely used algorithms in this category include DFP [18, 25] and BFGS [61, 30, 24, 10, 9]. Most of the existing convergence results predominantly focus on the asymptotic super-linear rate of convergence [70, 32, 12, 11, 15, 22, 75, 73, 74]. However, recent research on quasi-Newton updates has unveiled explicit and non-asymptotic rates of convergence [50, 52, 51, 41, 42]. Nonetheless, these analyses suffer from several significant drawbacks, such as assuming an infinite memory size and/or requiring access to the Hessian matrix. These limitations fundamentally undermine the essence of quasi-Newton methods, which are typically designed to be Hessian-free and maintain low per-iteration cost through their low-memory requirement and low-rank structure.

Recently, Kamzolov et al. [39] introduced an adaptive regularization technique combined with cubic regularization, with global, explicit (accelerated) convergence rates for any quasi-Newton method. The method incorporates a backtracking line search on the secant inexactness inequality that introduces a quadratic regularization. However, this algorithm relies on prior knowledge of the Lipschitz constant specified in Assumption 1. Unfortunately, the paper does not provide an adaptive method to find jointly the Lipschitz constant as well, as it is *a priory* too costly to know which parameter to update. This aspect makes the method impractical in real-world scenarios.

Paper Organization **Section 2** introduces the proposed novel generic updates and some essential theoretical results. **Section 3** presents the convergence analysis of the iterative algorithm, which uses one of the proposed updates. **Section 4** is dedicated to the accelerated version of the proposed framework. **Section 5** presents examples of methods generated by the proposed framework.

2 Type-I and Type-II Step

This section first examines a remarkable property shared by quasi-Newton and Anderson acceleration: the sequence of iterates of these methods can be expressed as a combination of *directions* formed by previous iterates and the current gradient. Building upon this observation, section 2.1 investigates how to obtain second-order information without directly computing the Hessian of the function f by *approximating* the Hessian within the subspace formed by these directions. Subsequently, section 2.2 demonstrates how to utilize this approximation to establish an *upper bound* for the function f and its gradient norm $\|\nabla f(x)\|$. Minimizing these upper bounds, respectively, leads to a type-I and type-II method.

Motivation: what quasi-Newton and nonlinear acceleration schemes actually do? The BFGS update is a widely used quasi-Newton method for unconstrained optimization. It approximates the inverse Hessian matrix using updates based on previous gradients and iterates. The update reads

$$x_{t+1} = x_t - h_t H_t \nabla f(x_t), \quad H_t = H_{t-1}\left(I - \frac{g_t d_t^T}{g_t^T d_t}\right) + d_t\left(d_t^T \frac{d_t^T g_t + g_t^T H_{t-1} d_t}{(g_t^T d_t)^2} - \frac{g_t^T H_{t-1}}{g_t^T d_t}\right)$$

where H_t is the approximation of the inverse Hessian at iteration t, h_t is the step size, $d_t = x_t - x_{t-1}$ is the step direction, $g_t = \nabla f(x_t) - \nabla f(x_{t-1})$ is the gradient difference. After unfolding the equation, the BFGS update can be seen as a combination of the d_i's and $\nabla f(x_t)$,

$$x_{t+1} - x_t = H_0 P_0 \dots P_t \nabla f(x_t) + \textstyle\sum_{i=1}^{t} \alpha_i d_i, \tag{2}$$

where P_i are projection matrices in $\mathbb{R}^{d\times d}$ and α_i are coefficients. Similar reasoning can be applied to other quasi-Newton formulas (see appendix B for more details).

This observation aligns with the principles of Anderson acceleration methods. Considering the same vectors d_t and g_t, Anderson acceleration updates x_{t+1} as:

$$\alpha^\star = \min_\alpha \|\nabla f(x_t) + \textstyle\sum_{i=0}^{t-1} \alpha_i g_i\|, \quad x_{t+1} - x_t = \textstyle\sum_{i=0}^{t} \alpha_i^\star \left(d_i - h_t g_i\right),$$

where h_t is the relaxation parameter, which can be seen as the step size of the method. As all x_i's belong to the span of previous gradients, the update is similar to (2), see appendix B for more details. This is not surprising, as it has been shown that Anderson acceleration can be viewed as a quasi-Newton method [23]. Some studies have explored the relationship between these two classes of optimization techniques and established strong connections in terms of their algorithmic behavior [23, 76, 59, 13].

Hence, quasi-Newton algorithms and nonlinear/Anderson acceleration methods utilize previous directions d_i and the current gradient $\nabla f(x_t)$ in subsequent iterations. However, their convergence is guaranteed only if a line search is used, and their convergence speed is heavily dependent on H_0 (quasi-Newton) or h_t (Anderson acceleration) [49].

2.1 Error Bounds on the Hessian-Vector Product Approximation by a Difference of Gradients

Consider the following $d \times N$ matrices that represent the *algorithm's memory*,

$$Y = [y_1, \dots, y_N], \quad Z = [z_1, \dots, z_N], \quad D = Y - Z, \quad G = [\dots, \nabla f(y_i) - \nabla f(z_i), \dots]. \tag{3}$$

For example, to mimic quasi-Newton techniques, the matrices Y and Z can be defined such that,

$$D = [\dots, x_{t-i+1} - x_{t-i}, \dots], \quad G = [\dots, \nabla f(x_{t-i+1}) - \nabla f(x_{t-i}), \dots], \quad i = 1 \dots N.$$

Motivated by (2), this paper studies the following update, defined as a linear combination of the previous directions d_i,

$$x_+ - x = D\alpha \quad \text{where} \quad \alpha \in \mathbb{R}^N. \tag{4}$$

The objective is to determine the optimal coefficients α based on the information contained in the matrices defined in (3). Notably, the absence of the gradient in the update (4) distinguishes this

approach from (2), allowing for the development of an adaptive method that eliminates the need for an initial matrix H_0 (quasi-Newton methods) or a mixing parameter h_t (Anderson acceleration).

Under assumption (1), the following bounds hold for all $x, y, z, x_+ \in \mathbb{R}^d$ [46],

$$\|\nabla f(y) - \nabla f(z) - \nabla^2 f(z)(y-z)\| \leq \tfrac{L}{2}\|y-z\|^2, \tag{5}$$

$$\left|f(x_+) - f(x) - \nabla f(x)(x_+ - x) - \tfrac{1}{2}(x_+ - x)^T \nabla^2 f(x)(x_+ - x)\right| \leq \tfrac{L}{6}\|x_+ - x\|^3. \tag{6}$$

The accuracy of the estimation of the matrix $\nabla^2 f(x)$, depends on the *error vector* ε,

$$\varepsilon \overset{\text{def}}{=} [\varepsilon_1, \ldots, \varepsilon_N], \quad \text{and} \quad \varepsilon_i \overset{\text{def}}{=} \|d_i\| \left(\|d_i\| + 2\|z_i - x\|\right). \tag{7}$$

The following Theorem 1 explicitly bounds the error of approximating $\nabla^2 f(x)D$ by G.

Theorem 1. *Let the function f satisfy Assumption 1. Let x_+ be defined as in (4) and the matrices D, G be defined as in (3) and vector ε as in (7). Then, for all $w \in \mathbb{R}^d$ and $\alpha \in \mathbb{R}^N$,*

$$-\tfrac{L\|w\|}{2}\textstyle\sum_{i=1}^N |\alpha_i|\varepsilon_i \leq w^T(\nabla^2 f(x)D - G)\alpha \leq \tfrac{L\|w\|}{2}\sum_{i=1}^N |\alpha_i|\varepsilon_i, \tag{8}$$

$$\|w^T(\nabla^2 f(x)D - G)\| \leq \tfrac{L\|w\|}{2}\|\varepsilon\|. \tag{9}$$

Proof sketch and interpretation. The theorem states that the Hessian-vector product $\nabla^2 f(x)(y-z)$ can be approximated by the difference of gradients $\nabla f(y) - \nabla f(z)$, providing a cost-effective approach to estimate $\nabla^2 f$ without computing it. This property is the basis of quasi-Newton methods. The detailed proof can be found in appendix F. The main idea of the proof is as follows. From (5) with $y = y_i$ and $z = z_i$, writing $d_i = y_i - z_i$, and Assumption 1,

$$\|\nabla f(y_i) - \nabla f(z_i) - \nabla^2 f(x)(y_i - z_i)\| \leq \frac{L}{2}\|d_i\|^2 + \|\nabla^2 f(x) - \nabla^2 f(z)\|\|d_i\| \leq \frac{L}{2}\varepsilon_i.$$

The *first* term in ε_i bounds the error of (5), while the *second* comes from the distance between (5) and the current point x where the Hessian is estimated. Then, it suffices to combine the inequalities with coefficients α to obtain Theorem 1.

2.2 Type I and Type II Inequalities and Methods

In the literature, Type-I methods often refer to algorithms that aim to minimize the function value $f(x)$, while type-II methods minimize the gradient norm $\|\nabla f(x)\|$ [23, 76, 13]. Applying the bounds (6) and (5) to the update in (4) yields the following Type-I and Type-II upper bounds, respectively.

Theorem 2. *Let the function f satisfy Assumption 1. Let x_+ be defined as in (4), the matrices D, G be defined as in (3) and ε be defined as in (7). Then, for all $\alpha \in \mathbb{R}^N$,*

$$f(x_+) \leq f(x) + \nabla f(x)^T D\alpha + \tfrac{\alpha^T H \alpha}{2} + \tfrac{L\|D\alpha\|^3}{6}, \quad H \overset{def}{=} \tfrac{G^T D + D^T G + \mathrm{I}L\|D\|\|\varepsilon\|}{2} \tag{10}$$

$$\|\nabla f(x_+)\| \leq \|\nabla f(x) + G\alpha\| + \tfrac{L}{2}\Big(\textstyle\sum_{i=1}^N |\alpha_i|\varepsilon_i + \|D\alpha\|^2\Big), \tag{11}$$

The proof can be found in appendix F. Minimizing eqs. (10) and (11) leads to algorithms 1 and 2, respectively, whose constant L is replaced by a parameter M, found by backtracking line-search. A study of the (strong) link between these proposed algorithms and nonlinear/Anderson acceleration and quasi-Newton methods can be found in appendix B.

Solving the sub-problems In algorithms 1 and 2, the coefficients α are computed by solving a minimization sub-problem in $O(N^3 + Nd)$ (see appendix C for more details). Usually, N is rather small (e.g. between 5 and 100); hence solving the subproblem is negligible compared to computing a new gradient $\nabla f(x)$. Here is the summary:

- **In algorithm 1**, the subproblem can be solved easily by a convex problem in two variables, which involves an eigenvalue decomposition of the matrix $H \in \mathbb{R}^{N\times N}$ [46].
- **In algorithm 2**, the subproblem can be cast into a linear-quadratic problem of $O(N)$ variables and constraints that can be solved efficiently with SDP solvers (e.g., `SDPT3`).

Algorithm 1 Type-I Subroutine with Backtracking Line-search

Require: First-order oracle for f, matrices $G,\ D$, vector ε, iterate x, initial smoothness M_0.

1: Initialize $M \leftarrow \frac{M_0}{2}$
2: **do**
3: $\quad M \leftarrow 2M \quad$ and $\quad H \leftarrow \frac{G^T D + D^T G}{2} + \mathrm{I}_N \frac{M\|D\|\|\varepsilon\|}{2}$
4: $\quad \alpha^\star \leftarrow \arg\min_\alpha f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}\alpha^T H \alpha + \frac{M\|D\alpha\|^3}{6}$
5: $\quad x_+ \leftarrow x + D\alpha$
6: **while** $f(x_+) \geq f(x) + \nabla f(x)^T D\alpha^\star + \frac{1}{2}[\alpha^\star]^T H \alpha^\star + \frac{M\|D\alpha^\star\|^3}{6}$
7: **return** x_+, M

Algorithm 2 Type-II Subroutine with Backtracking Line-search

Same as algorithm 1, but minimize and check the upper bound (11) instead of (10) on lines 4 and 6.

3 Iterative Type-I Method: Framework and Rates of Convergences

The rest of the paper analyzes the convergence rate of methods that use algorithm 1 as a subroutine; see algorithm 3. The analysis of methods that uses algorithm 2 is left for future work.

3.1 Main Assumptions and Design Requirements

This section lists the important assumptions on the function f. Some subsequent results require an upper bound on the radius of the sub-level set of f at $f(x_0)$.

Assumption 2. *The radius of the sub-level set* $\{x : f(x) \leq f(x_0)\}$ *is bounded by* $\mathrm{R} < \infty$.

To ensure the convergence toward $f(x^\star)$, some results require f to be star-convex or convex.

Assumption 3. *The function f is star convex if, for all $x \in \mathbb{R}^d$ and $\forall \tau \in [0,1]$,*

$$f((1-\tau)x + \tau x^\star) \leq (1-\tau)f(x) + \tau f(x^\star).$$

Assumption 4. *The function f is convex if, for all $y,\ z \in \mathbb{R}^d$, $f(y) \geq f(z) + \nabla f(z)(y-z)$.*

The matrices $Y,\ Z,\ D$ must meet some conditions listed below as "requirements" (see section 5 for details). All convergence results rely on *one* of these conditions on the projector onto $\mathbf{span}(D)$,

$$P_t \overset{\text{def}}{=} D_t (D_t^T D_t)^{-1} D_t^T. \tag{12}$$

Requirement 1a. *For all t, the projector P_t of the stochastic matrix D_t satisfies $\mathbb{E}[P_t] = \frac{N}{d}\boldsymbol{I}$.*

Requirement 1b. *For all t, the projector P_t satisfies $P_t \nabla f(x_t) = \nabla f(x_t)$.*

The first condition guarantees that, in expectation, the matrix D_t spans partially the gradient $\nabla f(x_t)$, since $\mathbb{E}[P_t \nabla f(x_t)] = \frac{N}{d}\nabla f(x_t)$. The second condition simply requires the possibility to move towards the current gradient when taking the step $x + D\alpha$. This condition resonates with the idea presented in (2), where the step $x_+ - x$ combines previous directions and the current gradient $\nabla f(x_t)$.

In addition, it is required that the norm of $\|\varepsilon\|$ does not grow too quickly, hence the next assumption.

Requirement 2. *For all t, the relative error $\frac{\|\varepsilon_t\|}{\|D_t\|}$ is bounded by δ.*

The Requirement 2 is also non-restrictive, as it simply prevents taking secant equations at $y_i - z_i$ and $z_i - x_i$ too far apart. Most of the time, δ satisfies $\delta \leq O(R)$.

Finally, the condition number of the matrix D also has to be bounded.

Requirement 3. *For all t, the matrix D_t is full-column rank, which implies that $D_t^T D_t$ is invertible. In addition, its condition number $\kappa_{D_t} \overset{def}{=} \sqrt{\|D_t^T D_t\|\|(D_t^T D_t)^{-1}\|}$ is bounded by κ.*

The condition on the rank of D is not overly restrictive. In most practical scenarios, this condition is typically satisfied without issue. However, the second condition might be hard to meet, but section 5 studies strategies that prevent κ_D from exploding by taking orthogonal directions or pruning D.

Algorithm 3 Generic Iterative Type-I Methods

Require: First-order oracle f, initial iterate and smoothness $x_0,\ M_0$, number of iterations T.
  for $t = 0, \ldots, T-1$ **do**
    Update G_t, D_t, ε_t (see section 5).
    $x_{t+1}, M_{t+1} \leftarrow$ `[algorithm 1]`$(f, G_t, D_t, \varepsilon_t, x_t, (M_t/2))$
  end for
  return x_T

3.2 Rates of Convergence

When f satisfies Assumption 1, algorithm 3 ensures a minimal function decrease at each step.

Theorem 3. *Let f satisfy Assumption 1. Then, at each iteration $t \geq 0$, algorithm 3 achieves*

$$f(x_{t+1}) \leq f(x_t) - \tfrac{M_{t+1}}{12}\|x_{t+1} - x_t\|^3, \quad M_{t+1} < \max\left\{2L\ ;\ \tfrac{M_0}{2^t}\right\}. \tag{13}$$

Under some mild assumptions, algorithm 3 converges to a critical point for non-convex functions.

Theorem 4. *Let f satisfy Assumption 1, and assume that f is bounded below by f^*. Let Requirements 1b to 3 hold, and $M_t \geq M_{\min}$. Then, algorithm 3 starting at x_0 with M_0 achieves*

$$\min_{i=1,\ldots,t}\|\nabla f(x_i)\| \leq \max\left\{\frac{3L}{t^{2/3}}\left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{2/3}\ ;\ \left(\frac{C_1}{t^{1/3}}\right)\left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{1/3}\right\},$$

where $C_1 = \delta L\left(\frac{\kappa+2\kappa^2}{2}\right) + \max_{i\in[0,t]}\|(I-P_i)\nabla^2 f(x_i)P_i\|$.

Going further, algorithm 3 converges to an optimum when the function is star-convex.

Theorem 5. *Assume f satisfy Assumptions 1 to 3. Let Requirements 1b to 3 hold. Then, algorithm 3 starting at x_0 with M_0 achieves, for $t \geq 1$,*

$$(f(x_t) - f^\star) \leq 6\frac{f(x_t)-f^\star}{t(t+1)(t+2)} + \frac{1}{(t+1)(t+2)}\frac{L(3R)^3}{2} + \frac{1}{t+2}\frac{C_2(3R)^2}{4},$$

where $C_2 \overset{def}{=} \delta L\frac{\kappa+2\kappa^2}{2} + \max_{i\in[0,t]}\|\nabla^2 f(x_i) - P_i\nabla^2 f(x_i)P_i\|$.

Finally, the next theorem shows that when algorithm 3 uses a stochastic D that satisfies Requirement 1a, then $f(x_t)$ also converges in expectation to $f(x^\star)$ when f is convex.

Theorem 6. *Assume f satisfy Assumptions 1, 2 and 4. Let Requirements 1a, 2 and 3 hold. Then, in expectation over the matrices D_i, algorithm 3 starting at x_0 with M_0 achieves, for $t \geq 1$,*

$$\mathbb{E}_{D_t}[f(x_t) - f^\star] \leq \frac{1}{1+\frac{1}{4}\left[\frac{N}{d}t\right]^3}(f(x_0)-f^\star) + \frac{1}{\left[\frac{N}{d}t\right]^2}\frac{L(3R)^3}{2} + \frac{1}{\left[\frac{N}{d}t\right]}\frac{C_3(3R)^2}{2},$$

where $C_3 \overset{def}{=} \delta L\frac{\kappa+2\kappa^2}{2} + \frac{(d-N)}{d}\max_{i\in[0,t]}\|\nabla^2 f(x_i)\|$.

Interpretation The rates presented in Theorems 4 to 6 combine the ones of cubic regularized Newton's method and gradient descent (or coordinate descent, as in Theorem 6) for functions with Lipschitz-continuous Hessian. As C_1, C_2, and C_3 decrease, the rates approach those of cubic Newton.

The constants C_1, C_2, and C_3 quantify the error of approximating $D\nabla^2 f(x)D$ by H in (10) into two terms. The first represents the error made by approximating $\nabla^2 f(x)D$ by G, while the second describes the low-rank approximation of $\nabla^2 f(x)$ in the subspace spanned by the columns of D. The approximation is more explicit in C_3, where increasing N reduces the constant up to $N = d$.

To retrieve the convergence rate of Newton's method with cubic regularization, the approximation needs to satisfy three properties: **1)** the points contained in Y_t and Z_t must be close to each other, and to x_t to reduce δ and $\|\varepsilon\|$; **2)** the condition number of D should be close to 1 to reduce κ; **3)** D should span a maximum dimension in $\mathbb{R}^d$ to improve the approximation of $\nabla^2 f(x)$ by $P\nabla^2 f(x)P$.

For example, $Z_t = x_t\mathbf{1}_N^T$, $D_t = h\mathrm{I}_N$ with h small, and $Y_t = Z_t + D_t$ achieve these conditions. This (naive) strategy estimates all directional second derivatives with a finite difference for all coordinates and is equivalent to performing a Newton's step in terms of complexity.

Algorithm 4 Type-I subroutine with backtracking for the accelerated method

Require: First-order oracle f, matrices $G,\ D$, vector ε, iterate x, smoothness M_0, minimal norm Δ

Initialize $M \leftarrow \frac{M_0}{2}$, $\gamma \leftarrow \frac{1}{4}\frac{\|\varepsilon\|}{\|D\|}\left(1+\kappa_D^2\right)$, `ExitFlag` $\leftarrow$ `False`

while `ExitFlag` is False **do**

Update M and $H \leftarrow \frac{G^T D + D^T G}{2} + \mathrm{I}_N \frac{M\|D\|\|\varepsilon\|}{2}$

$\alpha^* \leftarrow \arg\min_\alpha f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}\alpha^T H\alpha + \frac{M\|D\alpha\|^3}{6}$

$x_+ \leftarrow x + D\alpha$

If $-\nabla f(x_+)^T D\alpha \geq \frac{\|\nabla f(x_+)\|^{3/2}}{\sqrt{\frac{3M}{4}}}$ **and** $\|D\alpha\| \geq \Delta$ **then** `ExitFlag` $\leftarrow$ `LargeStep`

If $-f(x_+)^T D\alpha \geq \frac{\|\nabla f(x_+)\|^2}{M\left(\gamma + \frac{\|D\alpha\|}{2}\right)}$ **then** `ExitFlag` $\leftarrow$ `SmallStep`

end while

return x_+, α, M, γ, `ExitFlag`

Algorithm 5 Adaptive Accelerated Type-I Algorithm (Sketch, see appendix D for the full version)

Require: First-order oracle f, initial iterate and smoothness $x_0,\ M_0$, number of iterations T.

Initialize G_0, D_0, ε_0, $\lambda_0^{(1)}$, $\lambda_0^{(2)}$, Δ, x_1, M_1, $(M_0)_1$.

for $t = 1, \ldots, T-1$ **do**

Update G_t, D_t, ε_t.

do

Compute $v_t \leftarrow \arg\min \Phi_t$, set $y_t = \frac{t}{t+3}x_t + \frac{3}{t+3}v_t$, and update $(M_0)_t$

$\{x_{t+1}, \texttt{ExitFlag}\} \leftarrow$ `[algorithm 4]`$(f, G_t, D_t, \varepsilon_t, y_t, (M_0)_t, \Delta)$

if $\Phi_{t+1}(v_{t+1}) \leq f(x_{t+1})$ **then** `%% Parameters adjustment if needed`

`ValidBound` $\leftarrow$ `False`

if `ExitFlag` is `SmallStep` **then** $\lambda_t^{(1)} \leftarrow 2\lambda_t^{(1)}$, **otherwise** $\lambda_t^{(2)} \leftarrow 2\lambda_t^{(2)}$

else

`ValidBound` $\leftarrow$ `True` `%% Successful iteration`

end if

while `ValidBound is False`

end for

return x_T

4 Accelerated Algorithm for Convex Functions

This section introduces algorithm 5, an accelerated variant of algorithm 3 for convex functions, designed using the estimate sequence technique from [44]. It consists in iteratively building a function $\Phi_t(x)$, a regularized lower bound on f, that reads

$$\Phi_t(x) = \frac{1}{\sum_{i=0}^t b_i}\left(\sum_{i=0}^t b_i\left(f(x_i) + \nabla f(x_i)(x - x_i)\right) + \lambda_t^{(1)}\frac{\|x-x_0\|^2}{2} + \lambda_t^{(2)}\frac{\|x-x_0\|^3}{6}\right),$$

where $\lambda_t^{(1,2)}$ are non-decreasing. The key aspects of acceleration are as follows (see section 4 for more details): **1)** The accelerated algorithm makes a step at a linear combination between v_t, the optimum of Φ_t, and the previous iterate x_t. **2)** It uses a modified version of algorithm 1, see algorithm 4. **3)** Under some conditions, the step size can be considered as "large", i.e., similar to a cubic-Newton step. The $\Delta > 0$ ensures the step is sufficiently large to ensure theoretical convergence - but setting $\Delta = 0$ does not seem to impact the numerical convergence. The presence of both small and large steps is crucial to obtain the theoretical rate of convergence.

Theorem 7. *Assume f satisfy Assumptions 1, 2 and 4. Let Requirements 1b to 3 hold. Then, algorithm 5 starting at x_0 with M_0 achieves, for all $\Delta > 0$ and for $t \geq 1$,*

$$f(x_t) - f^\star \leq \frac{(M_0)_{\max}^2}{L}\left(\frac{3R}{t+3}\right)^2 + \frac{4(M_0)_{\max}}{3\sqrt{3}}\max\left\{1\ ;\ \tfrac{2}{\Delta}\right\}\left(\frac{3R}{t+3}\right)^3 + \frac{\frac{\tilde\lambda^{(1)}R^2}{2} + \frac{\tilde\lambda^{(2)}R^3}{6}}{(t+1)^3}.$$

where $\tilde\lambda^{(1)} = 0.5\cdot\delta\left(L\kappa + M_1\kappa^2\right) + \|\nabla f(x_0) - P_0\nabla f(x_0)P_0\|, \qquad \tilde\lambda^{(2)} = M_1 + L,$

$$(M_0)_{\max} = \tfrac{L}{2}(2\Delta + (2\kappa^2+\kappa)\delta) + (2\sqrt{3}-1)\max_{0\leq i\leq t}\|(I-P_i)\nabla^2 f(x_i)P_i\|.$$

Interpretation The interpretation is similar to the one from Section 3. Ignoring $\tilde{\lambda}^{(1,2)}$, the rate of Theorem 7 combines the one of accelerated gradient and accelerated cubic Newton [45, 44]. The constant M_0 blends the Lipschitz constant of the Hessian L with its approximation errors $(2\kappa^2 + \kappa)\delta$ and $\|(I - P)\nabla^2 f(x)\|$. The better the Hessian is approximated, the smaller the constant.

5 Some update strategies for matrices Y, Z, D, G

The framework presented in this paper is characterized by its generality, requiring only minimal assumptions on the matrix D and vector ε. This section explores different strategies for updating the matrices from (3), which can be classified into two categories: *online* and *batch techniques*.

Recommended method. Among all the methods presented in this section, the most promising technique seems to be the *Orthogonal Forward Estimates Only*, as it ensures that the condition number $\kappa_D = 1$ and the norm of the error vector $\|\varepsilon\|$ is small.

5.1 Online Techniques

The online technique updates the matrix D while algorithms 3 and 5 are running. To achieve Requirement 1b, the method employs either a steepest or orthogonal forward estimate, defined as

$$x_{t+\frac{1}{2}} = x_t - h\nabla f(x_t) \quad \textit{(steepest)} \quad \text{or} \quad x_{t+\frac{1}{2}} = x_t - h(I - P_{t-1})\frac{\nabla f(x_t)}{\|\nabla f(x_t)\|} \quad \textit{(orthogonal)}.$$

Then, it include $x_{t+\frac{1}{2}} - x_t$ in the matrix D_t. The projector P_{t-1} is defined in (12), and parameter h can be a fixed small value (e.g., $h = 10^{-9}$). This section investigates three different strategies for storing past information: *Iterates only*, *Forward Estimates Only*, and *Greedy*, listed below.

$$Y_t = [x_{t+\frac{1}{2}}, x_t, x_{t-1}, \ldots, x_{t-N+1}], \quad Z_t = [x_t, x_{t-1}, \ldots, x_{t-N}] \qquad \text{(Iterates only)}$$

$$Y_t = [x_{t+\frac{1}{2}}, x_{t-\frac{1}{2}}, \ldots, x_{t-N+\frac{1}{2}}], \quad Z_t = [x_t, x_{t-1}, \ldots, x_{t-N}] \qquad \text{(Forward Estimates Only)}$$

$$Y_t = [x_{t+\frac{1}{2}}, x_t, x_{t-\frac{1}{2}}, \ldots, x_{t-\frac{N+1}{2}}], \quad Z_t = [x_t, x_{t-\frac{1}{2}}, \ldots, x_{t-\frac{N}{2}}] \qquad \text{(Greedy)}$$

Iterates only: In the case of quasi-Newton updates and Nonlinear/Anderson acceleration, the iterates are constructed using the equation $x_{t+1} - x_t \in \nabla f(x_t) + \textbf{span}\{x_{t-i+1} - x_{t-i}\}_{i=1\ldots N}$. The update draws inspiration from this observation. However, it does not provide control over the condition number of D_t or the norm $\|\varepsilon\|$. To address this, one can either accept a potentially high condition number or remove the oldest points in D and G until the condition number is bounded (e.g., $\kappa = 10^9$).

Forward Estimates Only: This method provides more control over the iterates added to Y and Z. When using the *orthogonal* technique to compute $x_{i+\frac{1}{2}}$ reduces the constants in Theorems 4, 5 and 7: the condition number of D is equal to 1 as $D^T D = h^2 I$, and the norm of ε is small ($\|\varepsilon\| \leq O(h)$).

Greedy: The greedy approach involves storing both the iterates and the forward approximations. It shares the same drawback as the *Iterates only* strategy but retains at least the most recent information about the Hessian-vector product approximation, thereby reducing the $\|z_i - x_i\|$ term in ε (7).

5.2 Batch Techniques

Instead of making individual updates, an alternative approach is to compute them collectively, centered on x_t. This technique generates a matrix D_t consisting of N orthogonal directions $d_1, \cdots, d_N$ of norm h. The corresponding Y_t, Z_t, G_t matrices are then defined as follows:

$$Y_t = [x_t + d_1, \ldots, x_t + d_n], \quad Z_t = [x_t, \ldots, x_t], \quad G_t = [\ldots, \nabla f(x_t + d_i) - \nabla f(x_t), \ldots].$$

This section explores two batch techniques that generate orthogonal directions: *Orthogonalization* and *Random Subspace*. Both lead to $\delta = 3h$ and $\kappa = 1$ in Requirements 2 and 3. However, they require N additional gradient computations at each iteration (instead of one for the online techniques). For clarity, in the experiments, only the Greedy version is considered.

Orthogonalization: This technique involves using any online technique discussed in the previous section and storing the directions in a matrix $\tilde{D}_t$. Then, it constructs the matrices D_t by performing an orthogonalization procedure on $\tilde{D}_t$, such as the QR algorithm. This approach provides Hessian estimates in relevant directions, which can be more beneficial than random ones.

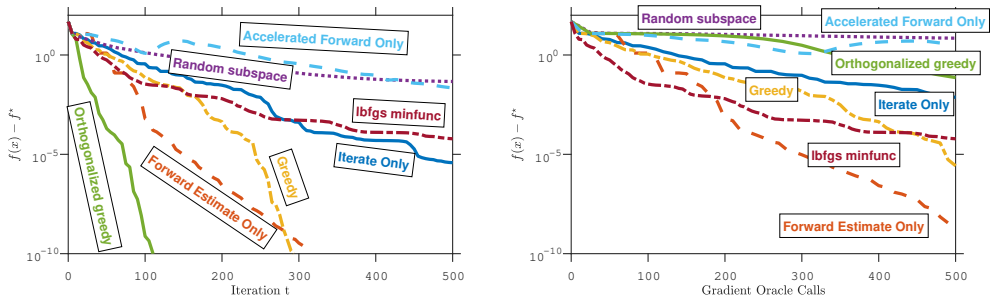

Figure 1: Comparison between the type-1 methods proposed in this paper and the optimized implementation of ℓ-BFGS from `minFunc` [53] with default parameters, except for the memory size. All methods use a memory size of $N = 25$.

Random Subspace: Inspired by [35], this technique randomly generates D_t at each iteration by either taking D_t to be N random (rescaled) canonical vectors or by using the Q matrix from the QR decomposition of a random $N \times D$ matrix. This ensures that D_t satisfies Requirement 1a. For clarity, in the experiments, only the QR version is considered.

6 Numerical Experiments

This section compares the methods generated by this paper's framework to the fine-tuned ℓ-BFGS algorithm from `minFunc` [53]. More experiments are conducted in appendix E. The tested methods are the Type-I iterative algorithms (algorithm 3 with the techniques from section 5). The step size of the forward estimation was set to $h = 10^{-9}$, and the condition number κ_{D_t} is maintained below $\kappa = 10^9$ with the iterates only and Greedy techniques. The accelerated algorithm 6 is used only with the *Forward Estimates Only* technique. The compared methods are evaluated on a logistic regression problem with no regularization on the Madelon UCI dataset [33]. The results are shown in fig. 1.

Regarding the number of iterations, the greedy orthogonalized version outperforms the others due to the orthogonality of directions (resulting in a condition number of one) and the meaningfulness of previous gradients/iterates. However, in terms of gradient oracle calls, the recommended method, *orthogonal forward iterates only*, achieves the best performance by striking a balance between the cost per iteration (only two gradients per iteration) and efficiency (small and orthogonal directions, reducing theoretical constants). Surprisingly, the accelerated method's performance is suboptimal, possibly because it tightens the theoretical analysis, diminishing its inherent adaptivity.

7 Conclusion, Limitation, and Future work

This paper introduces a generic framework for developing novel quasi-Newton and Anderson/Nonlinear acceleration schemes, offering a global convergence rate in various scenarios, including accelerated convergence on convex functions, with minimal assumptions and design requirements.

One limitation of the current approach is requiring an additional gradient step for the *forward estimate*, as discussed in Section 5. However, this forward estimate is crucial in enabling the algorithm's adaptivity, eliminating the need to initialize a matrix H_0 (quasi-Newton) or employ a mixing parameter h_0 (Anderson acceleration).

In future research, although unsuitable for large-scale problems, the method presented in this paper can achieve super-linear convergence rates, as with infinite memory, they would be as fast as cubic Newton methods. Utilizing the average-case analysis framework from existing literature, such as [48, 58, 21, 16, 47], could also improve the constants in Theorems 4 and 5 to match those in Theorem 6. Furthermore, exploring convergence rates for type-2 methods, which are believed to be effective for variational inequalities, is a worthwhile direction.

Ultimately, the results presented in this paper open new avenues for researchs. It may also provide a potential foundation for investigating additional properties of existing quasi-Newton methods and may even lead to the discovery of convergence rates for an adaptive, cubic-regularized BFGS variant.

References

[1] Donald G Anderson. "Iterative procedures for nonlinear integral equations". In: *Journal of the ACM (JACM)* 12.4 (1965), pp. 547–560.

[2] Kimon Antonakopoulos, Ali Kavis, and Volkan Cevher. "Extra-Newton: A First Approach to Noise-Adaptive Accelerated Second-Order Methods". In: *arXiv preprint arXiv:2211.01832* (2022).

[3] Claude Brezinski. "Application de l'ε-algorithme à la résolution des systèmes non linéaires". In: *Comptes Rendus de l'Académie des Sciences de Paris* 271.A (1970), pp. 1174–1177.

[4] Claude Brezinski. "Sur un algorithme de résolution des systèmes non linéaires". In: *Comptes Rendus de l'Académie des Sciences de Paris* 272.A (1971), pp. 145–148.

[5] Claude Brezinski and Michela Redivo–Zaglia. "The genesis and early developments of Aitken's process, Shanks' transformation, the ε–algorithm, and related fixed point methods". In: *Numerical Algorithms* 80.1 (2019), pp. 11–133.

[6] Claude Brezinski, Michela Redivo-Zaglia, and Yousef Saad. "Shanks sequence transformations and Anderson acceleration". In: *SIAM Review* 60.3 (2018), pp. 646–669.

[7] Claude Brezinski and M Redivo Zaglia. *Extrapolation methods: theory and practice*. Elsevier, 1991.

[8] Claude Brezinski et al. "Shanks and Anderson-type acceleration techniques for systems of nonlinear equations". In: *arXiv:2007.05716* (2020).

[9] Charles G Broyden. "The convergence of a class of double-rank minimization algorithms: 2. The new algorithm". In: *IMA journal of applied mathematics* 6.3 (1970), pp. 222–231.

[10] Charles George Broyden. "The convergence of a class of double-rank minimization algorithms 1. general considerations". In: *IMA Journal of Applied Mathematics* 6.1 (1970), pp. 76–90.

[11] Richard H Byrd and Jorge Nocedal. "A tool for the analysis of quasi-Newton methods with application to unconstrained minimization". In: *SIAM Journal on Numerical Analysis* 26.3 (1989), pp. 727–739.

[12] Richard H Byrd, Jorge Nocedal, and Ya-Xiang Yuan. "Global convergence of a cass of quasi-Newton methods on convex problems". In: *SIAM Journal on Numerical Analysis* 24.5 (1987), pp. 1171–1190.

[13] Marco Canini and Peter Richtárik. "Direct nonlinear acceleration". In: *Operational Research* 2192 (2022), p. 4406.

[14] Yair Carmon et al. "Recapp: Crafting a more efficient catalyst for convex optimization". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 2658–2685.

[15] Andrew R Conn, Nicholas IM Gould, and Ph L Toint. "Convergence of quasi-Newton matrices generated by the symmetric rank one update". In: *Mathematical programming* 50.1-3 (1991), pp. 177–195.

[16] Leonardo Cunha et al. "Only tails matter: Average-Case Universality and Robustness in the Convex Regime". In: 2022.

[17] Alexandre d'Aspremont, Damien Scieur, Adrien Taylor, et al. "Acceleration methods". In: *Foundations and Trends® in Optimization* 5.1-2 (2021), pp. 1–245.

[18] William C Davidon. "Variable metric method for minimization". In: *SIAM Journal on Optimization* 1.1 (1991), pp. 1–17.

[19] Nikita Doikov, El Mahdi Chayti, and Martin Jaggi. "Second-order optimization with lazy Hessians". In: *arXiv preprint arXiv:2212.00781* (2022).

[20] Nikita Doikov, Peter Richtárik, et al. "Randomized block cubic Newton method". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 1290–1298.

[21] Carles Domingo-Enrich, Fabian Pedregosa, and Damien Scieur. "Average-case acceleration for bilinear games and normal matrices". In: *arXiv preprint arXiv:2010.02076* (2020).

[22] John R Engels and Hector J Martinez. "Local and superlinear convergence for partially known quasi-Newton methods". In: *SIAM Journal on Optimization* 1.1 (1991), pp. 42–56.

[23] Haw-Ren Fang and Yousef Saad. "Two classes of multisecant methods for nonlinear acceleration". In: *Numerical Linear Algebra with Applications* 16.3 (2009), pp. 197–221.

[24] Roger Fletcher. "A new approach to variable metric algorithms". In: *The computer journal* 13.3 (1970), pp. 317–322.

[25] Roger Fletcher and Michael JD Powell. "A rapidly convergent descent method for minimization". In: *The computer journal* 6.2 (1963), pp. 163–168.

[26] William F Ford and Avram Sidi. "Recursive algorithms for vector extrapolation methods". In: *Applied numerical mathematics* 4.6 (1988), pp. 477–489.

[27] Alexander Gasnikov et al. "Near optimal methods for minimizing convex functions with lipschitz p-th derivatives". In: *Conference on Learning Theory*. PMLR. 2019, pp. 1392–1393.

[28] Eckart Gekeler. "On the solution of systems of equations by the epsilon algorithm of Wynn". In: *Mathematics of Computation* 26.118 (1972), pp. 427–436.

[29] Saeed Ghadimi, Han Liu, and Tong Zhang. "Second-order methods with cubic regularization under inexact information". In: *arXiv preprint arXiv:1710.05782* (2017).

[30] Donald Goldfarb. "A family of variable-metric methods derived by variational means". In: *Mathematics of computation* 24.109 (1970), pp. 23–26.

[31] Robert Gower et al. "Rsn: Randomized subspace newton". In: *Advances in Neural Information Processing Systems* 32 (2019).

[32] Andreas Griewank and Ph L Toint. "Local convergence analysis for partitioned quasi-Newton updates". In: *Numerische Mathematik* 39.3 (1982), pp. 429–448.

[33] Isabelle Guyon. "Design of experiments of the NIPS 2003 variable selection benchmark". In: *NIPS 2003 workshop on feature extraction and feature selection*. Vol. 253. 2003.

[34] Isabelle Guyon et al. "Design and analysis of the causation and prediction challenge". In: *Causation and Prediction Challenge*. PMLR. 2008, pp. 1–33.

[35] Filip Hanzely et al. "Stochastic subspace cubic Newton method". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 4027–4038.

[36] K Jbilou and H Sadok. "Vector extrapolation methods. Applications and numerical comparison". In: *Journal of Computational and Applied Mathematics* 122.1-2 (2000), pp. 149–165.

[37] Khalide Jbilou and Hassane Sadok. "Analysis of some vector extrapolation methods for solving systems of linear equations". In: *Numerische Mathematik* 70.1 (1995), pp. 73–89.

[38] Khalide Jbilou and Hassane Sadok. "Some results about vector extrapolation methods and related fixed-point iterations". In: *Journal of Computational and Applied Mathematics* 36.3 (1991), pp. 385–398.

[39] Dmitry Kamzolov et al. "Accelerated Adaptive Cubic Regularized Quasi-Newton Methods". In: *arXiv preprint arXiv:2302.04987* (2023).

[40] Dmitry Kovalev and Alexander Gasnikov. "The first optimal acceleration of high-order methods in smooth convex optimization". In: *arXiv preprint arXiv:2205.09647* (2022).

[41] Dachao Lin, Haishan Ye, and Zhihua Zhang. "Explicit convergence rates of greedy and random quasi-Newton methods". In: *Journal of Machine Learning Research* 23.162 (2022), pp. 1–40.

[42] Dachao Lin, Haishan Ye, and Zhihua Zhang. "Greedy and random quasi-newton methods with faster explicit superlinear convergence". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6646–6657.

[43] Renato DC Monteiro and Benar Fux Svaiter. "An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods". In: *SIAM Journal on Optimization* 23.2 (2013), pp. 1092–1125.

[44] Yurii Nesterov. "Accelerating the cubic regularization of Newton's method on convex problems". In: *Mathematical Programming* 112.1 (2008), pp. 159–181.

[45] Yurii Nesterov. *Introductory lectures on convex optimization*. Springer, 2004.

[46] Yurii Nesterov and Boris T Polyak. "Cubic regularization of Newton method and its global performance". In: *Mathematical Programming* 108.1 (2006), pp. 177–205.

[47] Courtney Paquette et al. "Halting Time is predictable for large models: A universality property and average-case analysis". In: *Foundations of Computational Mathematics* (2022).

[48] Fabian Pedregosa and Damien Scieur. "Acceleration through spectral density estimation". In: *Proceedings of the 37th International Conference on Machine Learning (ICML)*. 2020.

[49] MJD Powell. "How bad are the BFGS and DFP methods when the objective function is quadratic?" In: *Mathematical Programming* 34 (1986), pp. 34–47.

[50] Anton Rodomanov and Yurii Nesterov. "Greedy quasi-Newton methods with explicit superlinear convergence". In: *SIAM Journal on Optimization* 31.1 (2021), pp. 785–811.

[51] Anton Rodomanov and Yurii Nesterov. "New results on superlinear convergence of classical quasi-Newton methods". In: *Journal of optimization theory and applications* 188 (2021), pp. 744–769.

[52] Anton Rodomanov and Yurii Nesterov. "Rates of superlinear convergence for classical quasi-Newton methods". In: *Mathematical Programming* (2021), pp. 1–32.

[53] Mark Schmidt. "minFunc: unconstrained differentiable multivariate optimization in Matlab". In: *Software available at http://www. cs. ubc. ca/˜ schmidtm/Software/minFunc. htm* (2005).

[54] Robert B Schnabel. *Quasi-Newton Methods Using Multiple Secant Equations*. Tech. rep. COLORADO UNIV AT BOULDER DEPT OF COMPUTER SCIENCE, 1983.

[55] Damien Scieur. "Generalized framework for nonlinear acceleration". In: *arXiv preprint arXiv:1903.08764* (2019).

[56] Damien Scieur, Alexandre d'Aspremont, and Francis Bach. "Regularized nonlinear acceleration". In: *Advances in Neural Information Processing Systems (NIPS)*. 2016.

[57] Damien Scieur, Alexandre d'Aspremont, and Francis Bach. "Regularized nonlinear acceleration". In: *Mathematical Programming* (2020).

[58] Damien Scieur and Fabian Pedregosa. "Universal Asymptotic Optimality of Polyak Momentum". In: *Proceedings of the 37th International Conference on Machine Learning (ICML)*. 2020.

[59] Damien Scieur et al. "Generalization of Quasi-Newton methods: application to robust symmetric multisecant updates". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 550–558.

[60] Damien Scieur et al. "Online Regularized Nonlinear Acceleration". In: *arXiv:1805.09639* (2018).

[61] David F Shanno. "Conditioning of quasi-Newton methods for function minimization". In: *Mathematics of computation* 24.111 (1970), pp. 647–656.

[62] Avram Sidi. "Convergence and stability properties of minimal polynomial and reduced rank extrapolation algorithms". In: *SIAM Journal on Numerical Analysis* 23.1 (1986), pp. 197–209.

[63] Avram Sidi. "Efficient implementation of minimal polynomial and reduced rank extrapolation methods". In: *Journal of Computational and Applied Mathematics* 36.3 (1991), pp. 305–337.

[64] Avram Sidi. "Extrapolation vs. projection methods for linear systems of equations". In: *Journal of Computational and Applied Mathematics* 22.1 (1988), pp. 71–88.

[65] Avram Sidi. "Minimal polynomial and reduced rank extrapolation methods are related". In: *Advances in Computational Mathematics* 43.1 (2017), pp. 151–170.

[66] Avram Sidi. *Vector extrapolation methods with applications*. SIAM, 2017.

[67] Avram Sidi. "Vector extrapolation methods with applications to solution of large systems of equations and to PageRank computations". In: *Computers & Mathematics with Applications* 56.1 (2008), pp. 1–24.

[68] Avram Sidi and Jacob Bridger. "Convergence and stability analyses for some vector extrapolation methods in the presence of defective iteration matrices". In: *Journal of Computational and Applied Mathematics* 22.1 (1988), pp. 35–61.

[69] Avram Sidi and Yair Shapira. "Upper bounds for convergence rates of acceleration methods with initial iterations". In: *Numerical Algorithms* 18.2 (1998), pp. 113–132.

[70] Andrzej Stachurski. "Superlinear convergence of Broyden's bounded θ-class of methods". In: *Mathematical Programming* 20.1 (1981), pp. 196–212.

[71] Alex Toth and CT Kelley. "Convergence analysis for Anderson acceleration". In: *SIAM Journal on Numerical Analysis* 53.2 (2015), pp. 805–819.

[72] Homer F Walker and Peng Ni. "Anderson acceleration for fixed-point iterations". In: *SIAM Journal on Numerical Analysis* 49.4 (2011), pp. 1715–1735.

[73] Zengxin Wei et al. "The superlinear convergence of a modified BFGS-type method for unconstrained optimization". In: *Computational optimization and applications* 29 (2004), pp. 315–332.

[74] Hiroshi Yabe, Hideho Ogasawara, and Masayuki Yoshino. "Local and superlinear convergence of quasi-Newton methods based on modified secant conditions". In: *Journal of Computational and Applied Mathematics* 205.1 (2007), pp. 617–632.

[75] Hiroshi Yabe and Naokazu Yamaki. "Local and superlinear convergence of structured quasi-Newton methods for nonlinear optimization". In: *Journal of the Operations Research Society of Japan* 39.4 (1996), pp. 541–557.

[76] Junzi Zhang, Brendan O'Donoghue, and Stephen Boyd. "Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations". In: *SIAM Journal on Optimization* 30.4 (2020), pp. 3170–3197.

Supplementary Materials

Preconditioner for the Type 1 Step

This section presents a simple diagonal preconditionner that helps in reducing the theoretical constants that involves the error vector ε. This simple preconditionner impacts the efficiency of the methods presented in this paper, in particular, the accelerated Type-1 step.

The type-1 step (10) in Theorem 2 from section 2 is actually a simplified, looser upper bound. Looking at the last steps of proof of Theorem 2, the upper bound on f actually reads

$$f(x_+) \leq f(x) + \nabla f(x) D\alpha + \frac{1}{2}\left((D\alpha)^T G \alpha + \frac{L\|D\alpha\|}{2} \sum_{i=1}^{N} |\alpha_i| \varepsilon_i \right) + \frac{L}{6}\|D\alpha\|^3.$$

However, the minimization of the upper may be intractable as it is a non smooth, potentially non convex problem. Therefore, it uses the bounds

$$\sum_{i=1}^{N} |\alpha_i| \varepsilon_i = \alpha^T (\text{sign}(\alpha) \odot \varepsilon) \leq \|\alpha\| \|\varepsilon\|,$$
$$\|D\alpha\| \leq \|D\| \|\alpha\|.$$

Diagonal preconditioner Introducing a diagonal preconditioner $\mathcal{D}$ leads to those alternatives bounds,

$$\sum_{i=1}^{N} |\alpha_i| \varepsilon_i \leq \|\mathcal{D}\alpha\| \|\mathcal{D}^{-1}\varepsilon\|,$$
$$\|D\alpha\| \leq \|D\mathcal{D}^{-1}\| \|\mathcal{D}\alpha\|.$$

which gives the following type-1 upper bound on the function values,

$$f(x_+) \leq f(x) + \nabla f(x) D\alpha + \frac{\alpha^T \tilde{H} \alpha}{2} + \frac{L}{6}\|D\alpha\|^3,$$

where

$$\tilde{H} = \frac{R^T D + D^T R + L\|D\mathcal{D}^{-1}\| \|\mathcal{D}^{-1}\varepsilon\| \mathcal{D}^2}{2}.$$

The diagonal preconditioner can be set, for instance, to $\texttt{ddiag}(D^T D)$, where $\texttt{ddiag}$ is the operator that extract the diagonal of a matrix. There are two important benefits to use the diagonal preconditioner, as it **1)** diminishes the condition number of the matrix D, **2)** diminishes the constant δ. The effect of this preconditioner is more important when there is a big difference between the norm of the direction d_i, in particular for the *Greedy* strategies and memorize the difference between iterates $x_i - x_{i-1}$ (that can be large) and the forward estimates $x_{i+\frac{1}{2}} - x_i$ (that can be small).

A Known rates of convergence

This section explores the known rates of convergence for different optimization methods. Specifically, it focuses on two scenarios: functions with Lipschitz continuous gradient and functions with Lipschitz-continuous Hessian. For smooth functions, the rates of plain gradient descent and its accelerated version are examined. On the other hand, for functions with a Lipschitz-continuous Hessian, the rates of the cubic regularized Newton method and its accelerated variant are investigated.

When the function is smooth, i.e., has Lipschitz continuous gradients,

$$f(y) \leq f(x) + \nabla f(x)(y-x) + \frac{\mathcal{L}}{2}\|y-x\|^2,$$

the rates of plain gradient descent and its accelerated version read [45]

$$\min_{0\leq i\leq t} \|\nabla f(x_i)\| \leq \sqrt{\frac{\mathcal{L}f(x_0) - f^\star}{t+1}}, \qquad \text{(plain, non-convex)} \tag{14}$$

$$f(x_t) - f(x^\star) \leq \mathcal{L}\frac{2}{t+4}\|x_0 - x^\star\|^2, \qquad \text{(plain, convex)} \tag{15}$$

$$f(x_t) - f(x^\star) \leq \mathcal{L}\frac{4}{(t+2)^2}\|x_0 - x^\star\|^2. \qquad \text{(accelerated)} \tag{16}$$

However, the class of functions considered in this paper is *not* the class of smooth functions. However, if the sequence $\{x_t\}$ is monotone, the constant $\mathcal{L}$ can be estimated as

$$\mathcal{L} \leq LR.$$

On the other hand, when the function has a Lipschitz-continuous Hessian, the cubic regularized Newton method and its accelerated version converge with the following rates [46, 44, 35]:

$$\min_{0\leq i\leq t} \|\nabla f(x_i)\| \leq \frac{16L}{9}\left(\frac{3(f(x_0) - f^\star)}{2tM_{\min}}\right)^{2/3}, \qquad \text{(plain, non-convex)} \tag{17}$$

$$f(x_t) - f(x^\star) \leq \quad 9L\frac{R^3}{(t+4)^2}, \qquad \text{(plain, convex)} \tag{18}$$

$$\mathbb{E}[f(x_t)] - f(x^\star) \leq \left(\frac{d-N}{N}\right)\frac{\mathcal{L}(3R)^2}{2t} + \left(\frac{d}{N}\right)^2\frac{L(3R)^3}{3t^2} + O\left(\frac{1}{t^3}\right), \qquad \text{(stochastic, convex)} \tag{19}$$

$$f(x_t) - f(x^\star) \leq L\frac{14\|x_0 - x^\star\|}{t(t+1)(t+2)}. \qquad \text{(accelerated)} \tag{20}$$

Overall, the rates are faster than first order methods.

B Linking with Existing Methods

This section presents the fundamentals of Anderson/nonlinear acceleration (appendix B.1), quasi-Newton schemes (appendix B.2), and their relationship with the proposed method in this paper (appendix B.3).

B.1 Anderson Acceleration and Nonlinear Acceleration

Anderson acceleration, also known as nonlinear acceleration, is a powerful technique that enhances the convergence speed of fixed point iterations and optimization algorithms. Initially developed for solving linear systems, Anderson acceleration has gained popularity due to its effectiveness in accelerating iterative methods. The method leverages previous iterations to construct an improved estimate of the objective function's minimizer.

The Anderson acceleration algorithm employs the following approximation to compute weights:

$$\nabla f\left(\sum_{i=0}^{N}\beta_i x_i\right) \approx \sum_{i=0}^{N}\beta_i \nabla f(x_i), \quad \sum_{i=0}^{N}\beta_i = 1.$$

When the function f is quadratic, this approximation becomes an equality. The underlying idea is as follows: since the optimum satisfies $\nabla f(x^\star) = 0$,

$$\sum_{i=0}^{N}\beta_i \nabla f(x_i) \approx 0 \quad \Rightarrow \nabla f\left(\sum_{i=0}^{N}\beta_i x_i\right) \approx 0 \quad \Rightarrow \sum_{i=0}^{N}\beta_i x_i \approx x^\star.$$

The Anderson acceleration steps is thus given by

$$x_{t+1} = \sum_{i=0}^{N}\beta_i^\star x_{t-i+1}, \quad \beta^\star = \arg\min_{\beta} \|\sum_{i=0}^{N}\beta_i \nabla f(x_{t-i+1})\|^2$$

Over the past decades, the ideas behind Anderson acceleration have been refined. For example, the constraint can be eliminated by considering the step $x_{t+1} - x_t$ instead:

$$\begin{aligned} x_{t+1} - x_t &= \sum_{i=0}^{N}\beta_i x_{t-i+1} - x_t \\ &= \sum_{i=0}^{N}\tilde{\beta}_i x_{t-i+1}. \end{aligned}$$

The vector $\tilde{\beta}_i$ has the property that its sum equals zero. Hence, it can be rewritten as

$$\begin{aligned} x_{t+1} - x_t &= \sum_{i=1}^{N}\alpha_i (x_{t-i+1} - x_{t-i}) \\ \alpha &= \arg\min_{\alpha} \left\| \nabla f(x_t) + \sum_{i=1}^{N}\alpha_i (\nabla f(x_{t-i+1}) - \nabla f(x_{t-i})) \right\| \end{aligned}$$

where $\alpha \in \mathbb{R}^N$ has no constraint. By writing $d_i = x_{t-i+1} - x_{t-i}$, $g_i = \nabla f(x_{t-i+1}) - \nabla f(x_{t-i})$, and $D = [d_t, \ldots, d_{t-N+1}]$, $G = [g_t, \ldots, g_{t-N+1}]$, the step becomes

$$x_{t+1} - x_t = D_t\alpha, \quad \alpha = \arg\min_{\alpha} \|\nabla f(x_t) + G_t\alpha\|.$$

However, this version of Anderson acceleration is non-convergent because there is no contribution from $\nabla f(x_t)$ in the step $x_{t+1} - x_t$. The most popular solution to this problem is introducing a *mixing parameter* that combines gradient steps, resulting in the following expression:

$$x_{t+1} = x_t - h\nabla f(x_t) + (D - hG)\alpha, \quad \alpha = \arg\min_{\alpha} \|\nabla f(x_t) + G\alpha\|. \qquad \text{(AA Type II)}$$

Following a similar idea, recent works have introduced a type I variant of the algorithm [23, 72, 76, 13] that minimizes the function value instead of the gradient norm:

$$x_{t+1} = x_t - h\nabla f(x_t) + (D - hG)\alpha, \quad \alpha = \arg\min f(x_t) + \nabla f(x_t) D_t \alpha + \frac{1}{2}\alpha^T D_t^T G_t \alpha, \tag{AA Type I}$$

By incorporating regularization [56, 13], globalization techniques [76], or performing a line search on the parameter h, the algorithm converges towards $x^\star$.

B.2 Single-secant and Multisecant Quasi-Newton Methods

Quasi-Newton methods, such as the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method, approximate the Hessian matrix in order to efficiently solve unconstrained optimization problems. These methods avoid the expensive computation of the exact Hessian by using iterative updates based on previous iterates and gradients of the objective function.

While the BFGS method has been discussed previously (see section 2), this section focuses on other updates commonly used in quasi-Newton methods: the Davidon-Fletcher-Powell (DFP) formula, the Symmetric Rank-One (SR1) formula, and the Broyden type-1 and type-2 updates.

B.2.1 The Ideas Behind Single-Secant and Multisecant Hessian Approximation

In quasi-Newton methods, the Hessian approximation is updated using the *secant equation*, which relates the gradients and Hessian at two different points. For a twice continuously differentiable function, the secant equation is given by:

$$\nabla f(y) - \nabla f(x) = \nabla^2 f(\xi)(y - x),$$

where ξ is a point on the line segment connecting x and y. This equation serves as the basis for updating the Hessian approximation.

Based on this remarkable identity, quasi-Newton methods update an approximation of the Hessian B_t or its inverse H_t such that the approximation satisfies

$$\nabla f(x_t) - \nabla f(x_{t-1}) = B_t(x_t - x_{t-1}), \quad H_t\left(\nabla f(x_t) - \nabla f(x_{t-1})\right) = x_t - x_{t-1}.$$

What distinguishes the different updates is how to fix the remaining degrees of freedom. For instance, the simple SR-1 method updates H_t such that

$$\min_H \|H - H_{t-1}\|_F \quad : H = H^T, \;\; H\left(\nabla f(x_t) - \nabla f(x_{t-1})\right) = x_t - x_{t-1}. \tag{21}$$

Those methods are called *single-secant* as they update H_t only one secant equation at a time. Hence, in general, H_t only satisfies the latest secant equation.

Multisecant updates, on the other hand, approximate the Hessian using a batch of secant equations. By introducing matrices $D = [d_{t-N+1}, \ldots, d_t]$ and $G_t = [g_{t-N+1}, \ldots, g_t]$, the multisecant updates satisfy

$$G_t = B_t D_t, \quad H_t G_t = D_t.$$

Unfortunately, when imposing symmetry, it is impossible satisfy multiple secants at a time [54], although there are some works trying to enforce symmetry while approximating the secant equation in a least square sense [55, 59].

When symmetry is not imposed, the solution for B_t and H_t can be obtained as:

$$B_t = G_t[D_t]^\dagger + B_0(I - D_t D_t^\dagger), \quad H_t = D_t[G_t]^\dagger + H_0(I - G_t G_t^\dagger), \tag{22}$$

where B_0 and H_0 are the initial approximations, and $[A]^\dagger$ denotes the pseudo-inverse of matrix A. Different choices of pseudo-inverse lead to different methods.

The inversion of B_t can be computed using the Woodbury matrix identity, which provides an efficient way to compute the inverse. The update for B_t^{-1} is given by:

$$B_t^{-1} = B_0^{-1}\left(I - G_t\left(D_t^\dagger B_0^{-1} G_t\right)^{-1} D_t^\dagger B_0^{-1}\right) + D_t\left(D_t^\dagger B_0^{-1} G_t\right)^{-1} D_t^\dagger B_0^{-1}.$$

This update is equivalent to the update for H_t, given that

$$B_0^{-1} = H_0, \quad \text{and} \quad G_t^\dagger = \left(D_t^\dagger B_0^{-1} G_t\right)^{-1} D_t^\dagger B_0^{-1}. \tag{23}$$

In summary, quasi-Newton methods use the secant equation to update the Hessian approximation. Single-secant methods update the approximation one secant equation at a time, while multisecant methods use a batch of secant equations. The choice of updating strategy and pseudo-inverse affects the behavior of the method.

B.2.2 Davidon-Fletcher-Powell (DFP) Formula

The DFP formula is a Quasi-Newton update rule used to iteratively refine an approximation of the inverse Hessian matrix. It is defined as follows:

$$H_t = H_{t-1} + \frac{d_t d_t^T}{d_t^T g_t} - \frac{H_{t-1} g_t g_t^T H_{t-1}}{g_t^T H_{t-1} g_t}, \tag{24}$$

In the above equation, $g_t = \nabla f(x_t) - \nabla f(x_{t-1})$ represents the difference in gradients, and $d_t = x_t - x_{t-1}$ denotes the difference in parameter values. The DFP formula updates the matrix H_t using a rank-two matrix such that it remains symmetric and positive definite.

B.2.3 Symmetric Rank-One (SR1) Formula

The Symmetric Rank-One (SR1) formula is another Quasi-Newton update rule used to estimate the inverse Hessian matrix. It is defined as:

$$H_t = H_{t-1} + \frac{(d_t - H_{t-1} g_t)(d_t - H_{t-1} g_t)^T}{(d_t - H_{t-1} g_t)^T g_t}, \tag{25}$$

Here, $g_t = \nabla f(x_t) - \nabla f(x_{t-1})$ and $d_t = x_t - x_{t-1}$. The SR1 formula updates H_t at each iteration to approximate the inverse Hessian matrix, ensuring that the resulting matrix H_t remains symmetric.

B.2.4 Multisecant Broyden Methods

The multisecant Broyden methods utilize the update equation from (22), where $A^\dagger$ is chosen as the Moore-Penrose pseudo-inverse of A, given by $A^\dagger = (A^T A)^{-1} A$. In this equation, B_0 and H_0 are scaled identity matrices. After simplification, the two types of updates can be expressed as follows:

$$B_t^{-1} = D_t \left(D_t^\dagger G_t\right)^{-1} D_t^\dagger + B_0^{-1} \left(I - G_t \left(D_t^\dagger G_t\right)^{-1} D_t^\dagger\right), \tag{26}$$

$$H_t = D_t (G_t^T G_t)^{-1} G_t^T + H_0 \left(I - G_t \left(G_t^T G_t\right)^{-1} G_t^T\right). \tag{27}$$

Both updates are quite similar, differing mainly in the choice of the pseudo-inverse of the matrix G.

B.2.5 Link with Anderson Acceleration

The connection between quasi-Newton methods and Anderson Acceleration is strong, as for instance, there exists an equivalence between Broyden methods and Anderson acceleration. To illustrate this, let's closely examine the update of α in (AA Type I):

$$\begin{aligned}
&x_{t+1} = x_t - h\nabla f(x_t) + (D_t - hG_t)\alpha, \quad \alpha = \arg\min f(x_t) + \nabla f(x_t) D_t \alpha + \frac{1}{2}\alpha^T D_t^T G_t \alpha \\
\Leftrightarrow &x_{t+1} = x_t - h\nabla f(x_t) + (D_t - hG_t)\alpha, \quad \alpha : D_t^T \nabla f(x_t) + D_t^T G_t \alpha = 0 \\
\Leftrightarrow &x_{t+1} = x_t - h\nabla f(x_t) + (D_t - hG_t)\alpha, \quad \alpha : \alpha = -(D_t^T G_t)^{-1} D_t^T \nabla f(x_t) \\
\Leftrightarrow &x_{t+1} = x_t - h\nabla f(x_t) - (D_t - hG_t)(D_t^T G_t)^{-1} D_t^T \nabla f(x_t). \\
\Leftrightarrow &x_{t+1} = x_t - \left(D_t (D_t^T G_t)^{-1} D_t^T + h\left(I - G_t (D_t^T G_t)^{-1} D_t^T\right)\right) \nabla f(x_t)
\end{aligned}$$

The above step is precisely the quasi-Newton step $x_{t+1} = x_t - B_t^{-1}\nabla f(x_t)$, where B_t^{-1} corresponds to the Broyden update given by Equation 26, with $B_0^{-1} = hI$. A similar reasoning can be applied to Equation 27.

When considering the single secant updates, following the same reasoning as in Section 3 leads to the same conclusion for the SR-1 and DFP updates.

This result is expected since the approximations H_t or B_t^{-1} satisfy the single or multisecant equation:

$$H_t G_t = D_t,$$

indicating that the matrix H_t maps vectors from the span of previous gradients to the span of previous directions. This observation justifies the construction in (4).

B.3 Links with Algorithms 1 and 2

Both Algorithms 1 and 2 can be viewed as quasi-Newton and Anderson/nonlinear acceleration schemes. The update formulas are

$$\min_\alpha f(x_t) + \nabla f(x_t)^T D_t\alpha + \frac{\alpha^T H_t \alpha}{2} + \frac{M\|D_t\alpha\|^3}{6}, \quad H_t \stackrel{\text{def}}{=} \frac{G_t^T D_t + D_t^T G_t + \mathrm{I}M\|D_t\|\|\varepsilon_t\|}{2}. \tag{Type I}$$

$$\min_\alpha \|\nabla f(x_t) + G_t\alpha\| + \frac{M}{2}\Big(\sum_{i=1}^N |\alpha_i|[\varepsilon_t]_i + \|D_t\alpha\|^2\Big), \tag{Type II}$$

The resemblance with Anderson/nonlinear acceleration is strong, as the objective function are similar. In fact, if the function is quadratic, $L = 0$ and therefore M can be set to 0 as well. In this case, the coefficients α are *exactly* the type I and type II Anderson steps eqs. (AA Type I) and (AA Type II).

The same idea holds when comparing to quasi-Newton methods. In both cases, the optimal solution $\alpha^\star$ can be written implicitly:

$$\alpha^\star = -\left(H_t + \frac{MD_t^T D_t\|D_t\alpha^\star\|}{6}\right)^{-1} D_t^T\nabla f(x_t), \tag{Type I - solution}$$

$$\alpha^\star = -\left(G_t^T G_t + \tilde{M}D_t^T D_t\right)^{-1}\left(G_t^T\nabla f(x) + \frac{\tilde{M}\|\varepsilon_t\|}{2}\partial(|\alpha^\star|)\right), \tag{Type II - solution}$$

where $\tilde{M} \stackrel{\text{def}}{=} \|\nabla f(x_t) + G_t\alpha\|M$ and $\partial(|\alpha^\star|)$ is a subgradient of $|\alpha^*|$. The step then reads

$$x_{t+1} = x_t + D\alpha^\star \tag{Generic step}$$

$$x_{t+1} = x_t - D_t\left(H_t + \frac{MD_t^T D_t\|D_t\alpha^\star\|}{6}\right)^{-1} D_t^T\nabla f(x_t), \tag{Type I - step}$$

$$x_{t+1} = x_t - D_t\left(G_t^T G_t + \tilde{M}D_t^T D_t\right)^{-1}\left(G_t^T\nabla f(x) + \frac{\tilde{M}\|\varepsilon_t\|}{2}\partial(|\alpha^\star|)\right), \tag{Type II - step}$$

The type I is a quasi-Newton step with a symetrization of $G^T D$, along with a regularization, while the type II step can be seen as a quasi-Newton method with a regularization on $R^\dagger$, with a correction term on the gradient. The Hessian approximation therefore reads

$$B_t^{-1} = D_t\left(H_t + \frac{MD_t^T D_t\|D_t\alpha^\star\|}{6}\right)^{-1} D^T, \quad H_t = D_t\left(G_t^T G_t + \tilde{M}D_t^T D_t\right)^{-1} G_t^T.$$

Again, when the objective function is quadratic, $L = 0$ and therefore $M = 0$. Moreover, when f is quadratic, the matrix multiplication $D^T G$ satisfies $D^T G + G^T D = 2D^T G$ as $D^T G$ becomes symmetric. Hence,

$$x_{t+1} = x_t - D_t\left(D_t^T G_t\right)^{-1} D_t^T\nabla f(x_t), \tag{Type I - quadratic}$$

$$x_{t+1} = x_t - D_t\left(G_t^T G_t\right)^{-1} G_t^T\nabla f(x_t), \tag{Type II quadratic}$$

The steps are *exactly* the type I and type II multisecant Broyden methods from eqs. (26) and (27), with the only difference that there is no initialization H_0 or B_0. Again, this is expected by construction of the method, where the initialization is estimated with a forward estimate (see section 5).

C Solving the sub-problems

Solving the Type 1 Subproblem The Type 1 subproblem is a well-studied problem that involves minimizing a specific objective function. A method proposed by[46] has proven to be efficient for solving this problem. The method utilizes eigenvalue decomposition on a matrix to find the optimal solution. In this paper, the matrix involved in this problem is relatively small, therefore eigenvalue decomposition is not a concern even for large-scale problems. The subproblem aims to determine the norm of the solution, and this can be achieved through solving a system of nonlinear equations using bisection or secant method.

Solving the Type 2 Subproblem The Type 2 subproblem can be formulated as a Second-Order Cone Program (SOCP). The objective function of this subproblem consists of three terms: a norm term, a sum of absolute values term, and a quadratic term. The norm term can be transformed using singular value decomposition, and the sum of absolute values term can be expressed as linear programming. The quadratic term can be simplified using a rotated quadratic cone. By utilizing these techniques, the Type 2 subproblem can be effectively solved using existing SOCP solvers.

C.1 Solving the Type 1 Subproblem

The Type 1 subproblem can be expressed as follows:

$$\min_{\alpha} \nabla f(x) D\alpha + \frac{1}{2}\alpha^T H \alpha + \frac{M}{6}\|D\alpha\|^3,$$

where H is symmetric but not necessarily positive definite. This problem has been well-studied, and [46] proposed an efficient method to solve it using eigenvalue decomposition on the matrix H. Although eigenvalue decomposition may be challenging for large-scale problems, it is not a concern here since $H \in \mathbb{R}^{N\times N}$, with a relatively small N (e.g., $N = 25$ in the experiments).

In essence, the subproblem involves determining the norm of the solution $r = \|\alpha\|$. This can be accomplished through a simple bisection on the following system of nonlinear equations:

$$\left(H + \frac{MD^T Dr}{2} I\right)\alpha = -D^t \nabla f(x), \quad \|\alpha\| = r, \quad r \geq -\lambda_{\min}(H). \tag{28}$$

Interestingly, this problem is equivalent to the following formulation, as shown in Proposition 1:

$$\left(\Lambda + \frac{Mr}{2} I\right)\tilde{\alpha} = -V^T (D^T D)^{-1/2} D^t \nabla f(x),\ \|\alpha\| = r,\ r \geq -\lambda_{\min}(H),\ \tilde{\alpha} = V^T (D^T D)^{1/2}\alpha, \tag{29}$$

which involves the eigenvalue decomposition $(D^T D)^{-1/2} H (D^T D)^{-1/2} = V\Lambda V^T$.

Proposition 1. *Problems (28) and (29) are equivalent.*

Proof. The first step is to split $D^T D = (D^T D)^{1/2}(D^T D)^{1/2}$ and then employ an eigenvalue decomposition on $(D^T D)^{-1/2} H (D^T D)^{-1/2} = V\Lambda V^T$ (where V is orthonormal due to the symmetry of the matrix):

$$\begin{aligned}
&\left(H + \frac{MD^T Dr}{2} I\right)\alpha = -D^t \nabla f(x) \\
\Leftrightarrow &(D^T D)^{1/2}\left((D^T D)^{-1/2} H (D^T D)^{-1/2} + \frac{Mr}{2} I\right)(D^T D)^{1/2}\alpha = -D^t \nabla f(x) \\
\Leftrightarrow &(D^T D)^{1/2} V \left(\Lambda + \frac{Mr}{2} I\right) V^T (D^T D)^{1/2}\alpha = -D^t \nabla f(x) \\
\Leftrightarrow &\left(\Lambda + \frac{Mr}{2} I\right) V^T (D^T D)^{1/2}\alpha = -V^T (D^T D)^{-1/2} D^t \nabla f(x) \\
\Leftrightarrow &\left(\Lambda + \frac{Mr}{2} I\right)\tilde{\alpha} = -V^T (D^T D)^{-1/2} D^t \nabla f(x).
\end{aligned}$$

□

Once the eigenvalue decomposition is performed, the subproblem (29) becomes relatively simple since it involves solving a diagonal system of equations for a fixed value of r. The main objective is to find an interval $[r_{\min}, r_{\max}]$ that encompasses the optimal value $r = \|\alpha\|$. Once this interval is identified, a straightforward bisection or secant method can be employed to obtain the optimal solution.

Finding initial bounds Starting with $r_{\min} = \max\{0, -\lambda_{\min(H)}\}$ and $r_{\max} = \max\{2r_{\min}, 1\}$,

$$\text{do } r_{\max} \leftarrow 2r_{\max} \quad \text{while } \|\tilde{\alpha}\| \geq r_{\max}.$$

where $\tilde{\alpha} = -\left(\Lambda + \frac{Mr_{\max}}{2} I\right)^{-1} V^T (D^T D)^{-1/2} D^t \nabla f(x)$. Increasing $r_{\max}$ increases the regularization, hence reduces the norm of $\tilde{\alpha}$.

Finding α After $r^\star$ has been found such that $|r^\star - \|\tilde{\alpha}\||$ is sufficiently small, the best α is simply

$$\alpha = (D^T D)^{-1/2} V \tilde{\alpha} = -(D^T D)^{-1/2} V \left(\Lambda + \frac{M r^\star}{2} I\right)^{-1} V^T (D^T D)^{-1/2} D^t \nabla f(x).$$

In the case where the diagonal matrix is not invertible, which happens when $r^\star = r_{\min}$, it suffices to use the pseudo-inverse instead.

C.2 Solving the Type 2 Subproblem

The Type 2 subproblem is given by:

$$\min_{\alpha} \underbrace{\|\nabla f(x) + G\alpha\|}_{\textbf{(a)}} + \frac{L}{2}\Big(\underbrace{\sum_{i=1}^{N} |\alpha_i| \varepsilon_i}_{\textbf{(b)}} + \underbrace{\|D\alpha\|^2}_{\textbf{(c)}} \Big). \tag{30}$$

Although it may not be immediately apparent, this subproblem can be formulated as a Second-Order Cone Program (SOCP) with $O(N)$ variables and constraints.

C.2.1 Fundamentals of SOCP

SOCP solvers handle the following conic problems:

$$\begin{aligned}
&\min_{x, t_i, \omega_i} c_0 x + \sum_i c_i [t_i; \omega_i] \quad \text{subject to} \\
&A_0 x + \sum_{i=1}^{k} A_i [t_i; \omega_i] = b \qquad \text{(SOCP Standard Matrix Form)} \\
&x \geq 0 \\
&(t_i, \omega_i) \in \mathcal{K}_i \quad \Leftrightarrow t_i \geq \|\omega_i\|, \;\; t \geq 0.
\end{aligned}$$

Here, k represents the number of cones, and the cone $\mathcal{K}$ refers to the second-order cone, also known as the *Lorenz* cone.

A useful transformation is the *rotated quadratic cone*, defined as follows:

$$[a, b, c] \in \mathcal{K}_q \quad \Leftrightarrow \quad 2ab \geq \|c\|^2.$$

The rotated quadratic cone can be reformulated as a second-order cone using a linear transformation:

$$\text{if} \quad \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & I_K \end{bmatrix} \begin{bmatrix} t \\ \omega^{(0)} \\ \omega \end{bmatrix} \quad \text{then} \quad (t, [\omega^{(0)}; \omega]) \in \mathcal{K} \quad \Leftrightarrow \quad [a, b, c] \in \mathcal{K}_q.$$

Thanks to this transformation, the rotated quadratic cone can be included in SOCP solvers.

C.2.2 SOCP Formulation of the Type 2 Subproblem

The SOCP of (30) is composed of the three terms **a**, **b**, and **c**.

Term (a) Let $U_G \Sigma_G V_G^T$ be the singular value decomposition of G. Write $P_G = U_G U_G^T$ as the projector onto the columns of G. Then,

$$\begin{aligned}
\|\nabla f(x) + R\alpha\| &= \|P_G \nabla f(x) + P_G G\alpha + (I - P_G)\nabla f(x)\| \\
&= \sqrt{\|P_G \nabla f(x) + R\alpha\|^2 + \|(I - P_G)\nabla f(x)\|^2} \\
&= \sqrt{\left\|U_G \left(U_G^T \nabla f(x) + \Sigma_G V_G^T \alpha\right)\right\|^2 + \|(I - P_G)\nabla f(x)\|^2} \\
&= \sqrt{\left\|U_G^T \nabla f(x) + \Sigma_G V_G^T \alpha\right\|^2 + \|(I - P_G)\nabla f(x)\|^2}
\end{aligned}$$

Let the vector $\omega_1 = \left[U_G^T \nabla f(x) + \Sigma_G V \alpha;\ \|(I - P_G)\nabla f(x)\|\right]$. Hence,

$$\|\nabla f(x) + G\alpha\| = \min_{t_1,\, \alpha,\, \omega_1} t_1 : (t_1, \omega_1) \in \mathcal{K}_L, \quad \omega_1 = \left[U_G^T \nabla f(x) + \Sigma_G V \alpha;\ \|(I - P_G)\nabla f(x)\|\right].$$

Term (b) This term is standard in linear programming. Let $\alpha = \alpha_+ - \alpha_-$, with $\alpha_+, \alpha_- \geq 0$,

$$\sum_{i=1}^{N} |\alpha_i| \varepsilon_i = \sum_{i=1}^{N} (\alpha_+ + \alpha_-)\varepsilon_i.$$

Term (c) Let $U_D \Sigma_D V_D^T$ be the singular value decomposition of D. Using the rotated cone, the constraint can be written as

$$2t_3 b \geq \|U_D \Sigma_D V_D \alpha\|^2 = \|\Sigma_D V_D \alpha\|^2, \quad b = \frac{1}{2}.$$

Using the transformation into a Lorenz cone, this is equivalent to

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \Sigma_D V_D^T \end{bmatrix} \begin{bmatrix} t_3 \\ b \\ \alpha \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & I_k \end{bmatrix} \begin{bmatrix} t_2 \\ \omega_2^{(0)} \\ \omega_2 \end{bmatrix}, \quad b = \frac{1}{2}, \quad (t_2,\, [\omega_2^{(0)},\, \omega_2]) \in \mathcal{K}.$$

Simplification. Note that, since $b = \frac{1}{2}$, the value can be immediately replaced. Same idea with t_3: the constraint is written as

$$t_3 = \frac{t_2 + \omega_2^{(0)}}{\sqrt{2}}, \quad t_3 \geq 0.$$

Since, by construction, $t_2 \geq \omega_2^{(0)}$ and $t_2 \geq 0$, t_3 always satisfies the condition, which means both t_3 and its constraint can be removed. The constraints thus simplify into

$$\begin{bmatrix} \frac{1}{2} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \Sigma_D V_D^T \end{bmatrix} [\alpha] = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & I_k \end{bmatrix} \begin{bmatrix} t_2 \\ \omega_2^{(0)} \\ \omega_2 \end{bmatrix}, \quad (t_2,\, [\omega_2^{(0)},\, \omega_2]) \in \mathcal{K}.$$

Final formulation Gathering all terms, the final SOCP formulation reads

$$
\begin{aligned}
\text{minimize} \quad & t_1 + \frac{L}{2}\left((\alpha_+ + \alpha_-)^T\varepsilon + t_2\right) \\
\text{subject to} \quad & \omega_1 = \left[U_G^T\nabla f(x) + \Sigma_G V_G^T\alpha \;;\; \|(I - P_G)\nabla f(x)\|\right], \\
& \alpha_+, \alpha_- \geq 0 \\
& \alpha = \alpha_+ - \alpha_- \\
& \begin{bmatrix} \mathbf{0}_{1\times N} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \Sigma_D V_D^T & \mathbf{0}_{N\times 1} & \mathbf{0}_{N\times 1} & -I_N \end{bmatrix} \begin{bmatrix} \alpha \\ t_2 \\ \omega_2^{(0)} \\ \omega_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{1}{2} \\ \mathbf{0}_{N\times 1} \end{bmatrix} \\
& (t_1,\, \omega_1) \in \mathcal{K}, \quad (t_2, [\omega_2^{(0)}; \omega_2]) \in \mathcal{K}_L, \quad t_2 \geq 0.
\end{aligned}
$$

Standard matrix formulation The SOCP can be written under the standard matrix form (SOCP Standard Matrix Form). Let the variables

$$
\alpha_+,\, \alpha_- \geq 0, \quad (t_1, \omega_1) \in \mathcal{K}_1, \quad (t_2, [\omega_2^{(0)}\omega_2]) \in \mathcal{K}_2,
$$

where t_1, t_2, and $\omega_2^{(0)}$ are scalars, ω_2, α_+, and α_- are vectors of size N, and ω_1 is a vector of size $N+1$. The SOCP matrices read

$$
\begin{aligned}
& c_0 = \begin{bmatrix} \frac{L\varepsilon^T}{2} & \frac{L\varepsilon^T}{2} \end{bmatrix} \quad c_1 = [1 \quad \mathbf{0}_{1\times N+1}] \quad c_2 = \begin{bmatrix} \frac{L}{2\sqrt{2}} & \frac{L}{2\sqrt{2}} & \mathbf{0}_{1\times N} \end{bmatrix} \\
& A_0 = \begin{bmatrix} -\Sigma_G V_G^T & \Sigma_G V_G^T \\ \mathbf{0}_{2\times N} & \mathbf{0}_{2\times N} \\ \Sigma_D V_D^T & -\Sigma_D V_D^T \end{bmatrix} \\
& A_1 = \begin{bmatrix} \mathbf{0}_{N+1\times 1} & I_{N+1\times N+1} \\ \mathbf{0}_{N+1\times 1} & \mathbf{0}_{N+1\times N+1} \end{bmatrix} \\
& A_2 = \begin{bmatrix} \mathbf{0}_{N+1\times 1} & \mathbf{0}_{N+1\times 1} & \mathbf{0}_{N+1\times N} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \mathbf{0}_{1\times N} \\ \mathbf{0}_{N\times 1} & \mathbf{0}_{N\times 1} & -I_{N\times N} \end{bmatrix} \\
& b = \begin{bmatrix} \nabla f(x)^T U_G & \|(I - P_R)\nabla f(x)\| & -\frac{1}{2} & \mathbf{0}_{N\times 1} \end{bmatrix}^T.
\end{aligned}
$$

This completes the SOCP formulation of the type 2 subproblem.

D Accelerated Algorithm

Algorithm 6 Adaptive Accelerated Type-I Iterative Algorithm

Require: First-order oracle f, initial iterate and smoothness $x_0,\ M_0$, number of iterations T.

$\lambda_0^{(1)} \leftarrow 0, \lambda_0^{(2)} \leftarrow 0, \Delta \leftarrow -\infty, (M_0)_t \leftarrow M_0$

Initialize G_0, D_0, ε_0 (see section 5)

$x_1, M_1 \leftarrow$ `[algorithm 1]`$(f, G_0, D_0, \varepsilon_0, x_0, (M_0)_0)$

Initialize $\ell_0^{(0)} = f(x_1), \quad \ell_0^{(1)} = 0, \Delta = \|x_1 - x_0\|$

for $t = 1, \ldots, T-1$ **do**

Update G_t, D_t, ε_t (see section 5)

Set $b_t \leftarrow \frac{(t+1)(t+2)}{2}$, $B_t \leftarrow \frac{t(t+1)(t+2)}{6}$, $\beta_t \leftarrow \frac{3}{t+3}$.

Update $\ell_t^{(0)} \leftarrow \ell_{t-1}^{(0)} + b_{t-1}[f(x_t) - \nabla f(x_t)^T x_t], \quad \ell_t^{(1)} \leftarrow \ell_{t-1}^{(1)} + b_{t-1}\nabla f(x_t)$

do

`ValidBound` $\leftarrow$ `True`

Set $v_t \leftarrow \arg\min_v \phi_t(v)$ (See proposition 2).

Let $y_t \leftarrow \frac{3}{t+3} v_t + \frac{k}{t+3} x_t$

$\{x_{t+1}, \alpha_t\, M_{t+1}, \gamma_t,$ `ExitFlag`$\} \leftarrow$ `[algorithm 4]`$(f, G_t, D_t, \varepsilon_t, y_t, (M_0)_t, \Delta)$

`%% Check if the next` ϕ `is still a lower bound for` $f(x_{t+1})$

Define $\phi_+ = \phi_t + b_t[f(x_{t+1} + \nabla f(x_{t+1})(x - x_{t+1})]$.

Set $v_+ \leftarrow \arg\min_v \phi_+(v)$ (See proposition 2).

if $\Phi_+(v_+) \le f(x_{t+1})$ **then** `%% Parameters adjustment if needed`

`ValidBound` $\leftarrow$ `False` `%% Unsuccessful iteration:` $\phi_{t+1}(v_{t+1}) \ge f(x_{t+1})$.

if `ExitFlag` is `LargeStep` **then**

If $\lambda_t^{(2)} = 0$ **then** $\lambda_t^{(2)} \leftarrow \frac{16}{9} \frac{(b_t \|\nabla f(x_{t+1})\|)^3}{(B_{t+1} \nabla f(x_{t+1})^T D_t \alpha_t)^2}$. **Else,** $\lambda_t^{(2)} \leftarrow 2\lambda_t^{(2)}$.

else `%% Exitflag is SmallStep`

If $\lambda_t^{(1)} = 0$ **then** $\lambda_t^{(1)} \leftarrow \frac{-b_t^2 \|\nabla f(x_{t+1})\|^2}{2B_{t+1} \nabla f(x_{t+1})^T D_t \alpha_t}$. **Else,** $\lambda_t^{(1)} \leftarrow 2\lambda_t^{(1)}$.

end if

if $(M_0)_{t+1} < M_{t+1}$ **then** $(M_0)_{t+1} \leftarrow M_{t+1} \left(\frac{\|\varepsilon_t\|}{\|D_t\|} + \frac{\|D_t \alpha_t\|}{2} \right)$ `%% Rescaling`

end if

else

$\{\lambda_{t+1}^{(1)}, \lambda_{t+1}^{(2)}\} \leftarrow \{\lambda_t^{(1)}, \lambda_t^{(2)}\}, \quad (M_0)_{t+1} \leftarrow \frac{M_{t+1}}{2}$ `%% Successful iteration`

end if

while `ValidBound is False`

end for

return x_T

Proposition 2. *Let v_t be the the minimizer of*

$$\phi_t(v) = \ell_t^{(0)} + \left[\ell_t^{(1)}\right]^T v + \frac{\lambda_t^{(1)}}{2}\|v - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v - x_0\|^3.$$

where $\lambda_t^{(1,2)} \ge 0$. Let $r_t = \|v_t - x_0\|$. Then,

$$r_t = \|v_t - x_0\| = \begin{cases} 0 & \text{if } \lambda_t^{(1)} = \lambda_t^{(2)} = 0 \\ \frac{\|\ell_t^{(1)}\|}{\lambda_t^{(1)}} & \text{if } \lambda_t^{(1)} > 0 \text{ and } \lambda_t^{(2)} = 0 \\ \frac{-\lambda_t^{(1)} + \sqrt{[\lambda_t^{(1)}]^2 + 2\lambda_t^{(2)}\|\ell_k\|}}{\lambda_2^{(2)}} & \text{if } \lambda_t^{(2)} > 0 \end{cases}$$

$$v_t = \arg\min \Phi_t(x) = x_0 - r_t \frac{\ell_t^{(1)}}{\|\ell_t^{(1)}\|}$$

E Additional Numerical Experiments

This section presents additional numerical experiments.

Methods The methods compared are the type 1 and type 2 steps with the following strategies: *Iterate only*, *Forward estimate only*, *Greedy* (refer to section 5), and the accelerated type 1 method with the strategy *forward estimate only*. The batch methods are not included as they perform poorly in terms of the number of oracle calls. The baseline is the L-BFGS method from `minFunc` [53].

Method parameters In all experiments, the memory of the methods is set to $N = 25$. The parameters of the L-BFGS are left untouched except for the memory. The initial smoothness parameter is set to 1 for the type 1 and type 2 methods. The initial point is randomly generated by the function `randn()` in `Matlab`, with a seed of 0.

Functions The minimized problems are: square loss with cubic regularization, logistic loss with small quadratic regularization, and the generalized Rosenbrock function. The regularization parameter of the square loss is set to $1e-3$ times the norm of the Hessian, and the regularization of the logistic loss is set to $1e-10$ times the square norm of the feature matrix.

Dataset The datasets for the square loss and the logistic loss are Madelon [33], Sido0 [34], and Marti2 [34] datasets.

Post-processing The dataset matrix is normalized by its norm, and a feature vector of ones is added to the data matrix.

E.1 Nonconvex optimization

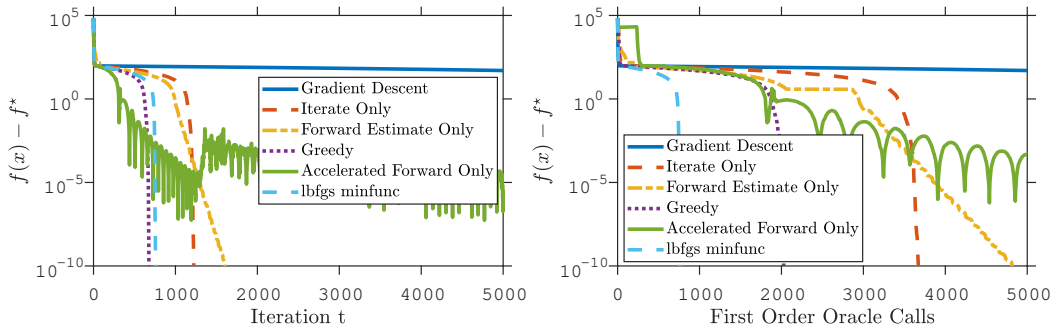

Figure 2: Comparison of type 1 methods on the Generalized Rosenbrock function in $\mathbb{R}^{100}$

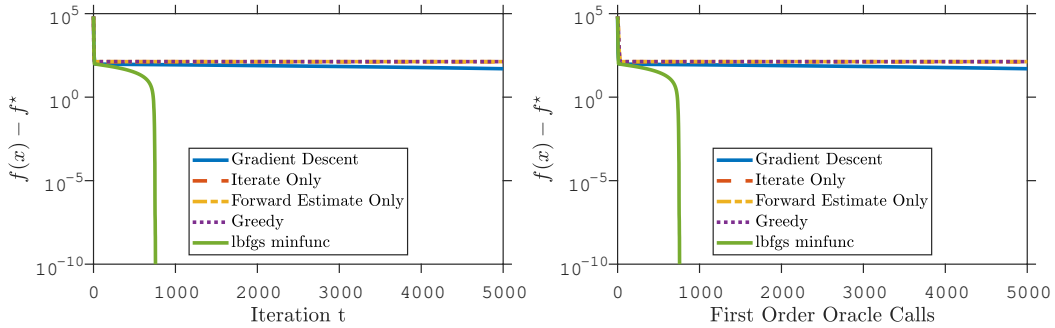

Figure 3: Comparison of type 2 methods on the Generalized Rosenbrock function in $\mathbb{R}^{100}$

E.2 Comparison of Type 1 Methods on Convex Problems

E.2.1 Square loss and cubic regularization

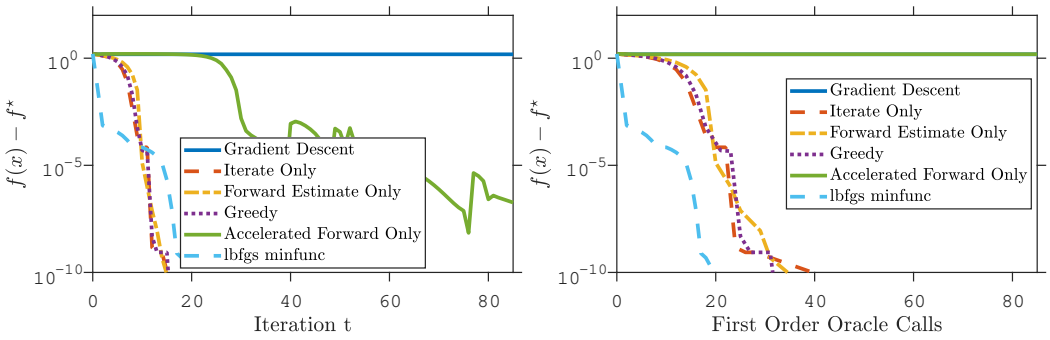

Figure 4: Comparison of type 1 methods: Square loss and cubic regularization on Madelon dataset

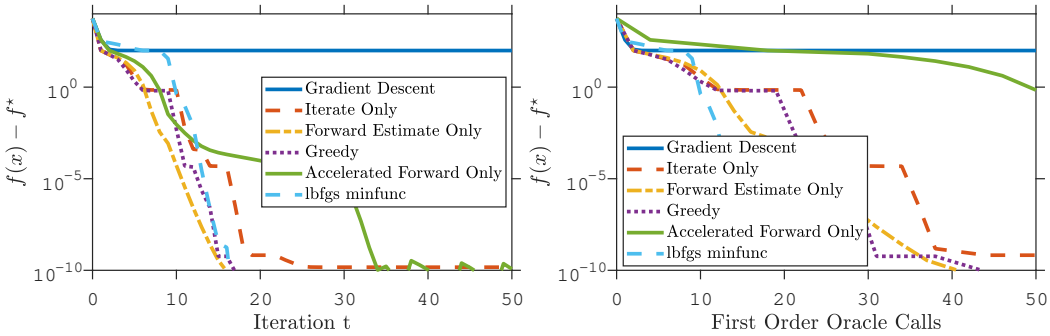

Figure 5: Comparison of type 1 methods: Square loss and cubic regularization on sido0 dataset

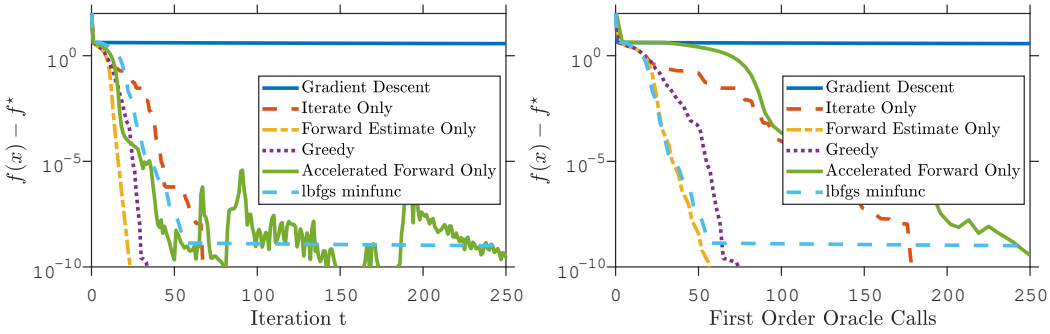

Figure 6: Comparison of type 1 methods: Square loss and cubic regularization on marti2 dataset

E.2.2 Logistic regression

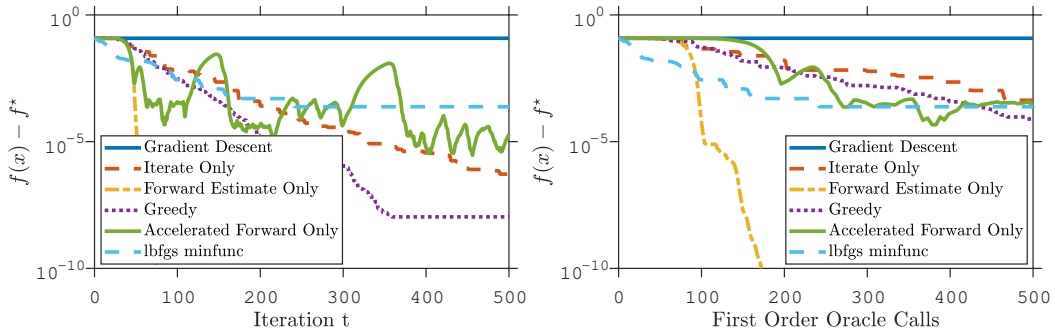

Figure 7: Comparison of type 1 methods: Logistic loss and cubic regularization on Madelon dataset

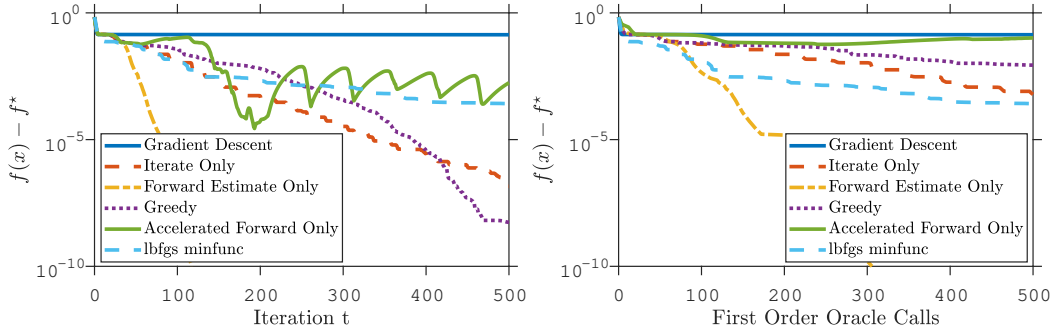

Figure 8: Comparison of type 1 methods: Logistic loss and cubic regularization on sido0 dataset

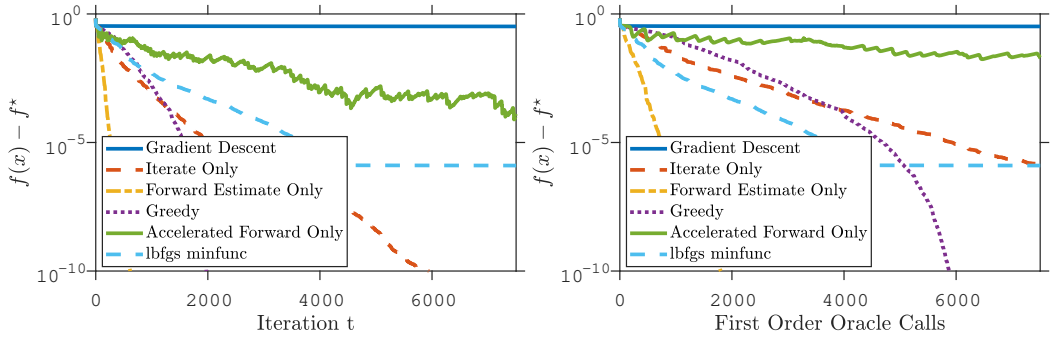

Figure 9: Comparison of type 1 methods: Logistic loss and cubic regularization on marti2 dataset

E.3 Comparison of Type 2 Methods on Convex Problems

E.3.1 Square loss and cubic regularization

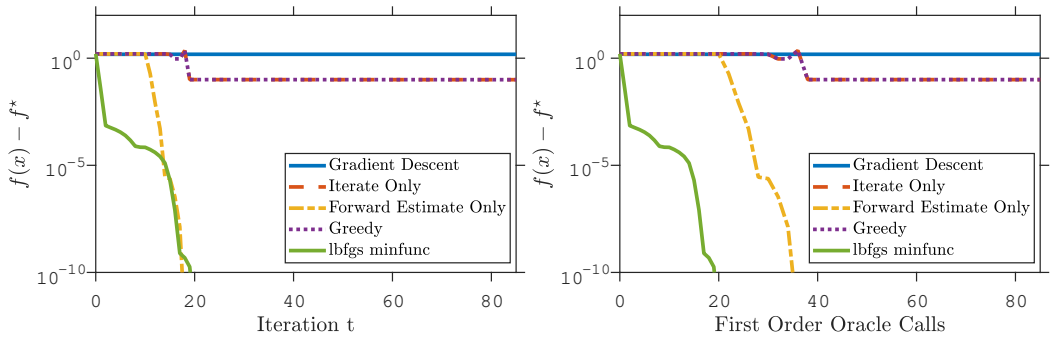

Figure 10: Comparison of type 2 methods: Square loss and cubic regularization on Madelon dataset

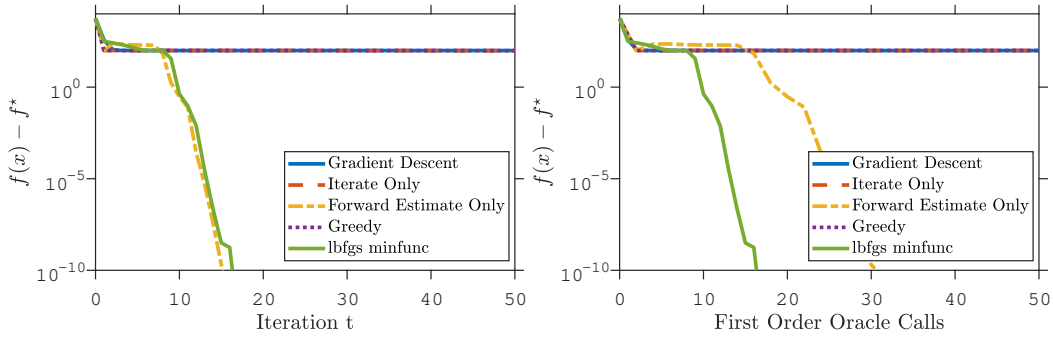

Figure 11: Comparison of type 2 methods: Square loss and cubic regularization on sido0 dataset

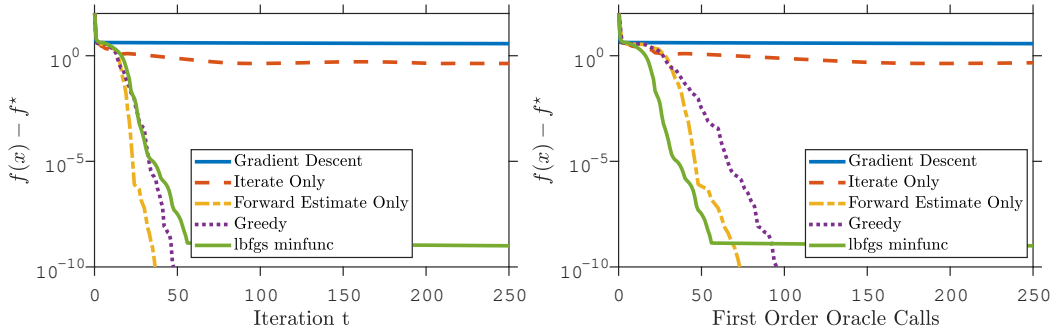

Figure 12: Comparison of type 2 methods: Square loss and cubic regularization on marti2 dataset

E.3.2 Logistic regression

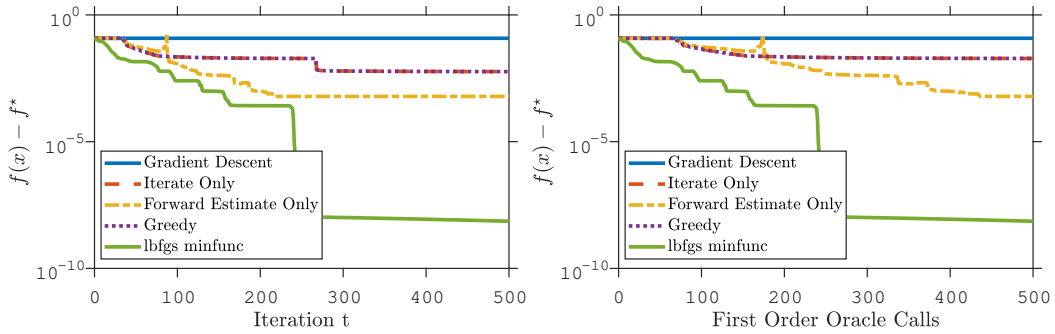

Figure 13: Comparison of type 2 methods: Logistic loss and cubic regularization on Madelon dataset

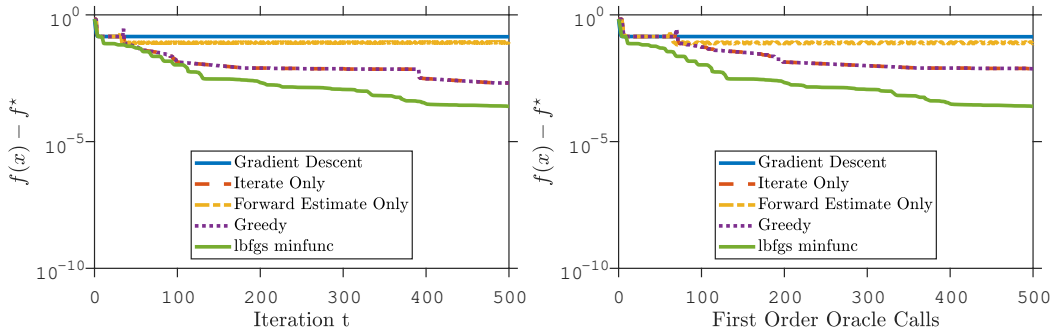

Figure 14: Comparison of type 2 methods: Logistic loss and cubic regularization on sido0 dataset

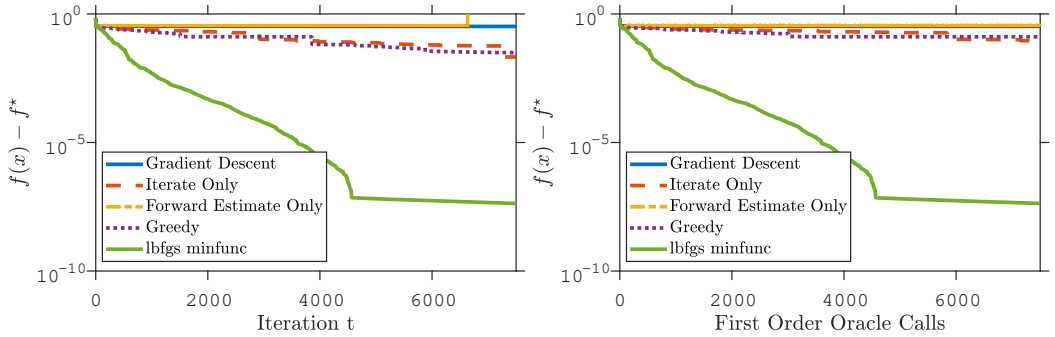

Figure 15: Comparison of type 2 methods: Logistic loss and cubic regularization on marti2 dataset

F Missing proofs

F.1 Technical results

In this section, the following definitions simplify the notations:

$$D_\dagger = (D^T D)^{-1} D^T, \tag{31}$$
$$D_\dagger^T = D(D^T D)^{-1}, \tag{32}$$
$$\kappa_D = \|D_\dagger\| \|D\|, \tag{33}$$
$$\tilde{H} = D_\dagger^T H D_\dagger \qquad \text{where } H \text{ is defined in (10).} \tag{34}$$

Note that the pseudo inverse $D_\dagger$ exists under Requirement 3.

Proposition 3. *The first-order and second-order conditions of the subproblem in algorithm 1 read*

$$D^T \nabla f(x) + H\alpha + \frac{M}{2} D^T D\alpha \|D\alpha\| = 0, \tag{35}$$
$$H + \frac{M}{2} D^T D \|D\alpha\| \succeq 0. \tag{36}$$

Proof. See [44], equation (3.3), and [46], equation (2.7). □

Proposition 4. *Let f satisfies Assumption 1 and $B \in \mathbb{R}^{d\times d}$ be any matrix. Then,*

$$\|\nabla f(x) + BD\alpha - \nabla f(x_+)\| \leq \frac{L}{2}\|D\alpha\|^2 + \|[B - \nabla^2 f(x)]D\alpha\|.$$

Proof. The result follows directly from (5),

$$\begin{aligned}\|\nabla f(x) + BD\alpha - \nabla f(x_+)\| &\leq \|\nabla f(x) + \nabla^2 f(x) D\alpha - \nabla f(x_+)\| + \|BD\alpha - \nabla^2 f(x) D\alpha\| \\ &\leq \frac{L}{2}\|D\alpha\|^2 + \|[B - \nabla^2 f(x)]D\alpha\|.\end{aligned}$$

□

Proposition 5. *Assume the matrix D satisfies Requirement 1b, and α satisfies the first-order condition (35). Let $\tilde{H}$ be defined in (34). Then,*

$$\|\nabla f(x) + BD\alpha - \nabla f(x_+)\| = \|(\tilde{H} - B + \frac{M\|D\alpha\|}{2})D\alpha + \nabla f(x_+)\|$$

Proof. The following equation follows from the optimality condition multiplied by $D(D^T D)^{-1}$, writing $P = DD_\dagger = D_\dagger^T D^T$, assuming $P\nabla f(x) = \nabla f(x)$,

$$\nabla f(x) + (\tilde{H} + \frac{M\|D\alpha\|}{2})D\alpha = 0.$$

It suffices to replace $\nabla f(x)$. □

Proposition 6. *Assume D satisfies Requirement 1b. Let $\tilde{H}$ be defined in (34). Then, if $B = \tilde{H} - M\gamma$ in proposition 4, the following holds:*

$$\|[B - \nabla^2 f(x)]D\alpha\| \leq \|D\alpha\| \left(\tfrac{L}{2}\|D_\dagger\|\|\varepsilon\| + \|(I-P)\nabla^2 f(x) P\| + M\left\|D_\dagger^T D_\dagger \tfrac{\|D\|\|\varepsilon\|}{2} - \gamma P\right\|\right)$$

Proof. Since

$$\nabla^2 f(x) D\alpha = P\nabla^2 f(x) PD\alpha + (I-P)\nabla^2 f(x) PD\alpha,$$

where $P = D(D^T D)^{-1} D^T$, and because $PD = D$ and

$$\tilde{H} = D_\dagger^T \left(\frac{D^T G + G^T D}{2} + \frac{M\|D\|\|\varepsilon\|}{2}\right) D_\dagger = \frac{PGD_\dagger + D_\dagger^T G^T P}{2} + D_\dagger^T D_\dagger \frac{M\|D\|\|\varepsilon\|}{2},$$

the inequality becomes

$$\|[B-\nabla^2 f(x)]D\alpha\| \leq \left\| \left(\frac{PGD_\dagger + D_\dagger^T G^T P}{2} - P\nabla^2 f(x) P \right) D\alpha \right\| \tag{37}$$

$$+ \left\| \left(D_\dagger^T D_\dagger \frac{M\|D\|\|\varepsilon\|}{2} - M\gamma P - (I-P)\nabla^2 f(x) P \right) D\alpha \right\| \tag{38}$$

The term (38) can be decomposed into

$$\left\| \left(D_\dagger^T D_\dagger \frac{M\|D\|\|\varepsilon\|}{2} - M\gamma P - (I-P)\nabla^2 f(x) P \right) D\alpha \right\|$$
$$= \left\| P \left(\left(D_\dagger^T D_\dagger \frac{M\|D\|\|\varepsilon\|}{2} - M\gamma \right) D\alpha - (I-P)\nabla^2 f(x) P D\alpha \right) \right\|$$

Since P satisfies $\|Pv_1 + (I-P)v_2\| = \|Pv_1\| + \|(I-P)v_2\|$,

$$\|[B-\nabla^2 f(x)]D\alpha\| \leq \left\| \left[\frac{PGD_\dagger + D_\dagger^T G^T P}{2} - P\nabla^2 f(x) P \right] D\alpha \right\|$$
$$+ M\|D\alpha\| \left\| D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2} - P\gamma \right\| \tag{39}$$
$$+ \|D\alpha\|\|(I-P)\nabla^2 f(x) P\|.$$

It remains to bound the first from (37). Since $D^T D_\dagger = D_\dagger^T D^T = P$, $D_\dagger D = \mathrm{I}$, $PD = D$, and $\|P\| = 1$,

$$\begin{aligned} & \left\| \left[\frac{PGD_\dagger + D_\dagger^T G^T P}{2} - P\nabla^2 f(x) P \right] D\alpha \right\| \\ \leq\ & \frac{1}{2} \left(\left\| (PGD_\dagger - P\nabla^2 f(x) P) D\alpha \right\| + \left\| (D_\dagger^T G^T P - P\nabla^2 f(x) P) D\alpha \right\| \right) \\ \leq\ & \frac{1}{2} \left(\left\| G\alpha - \nabla^2 f(x) D\alpha \right\| + \|D_\dagger\| \left\| (G^T - D^T \nabla^2 f(x)) D\alpha \right\| \right) \end{aligned}$$

Using inequality (8) for the first term and (9) for second gives

$$\left\| \left[\frac{PGD_\dagger + D_\dagger^T G^T P}{2} - P\nabla^2 f(x) P \right] D\alpha \right\| \leq \frac{1}{2} \left(\frac{L}{2} \sum_{i=1}^N |\alpha_i| \varepsilon_i + \|D_\dagger\| \frac{L\|D\alpha\|}{2} \|\varepsilon\| \right)$$

Because $\sum_{i=1}^N |\alpha_i| \varepsilon_i \leq \|\alpha\|\|\varepsilon\| \leq \|D_\dagger\|\|D\alpha\|$,

$$\left\| \left[\frac{PGD_\dagger + D_\dagger^T G^T P}{2} - P\nabla^2 f(x) P \right] D\alpha \right\| \leq \frac{L}{2} \|D_\dagger\|\|\varepsilon\|\|D\alpha\|.$$

Injecting this result back in (39) gives the desired result,

$$\|[B-\nabla^2 f(x)]D\alpha\| \leq \|D\alpha\| \left(\frac{L\|D_\dagger\|\|\varepsilon\|}{2} + \|(I-P)\nabla^2 f(x) P\| \right)$$
$$+ M\|D\alpha\| \left\| D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2} - \mathrm{I}\gamma \right\|.$$

□

Proposition 7. *Under the assumptions of propositions 4 to 6, setting $\gamma = 0$ gives*

$$\left\| \frac{M\|D\alpha\|}{2} D\alpha + \nabla f(x_+) \right\|$$
$$\leq \frac{L}{2} \|D\alpha\|^2 + \|D\alpha\| \left(\frac{\|\varepsilon\|}{\|D\|} \left(\frac{L + M\kappa_D}{2} \right) \kappa_D + \|(I-P)\nabla^2 f(x) P\| \right). \tag{40}$$

Proof. Using propositions 4 and 5, setting $B = \tilde{H} - M\gamma P$ and $\gamma = 0$ gives

$$\begin{aligned}&\left\|\frac{M\|D\alpha\|}{2}D\alpha + \nabla f(x_+)\right\| \\ &\leq \frac{L\|D\alpha\|^2}{2} + \|D\alpha\|\left(\frac{L}{2}\|D_\dagger\|\|\varepsilon\| + \|(I-P)\nabla^2 f(x)P\| + M\left\|D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2}\right\|\right)\end{aligned}$$

Moreover,

$$\left\|D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2}\right\| \leq \frac{\|D_\dagger\|^2\|D\|\|\varepsilon\|}{2}$$

All together, and by definition of κ_D (33),

$$\begin{aligned}&\left\|\frac{M\|D\alpha\|}{2}D\alpha + \nabla f(x_+)\right\| \\ \leq &\frac{L}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right).\end{aligned}$$

□

Proposition 8. *Under the assumptions of propositions 4 to 6, setting $\gamma = \frac{\|D\alpha\|}{2}$ gives*

$$\|\nabla f(x_+)\| \leq \frac{L+M}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right). \quad (41)$$

Proof. Using propositions 4 and 5, setting $B = \tilde{H} - M\gamma \mathrm{I}$ and $\gamma = \frac{\|D\alpha\|}{2}$ gives

$$\begin{aligned}&\|\nabla f(x_+)\| \\ &\leq \frac{L\|D\alpha\|^2}{2} + \|D\alpha\|\left(\frac{L}{2}\|D_\dagger\|\|\varepsilon\| + \|(I-P)\nabla^2 f(x)P\| + M\left\|D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2} - \frac{\|D\alpha\|}{2}P\right\|\right)\end{aligned}$$

Moreover,

$$\left\|D_\dagger^T D_\dagger \frac{\|D\|\|\varepsilon\|}{2} - \frac{\|D\alpha\|}{2}P\right\| \leq \frac{\|D_\dagger\|^2\|D\|\|\varepsilon\|}{2} + \frac{\|D\alpha\|}{2}$$

All together, and by definition of κ_D (33),

$$\|\nabla f(x_+)\| \leq \frac{L+M}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right)$$

□

Proposition 9. *Let Assumption 1 and Requirements 1b to 3 hold. Then, $\forall y \in \mathbb{R}^d$, algorithm 1 ensures*

$$f(x_+) \leq f(y) + \frac{M+L}{6}\|y-x\|^3 + \frac{\|y-x\|^2}{2}\left(\|\nabla^2 f(x) - P\nabla^2 f(x)P\| + \delta\frac{L\kappa + M\kappa^2}{2}\right)$$

Proof. The output of algorithm 1 ensures that

$$\begin{aligned}&f(x_+) \leq \\ &\min_\alpha f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}(D\alpha)^T\nabla^2 f(x)D\alpha + \frac{1}{2}\alpha^T\left(H - D^T\nabla^2 f(x)D\right)\alpha + \frac{M}{6}\|D\alpha\|^3\end{aligned}$$

However, by the definition of H (10),

$$\begin{aligned}&\frac{1}{2}\alpha^T\left(H - D^T\nabla^2 f(x)D\right)\alpha \\ \leq &\frac{1}{2}\left(\alpha^T\left(\frac{G^TD + D^TG}{2} - D^T\nabla^2 f(x)D\right)\alpha + \|\alpha\|^2\frac{M\|D\|\|\varepsilon\|}{2}\right) \\ \leq &\frac{1}{2}\left(\alpha^T\left(\frac{G^TD + D^TG}{2} - D^T\nabla^2 f(x)D\right)\alpha + \|D^\dagger\|^2\|D\alpha\|\frac{M\|D\|\|\varepsilon\|}{2}\right) \\ = &\frac{1}{2}\left((D\alpha)^T\left(G - \nabla^2 f(x)D\right)\alpha + \|D^\dagger\|^2\|D\alpha\|\frac{M\|D\|\|\varepsilon\|}{2}\right).\end{aligned}$$

The last equality comes from the fact that

$$\alpha^T \left(D^T G\right) \alpha = \alpha^T \left(\frac{D^T G + G^T D}{2} + \frac{D^T G - G^T D}{2}\right) \alpha = \alpha^T \left(\frac{D^T G + G^T D}{2}\right) \alpha.$$

Now, using (8) with $w = D\alpha$ gives

$$\frac{1}{2}\alpha^T \left(H - D^T \nabla^2 f(x) D\right) \alpha \leq \frac{L\|D\alpha\|}{4} \sum_{i=1}^{N} |\alpha_i|\varepsilon_i + \|D^\dagger\|^2 \|D\alpha\| \frac{M\|D\|\|\varepsilon\|}{4}.$$

Finally, since

$$\sum_{i=1}^{N} |\alpha_i|\varepsilon_i \leq \|\alpha\|\|\varepsilon\| \leq \|D^\dagger\|\|D\alpha\|\|\varepsilon\|,$$

the inequality becomes

$$\begin{aligned}\frac{1}{2}\alpha^T \left(H - D^T \nabla^2 f(x) D\right) \alpha &\leq \frac{\|D\alpha\|^2}{4} \left(L\|D^\dagger\|\|\varepsilon\| + M\|D^\dagger\|^2\|D\|\|\varepsilon\|\right) \\ &= \frac{\|D\alpha\|^2}{4} \frac{\|\varepsilon\|}{\|D\|} \left(L\kappa_D + M\kappa_D^2\right).\end{aligned}$$

All together,

$$\begin{aligned}&f(x_+) \\ \leq &\min_\alpha f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}(D\alpha)^T \nabla^2 f(x) D\alpha + \frac{1}{2}\alpha^T \left(H - D^T \nabla^2 f(x) D\right) \alpha + \frac{M}{6}\|D\alpha\|^3 \\ \leq &\min_\alpha f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}(D\alpha)^T \nabla^2 f(x) D\alpha + \frac{\|D\alpha\|^2}{4} \frac{\|\varepsilon\|}{\|D\|} \left(L\kappa_D + M\kappa_D^2\right) + \frac{M}{6}\|D\alpha\|^3\end{aligned}$$

Now, by Requirement 3, for all y, one can find α such that

$$D\alpha = P(y - x) = DD^\dagger(y - x).$$

Indeed, multiplying both sides by $D^\dagger$ gives

$$\alpha = D^\dagger(y - x).$$

Therefore, the minimum can be written as a function of y instead of α,

$$\begin{aligned}f(x_+) \leq \min_{y \in \mathbb{R}^d} \; &f(x) + \nabla f(x)^T P(y - x) + \frac{1}{2}(P(y - x))^T \nabla^2 f(x) P(y - x) \\ &+ \frac{\|P(y - x)\|^2}{4} \frac{\|\varepsilon\|}{\|D\|} \left(L\kappa_D + M\kappa_D^2\right) + \frac{M}{6}\|P(y - x)\|^3. \end{aligned} \tag{42}$$

Since $P\nabla f(x) = \nabla f(x)$ by Requirement 1b, and using the crude bound $\|P(y - x)\| \leq \|y - x\|$,

$$\begin{aligned}f(x_+) \leq \min_{y \in \mathbb{R}^d} \; &f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x) \\ &+ \frac{1}{2}(y - x)\left[\nabla^2 f(x) - P\nabla^2 f(x) P\right](y - x) \\ &+ \frac{\|y - x\|^2}{4} \frac{\|\varepsilon\|}{\|D\|} \left(L\kappa_D + M\kappa_D^2\right) + \frac{M}{6}\|y - x\|^3.\end{aligned}$$

Using the lower bound (6),

$$f(x) + \nabla f(x)^T (y - x) + \frac{1}{2}(y - x)^T \nabla^2 f(x)(y - x) - \frac{L}{6}\|y - x\|^3 \leq f(y),$$

the crude bound $(y - x)\left[\nabla^2 f(x) - P\nabla^2 f(x) P\right](y - x) \leq \|\nabla^2 f(x) - P\nabla^2 f(x) P\|\|y - x\|^2$, and Requirements 2 and 3 lead to the desired result,

$$f(x_+) \leq f(y) + \frac{M + L}{6}\|y - x\|^3 + \frac{\|y - x\|^2}{2}\left(\|\nabla^2 f(x) - P\nabla^2 f(x) P\| + \delta\frac{L\kappa + M\kappa^2}{2}\right)$$

□

Proposition 10. *Let Assumption 1 and Requirements 1a, 2 and 3 hold. Then, $\forall y \in \mathbb{R}^d$, algorithm 1 ensures*

$$\mathbb{E}f(x_+) \leq \left(1 - \frac{N}{d}\right) f(x) + \frac{N}{d} f(y) + \frac{N}{d}\frac{(M+L)}{6}\|y-x\|^3 + \frac{N}{d}\frac{\|y-x\|^2}{2}\left(\delta\frac{L\kappa + M\kappa^2}{2} + \frac{(d-N)}{d}\|\nabla^2 f(x)\|\right)$$

Proof. The proof is the same as for proposition 9, until equation (42),

$$f(x_+) \leq \min_{y\in\mathbb{R}^d} \; f(x) + \nabla f(x)^T P(y-x) + \frac{1}{2}(P(y-x))^T\nabla^2 f(x)P(y-x) + \frac{\|P(y-x)\|^2}{4}\frac{\|\varepsilon\|}{\|D\|}\left(L\kappa_D + M\kappa_D^2\right) + \frac{M}{6}\|P(y-x)\|^3.$$

With Requirement 1a, the following relations hold (see [35, lemma 5.7])

$$\mathbb{E}[\|P(y-x)\|^2] = (y-x)^T\mathbb{E}[P](y-x) = \frac{N}{d}\|y-x\|^2, \tag{43}$$

$$\mathbb{E}[\|P(y-x)\|^3] \leq \mathbb{E}[\|P(y-x)\|^2]\|y-x\| = \frac{N}{d}\|y-x\|^2, \tag{44}$$

$$\mathbb{E}[(y-x)^T P\nabla^2 f(x)P(y-x)] \leq \frac{N^2}{d^2}(y-x)\nabla^2 f(x)(y-x) + \frac{N(d-N)}{d^2}\|\nabla^2 f(x)\|\|y-x\|^2 \tag{45}$$

Hence, removing the minimum and taking the expectation of (42) gives

$$\mathbb{E}f(x_+) \leq f(x) + \frac{N}{d}\nabla f(x)^T(y-x) + \frac{1}{2}\left(\frac{N^2}{d^2}(y-x)\nabla^2 f(x)(y-x) + \frac{N(d-N)}{d^2}\|\nabla^2 f(x)\|\|y-x\|^2\right) + \frac{N}{d}\frac{\|y-x\|^2}{4}\frac{\|\varepsilon\|}{\|D\|}\left(L\kappa_D + M\kappa_D^2\right) + \frac{N}{d}\frac{M}{6}\|y-x\|^3.$$

Using the lower bound from (6)

$$\frac{1}{2}(y-x)\nabla^2 f(x)(y-x) \leq f(y) + \frac{L}{6}\|y-x\|^3 - f(x) - \nabla f(x)(y-x)$$

in the inequality over the expectation gives

$$\mathbb{E}f(x_+) \leq f(x) + \frac{N}{d}\nabla f(x)^T(y-x) + \frac{N^2}{d^2}\left(f(y) + \frac{L}{6}\|y-x\|^3 - f(x) - \nabla f(x)(y-x)\right) + \frac{1}{2}\frac{N(d-N)}{d^2}\|\nabla^2 f(x)\|\|y-x\|^2 + \frac{N}{d}\frac{\|y-x\|^2}{4}\frac{\|\varepsilon\|}{\|D\|}\left(L\kappa_D + M\kappa_D^2\right) + \frac{N}{d}\frac{M}{6}\|y-x\|^3.$$

After simplification,

$$\mathbb{E}f(x_+) \leq \left(1 - \frac{N^2}{d^2}\right) f(x) + \frac{N^2}{d^2} f(y) + \frac{N}{d}\left(1 - \frac{N}{d}\right)\nabla f(x)^T(y-x) + \frac{1}{2}\frac{N(d-N)}{d^2}\|\nabla^2 f(x)\|\|y-x\|^2 + \frac{N}{d}\frac{\|y-x\|^2}{4}\frac{\|\varepsilon\|}{\|D\|}\left(L\kappa_D + M\kappa_D^2\right) + \left(\frac{N^2 L}{6d^2} + \frac{NM}{6d}\right)\|y-x\|^3.$$

To simplify the expression, since $N \leq d$,

$$\left(\frac{N^2 L}{6d^2} + \frac{NM}{6d}\right)\|y-x\|^3 \leq \frac{N(M+L)}{6d}\|y-x\|^3.$$

Finally, since the function is convex,

$$\frac{N}{d}\left(1-\frac{N}{d}\right)\nabla f(x)^T(y-x) \leq \frac{N}{d}\left(1-\frac{N}{d}\right)(f(y)-f(x)).$$

From this last relation, Requirement 2 and Requirement 3 comes the desired result,

$$\begin{aligned}\mathbb{E}f(x_+) \leq &\left(1-\frac{N}{d}\right)f(x) + \frac{N}{d}f(y) + \frac{N(M+L)}{6d}\|y-x\|^3 \\ &+ \frac{\|y-x\|^2}{2}\left(\frac{N}{d}\delta\frac{L\kappa + M\kappa^2}{2} + \frac{N(d-N)}{d^2}\|\nabla^2 f(x)\|\right)\end{aligned}$$

□

Proposition 11. *Under the assumptions of propositions 4 to 6, setting*

$$\gamma \geq \frac{1}{4}\frac{\|\varepsilon\|}{\|D\|}\left(1+\kappa_D^2\right)$$

gives

$$\|M\left(\gamma + \frac{\|D\alpha\|}{2}\right)D\alpha + \nabla f(x_+)\| \tag{46}$$

$$\leq \|D\alpha\|\left(\frac{L}{2}\|D\alpha\| + \frac{L}{2}\frac{\|\varepsilon\|}{\|D\|}\kappa_D + \|(I-P)\nabla^2 f(x)P\| + M\left(\gamma - \frac{\|\varepsilon\|}{2\|D\|}\right)\right) \tag{47}$$

Proof. Using propositions 4 to 6, setting $B = \tilde{H} - M\gamma P$ gives

$$\begin{aligned}&\|M\left(\gamma + \frac{\|D\alpha\|}{2}\right)D\alpha + \nabla f(x_+)\| \\ \leq &\frac{L}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{L}{2}\|D_\dagger\|\|\varepsilon\| + \|(I-P)\nabla^2 f(x)P\| + M\left\|D_\dagger^T D_\dagger\frac{\|D\|\|\varepsilon\|}{2} - \mathrm{I}\gamma\right\|\right)\end{aligned}$$

It remains to bound the last term,

$$\left\|D_\dagger^T D_\dagger\frac{\|D\|\|\varepsilon\|}{2} - P\gamma\right\| = \left\|D(D^TD)^{-\frac{1}{2}}\left((D^TD)^{-1}\frac{\|D\|\|\varepsilon\|}{2} - \gamma\right)(D^TD)^{-\frac{1}{2}}D^T\right\|.$$

Since the smallest and largest eigenvalue of $(D^TD)^{-1}$ are $\frac{1}{\sigma_{\max}^2(D)}$, $\frac{1}{\sigma_{\min}^2(D)}$ the norm can be explicitly bounded as follow:

$$\left\|D_\dagger^T D_\dagger\frac{\|D\|\|\varepsilon\|}{2} - P\gamma\right\| \leq \max\left\{\frac{\|D\|\|\varepsilon\|}{2\sigma_{\min}^2(D)} - \gamma\ ;\ \gamma - \frac{\|D\|\|\varepsilon\|}{2\sigma_{\max}^2(D)}\right\}$$

Setting γ such that the maximum is attained at the right-hand-side, i.e.,

$$\gamma \geq \frac{\sigma_{\min}^{-2}(D) + \sigma_{\max}^{-2}(D)}{4}\|D\|\|\varepsilon\| = \frac{\kappa_D^2+1}{4}\frac{\|\varepsilon\|}{\|D\|},$$

simplifies the bound into

$$\left\|D_\dagger^T D_\dagger\frac{\|D\|\|\varepsilon\|}{2} - P\gamma\right\| \leq \gamma - \frac{\|\varepsilon\|}{2\|D\|}.$$

The last step consist in replacing $\|D_\dagger\|$ by $\frac{\kappa_D}{\|D\|}$. □

F.2 Missing proofs from Section 2

Theorem 1. *Let the function f satisfy Assumption 1. Let x_+ be defined as in (4) and the matrices D, G be defined as in (3) and vector ε as in (7). Then, for all $w \in \mathbb{R}^d$ and $\alpha \in \mathbb{R}^N$,*

$$-\tfrac{L\|w\|}{2}\textstyle\sum_{i=1}^N |\alpha_i|\varepsilon_i \leq w^T(\nabla^2 f(x)D - G)\alpha \leq \tfrac{L\|w\|}{2}\sum_{i=1}^N |\alpha_i|\varepsilon_i, \tag{8}$$

$$\|w^T(\nabla^2 f(x)D - G)\| \leq \tfrac{L\|w\|}{2}\|\varepsilon\|. \tag{9}$$

Proof. Using Cauchy-Schwartz with (5) gives that, for all v,

$$v^T\left(\nabla f(y) - \nabla f(z) - \nabla^2 f(z)(y-z)\right) \leq \frac{L\|v\|}{2}\|y-z\|^2.$$

Let $v = v_i$, $y = y_i$, and $z = z_i$. By the definition of Y, Z, D, G in (3),

$$v_i^T\left(r_i - \nabla^2 f(z_i)d_i\right) \leq \frac{L\|v_i\|}{2}\|d_i\|^2.$$

Introducing $\nabla^2 f(x)$ gives

$$v_i^T\left(r_i - \nabla^2 f(z_i)d_i\right) = v_i^T\left(r_i - \nabla^2 f(x)d_i\right) + v_i^T(\nabla^2 f(z_i) - \nabla^2 f(x))d_i.$$

Since the Hessian is L-Lipchitz-continuous Assumption 1, $(\nabla^2 f(z_i) - \nabla^2 f(x))d_i \leq L\|d_i\|\|z_i - x\|$. Therefore, by the definition of ε_i,

$$v_i^T\left(r_i - \nabla^2 f(x)d_i\right) \leq \frac{L\|v_i\|\varepsilon_i}{2}. \tag{48}$$

Let $v_i = \operatorname{sign}(\alpha_i)w$. Summing all inequalities multiplied by $|\alpha_i|$ gives the first desired result:

$$w^T\left(G - \nabla^2 f(x)D\right)\alpha \leq \frac{L\|w\|\sum_{i=1}^N \varepsilon_i|\alpha_i|}{2}.$$

The second result is rather straightforward, since (48) with $v_i = w$ gives

$$w^T\left(r_i - \nabla^2 f(x)d_i\right) \leq \frac{L\|w\|\varepsilon_i}{2}.$$

Therefore,

$$\sqrt{\sum_{i=1}^N \left(w^T\left(r_i - \nabla^2 f(x)d_i\right)\right)^2} \leq \|w\|\sqrt{\sum_{i=1}^N \left(r_i - \nabla^2 f(x)d_i\right)^2} \leq \|w\|\sqrt{\sum_{i=1}^N L\varepsilon_i^2} \leq \frac{L\|w\|\|\varepsilon\|}{2}.$$

□

Theorem 2. *Let the function f satisfy Assumption 1. Let x_+ be defined as in (4), the matrices D, G be defined as in (3) and ε be defined as in (7). Then, for all $\alpha \in \mathbb{R}^N$,*

$$f(x_+) \leq f(x) + \nabla f(x)^T D\alpha + \tfrac{\alpha^T H\alpha}{2} + \tfrac{L\|D\alpha\|^3}{6}, \quad H \overset{def}{=} \tfrac{G^T D + D^T G + \mathrm{I}L\|D\|\|\varepsilon\|}{2} \tag{10}$$

$$\|\nabla f(x_+)\| \leq \|\nabla f(x) + G\alpha\| + \tfrac{L}{2}\Big(\textstyle\sum_{i=1}^N |\alpha_i|\varepsilon_i + \|D\alpha\|^2\Big), \tag{11}$$

Proof. The inequality (11) is a direct consequence of (5) (with $y = x_+$, $z = x$) combined with (9),

$$\begin{aligned}
&\|\nabla f(x_+) - \nabla f(x) - \nabla^2 f(x)D\alpha\| \leq \frac{L}{2}\|D\alpha\|^2 \\
\Leftrightarrow\ & w^T\left(\nabla f(x_+) - \nabla f(x) - \nabla^2 f(x)D\alpha\right) \leq \frac{L\|w\|}{2}\|D\alpha\|^2 \\
\Leftrightarrow\ & w^T\nabla f(x_+) \leq \frac{L}{2}\|D\alpha\|^2 + w^T\left(\nabla f(x) + \nabla^2 f(x)D\alpha\right) \\
\Leftrightarrow\ & w^T\nabla f(x_+) \overset{(8)}{\leq} \frac{L\|w\|}{2}\left(\|D\alpha\|^2 + \sum_{i=1}^N |\alpha_i|\varepsilon_i\right) + w^T\left(\nabla f(x) + G\alpha\right) \\
\Leftrightarrow\ & w^T\nabla f(x_+) \leq \|w\|\left(\frac{L}{2}\left(\|D\alpha\|^2 + \sum_{i=1}^N |\alpha_i|\varepsilon_i\right) + \|\nabla f(x) + G\alpha\|\right)
\end{aligned}$$

Setting $w = \nabla f(x_+)$ gives (11).

The inequality (10) instead comes from (6) combined with (9). Indeed,

$$\begin{aligned} f(x_+) &\leq f(x) + \nabla f(x) D\alpha + \frac{1}{2}(D\alpha)^T \nabla^2 f(x)(D\alpha) + \frac{L}{6}\|D\alpha\|^3 \\ &\overset{(9)}{\leq} f(x) + \nabla f(x) D\alpha + \frac{1}{2}\left((D\alpha)^T G\alpha + \frac{L\|D\alpha\|}{2} \sum_{i=1}^{N} |\alpha_i|\varepsilon_i \right) + \frac{L}{6}\|D\alpha\|^3 \end{aligned}$$

It remains to use the followings bounds:

$$\sum_{i=1}^{N} |\alpha_i|\varepsilon_i = \alpha^T(\operatorname{sign}(\alpha) \odot \varepsilon) \leq \|\alpha\|\|\varepsilon\|,$$
$$\|D\alpha\| \leq \|D\|\|\alpha\|.$$

All together,

$$f(x_+) \leq f(x) + \nabla f(x) D\alpha + \frac{1}{2}(D\alpha)^T G\alpha + \frac{L}{4}\|\alpha\|^2\|D\|\|\varepsilon\| + \frac{L}{6}\|D\alpha\|^3$$

Finally, since $(D\alpha)^T G\alpha$ is a quadratic form, only the symmetric counterpart of $D^T G$ counts. That means, writing $H = \frac{D^T G + G^T D}{2} + \mathrm{I}\frac{L}{2}\|D\|\|\varepsilon\|$ gives the desired result,

$$f(x_+) \leq f(x) + \nabla f(x) D\alpha + \frac{\alpha^T H \alpha}{2} + \frac{L}{6}\|D\alpha\|^3.$$

□

F.3 Missing proofs from Section 3

Theorem 3. *Let f satisfy Assumption 1. Then, at each iteration $t \geq 0$, algorithm 3 achieves*

$$f(x_{t+1}) \leq f(x_t) - \tfrac{M_{t+1}}{12}\|x_{t+1} - x_t\|^3, \quad M_{t+1} < \max\left\{2L\,;\, \tfrac{M_0}{2^t}\right\}. \tag{13}$$

Proof. Using (35), at each iteration, after the while loop, the first-order condition of the subroutine algorithm 1 reads

$$D_t^T \nabla f(x_t) + H_t \alpha_{t+1} + \frac{M_{t+1}}{2} D_t^T D_t \alpha_{t+1} \|D_t \alpha_{t+1}\| = 0. \tag{49}$$

The subscript t is dropped for clarity. After multiplying by α,

$$\nabla f(x_t)^T D\alpha + \alpha^T H \alpha + \frac{M}{2}\|D\alpha\|^3 = 0.$$

In addition, multiplying both times by α the second-order condition (36) gives

$$\alpha^T H \alpha \geq -\frac{M}{2}\|D\alpha\|^3.$$

which gives, after replacing it in (49),

$$\nabla f(x_t)^T D\alpha \leq -\frac{M}{2}\|D\alpha\|^3 + \frac{M}{2}\|D\alpha\|^3 = 0. \tag{50}$$

Injecting eqs. (49) and (50) into the while condition of algorithm 1 gives the desired result:

$$\begin{aligned} f(x_+) &\leq f(x) + \nabla f(x)^T D\alpha + \frac{1}{2}\alpha^T H \alpha + \frac{M\|D\alpha\|^3}{6}, \qquad (51) \\ &= f(x) - \frac{1}{2}\nabla f(x)^T D\alpha - \frac{M\|D\alpha\|^3}{12} \\ &\leq f(x) - \frac{M\|D\alpha\|^3}{12}. \end{aligned}$$

Where (51) is guaranteed if $M > L$. Therefore, in the worst case, $M < 2L$. □

Theorem 4. *Let f satisfy Assumption 1, and assume that f is bounded below by f^*. Let Requirements 1b to 3 hold, and $M_t \geq M_{\min}$. Then, algorithm 3 starting at x_0 with M_0 achieves*

$$\min_{i=1,\dots,t} \|\nabla f(x_i)\| \leq \max\left\{ \frac{3L}{t^{2/3}} \left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{2/3} \; ; \; \left(\frac{C_1}{t^{1/3}}\right)\left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{1/3} \right\},$$

where $C_1 = \delta L\left(\frac{\kappa+2\kappa^2}{2}\right) + \max_{i\in[0,t]} \|(I-P_i)\nabla^2 f(x_i)P_i\|$.

Proof. The starting inequality is (41):

$$\|\nabla f(x_+)\| \leq \frac{L+M}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right).$$

The result is obtained by decomposing the inequality using a maximum,

$$\begin{aligned}&\|\nabla f(x_+)\| \\ &\leq \max\left\{(L+M)\|D\alpha\|^2 \; ; \; 2\|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right)\right\}.\end{aligned}$$

In the first case,

$$\|D\alpha\| \geq \sqrt{\frac{\|\nabla f(x_+)\|}{L+M}}, \tag{52}$$

while in the second case,

$$\|D\alpha\| \geq \frac{\|\nabla f(x_+)\|}{\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|}.$$

Let C_t be defined as

$$C_t = \frac{\|\varepsilon_t\|}{\|D_t\|}\left(\frac{L+M_{t+1}\kappa_{D_t}}{2}\right)\kappa_{D_t} + \|(I-P_t)\nabla^2 f(x_t)P_t\|.$$

Then, using Requirements 2 and 3, and since $M < 2L$ by Theorem 3,

$$C_t \leq C = \delta L\left(\frac{1+2\kappa}{2}\right)\kappa + \max_t \|(I-P_t)\nabla^2 f(x_t)P_t\|$$

Therefore,

$$\|D\alpha\| \geq \frac{\|\nabla f(x_+)\|}{C}. \tag{53}$$

At each iteration t, combining eqs. (52) and (53) into Theorem 3 gives

$$f(x_t) - f(x_{t+1}) \geq \frac{M_{t+1}}{12}\|\underbrace{x_{t+1}-x_t}_{=D_t\alpha_t}\|^3 \geq \frac{M_{t+1}}{12}\min\left\{\left(\frac{\|\nabla f(x_+)\|}{L+M_{t+1}}\right)^{3/2} \; ; \; \left(\frac{\|\nabla f(x_+)\|}{C}\right)^3\right\}$$

Therefore,

$$\begin{aligned}
f(x_0)-f^\star &\geq f(x_0)-f(x_t) \\
&= \sum_{i=0}^{t-1} f(x_i) - f(x_{i+1}) \\
&\geq \sum_{i=0}^{t-1}\left(\frac{M_{i+1}}{12}\|x_{i+1}-x_i\|^3\right) \\
&\geq \sum_{i=0}^{t-1}\min_t \frac{M_{i+1}}{12}\left\{\left(\frac{\|\nabla f(x_{i+1})\|}{L+M_{i+1}}\right)^{3/2} \; ; \; \left(\frac{\|\nabla f(x_{i+1})\|}{C}\right)^3\right\} \\
&\geq t\min_{i\in[0,t-1]} \frac{M_{i+1}}{12}\min\left\{\left(\frac{\|\nabla f(x_{i+1})\|}{L+M_{i+1}}\right)^{3/2} \; ; \; \left(\frac{\|\nabla f(x_{i+1})\|}{C}\right)^3\right\} \\
&\geq t\frac{M_{\min}}{12}\min\left\{\min_{i\in[1,t]}\left(\frac{\|\nabla f(x_i)\|}{3L}\right)^{3/2} \; ; \; \min_{i\in[1,t]}\left(\frac{\|\nabla f(x_i)\|}{C}\right)^3\right\}
\end{aligned}$$

After analyzing separately each case of the minimum, either

$$\left(\frac{\min_{i\in[1,t]}\|\nabla f(x_i)\|}{3L}\right)^{3/2} \leq 12\frac{f(x_0)-f^\star}{tM_{\min}} \quad \text{or} \quad \left(\frac{\min_{i\in[1,t]}\|\nabla f(x_{t+1})\|}{C}\right)^{3} \leq 12\frac{f(x_0)-f^\star}{tM_{\min}}.$$

It remains to simplify to obtain the desired result,

$$\min_{i=1\dots t}\|\nabla f(x_i)\| \leq \max\left\{\frac{3L}{t^{2/3}}\left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{2/3} \; ; \; \left(\frac{C}{t^{1/3}}\right)\left(12\frac{f(x_0)-f^\star}{M_{\min}}\right)^{1/3}\right\}.$$

□

Theorem 5. *Assume f satisfy Assumptions 1 to 3. Let Requirements 1b to 3 hold. Then, algorithm 3 starting at x_0 with M_0 achieves, for $t \geq 1$,*

$$(f(x_t)-f^\star) \leq 6\frac{f(x_t)-f^\star}{t(t+1)(t+2)} + \frac{1}{(t+1)(t+2)}\frac{L(3R)^3}{2} + \frac{1}{t+2}\frac{C_2(3R)^2}{4},$$

where $C_2 \stackrel{def}{=} \delta L\frac{\kappa+2\kappa^2}{2} + \max_{i\in[0,t]}\|\nabla^2 f(x_i) - P_i\nabla^2 f(x_i)P_i\|$.

Proof. Starting from the inequality in proposition 9,

$$f(x_{t+1}) \leq f(y) + \frac{M_{t+1}+L}{6}\|y-x_t\|^3 + \frac{\|y-x_t\|^2}{2}C_2^{(t)},$$

where

$$C_2^{(t)} = \|\nabla^2 f(x_t) - P_t\nabla^2 f(x_t)P_t\| + \delta\frac{L\kappa + M_{t+1}\kappa^2}{2},$$

and setting $y = (1-\beta_t)x_t + \beta_t x^\star$ and $f(x^\star) = f^\star$ gives

$$f(x_{t+1}) - f^\star \leq f((1-\beta_t)x_t + \beta_t x^\star) - f^\star + \frac{M_{t+1}+L}{6}\beta_t^3\|x_t - x^\star\|^3 + \frac{\beta_t^2\|x_t-x^\star\|^2}{2}C_2^{(t)}.$$

Because the function is star-convex,

$$f(x_{t+1}) - f^\star \leq (1-\beta_t)(f(x_t)-f^\star) + \frac{M_{t+1}+L}{6}\beta_t^3\|x_t - x^\star\|^3 + \frac{\beta_t^2\|x_t-x^\star\|^2}{2}C_2^{(t)}.$$

Since algorithm 1 ensure a decrease in the function value, the iterate x_t satisfies

$$x_t \in \{x : f(x \leq f(x_0))\},$$

and therefore, $\|x_t - x^\star\| \leq R$ by Assumption 2. In addition, $M < 2L$ by Theorem 3. The inequality now becomes

$$(f(x_{t+1})-f^\star) \leq (1-\beta_t)(f(x_t)-f^\star) + \beta_t^3\frac{LR^3}{2} + \beta_t^2\frac{R^2C_2^{(t)}}{2}. \tag{54}$$

Finally, since $M < 2L$, the scalar C_2^t is bounded over time by C_2:

$$C_2^{(t)} \leq C_2 \stackrel{\text{def}}{=} \delta L\frac{\kappa+2\kappa^2}{2} + \max_t\|\nabla^2 f(x_t) - P_t\nabla^2 f(x_t)P_t\|.$$

Now, let

- $B_t = \frac{t(t+1)(t+2)}{6}$,
- $b_t : B_t = B_{t-1} + b_t$, hence $b_t = \frac{t(t+1)}{2}$, and
- $\beta_t = \frac{b_{t+1}}{B_{t+1}}$.

Therefore, for $t \geq 1$,

$$1 = \frac{B_t}{B_t} = \frac{B_{t-1}}{B_t} + \frac{b_t}{B_t} = \frac{B_{t-1}}{B_t} + \beta_{t-1} \quad \Rightarrow \quad 1 - \beta_{t-1} = \frac{B_{t-1}}{B_t}.$$

Injecting those relations in (54) gives

$$(f(x_{t+1}) - f^\star) \leq \frac{B_t}{B_{t+1}}(f(x_t) - f^\star) + \left(\frac{b_{t+1}}{B_{t+1}}\right)^3 \frac{LR^3}{2} + \left(\frac{b_{t+1}}{B_{t+1}}\right)^2 \frac{R^2 C_2}{2},$$

hence the recursion

$$\begin{aligned} B_{t+1}(f(x_{t+1}) - f^\star) &\leq B_t(f(x_t) - f^\star) + \frac{b_{t+1}^3}{B_{t+1}^2}\frac{LR^3}{2} + \frac{b_{t+1}^2}{B_{t+1}}\frac{R^2C_2}{2} \\ &\leq B_0(f(x_t) - f^\star) + \sum_{i=0}^{t} \frac{b_{i+1}^3}{B_{i+1}^2}\frac{LR^3}{2} + \sum_{i=0}^{t} \frac{b_{i+1}^2}{B_{i+1}}\frac{R^2C_2}{2}. \end{aligned}$$

$$(f(x_{t+1}) - f^\star) \leq \frac{B_0}{B_{t+1}}(f(x_t) - f^\star) + \frac{\sum_{i=0}^t \frac{b_{i+1}^3}{B_{i+1}^2}}{B_{t+1}}\frac{LR^3}{2} + \frac{\sum_{i=0}^t \frac{b_{i+1}^2}{B_{i+1}}}{B_{t+1}}\frac{R^2C_2}{2}.$$

Therefore, the rate reads By the definition of b_t and B_t,

$$\begin{aligned} \frac{b_{i+1}^3}{B_{i+1}^2} &= \frac{36}{8}\frac{(i+1)^3(i+2)^3}{(i+1)^2(i+2)^2(i+3)^2} = \frac{9}{2}\frac{(i+1)(i+2)}{(i+3)^2} \leq \frac{9}{2}, \\ \frac{b_{i+1}^2}{B_{i+1}} &= \frac{6}{4}\frac{(i+1)^2(i+2)^2}{(i+1)(i+2)(i+3)} = \frac{3}{2}\frac{(i+2)}{(i+3)}(i+1) \leq \frac{3}{2}(i+1). \end{aligned}$$

Hence,

$$\begin{aligned} \frac{\sum_{i=0}^t \frac{b_{i+1}^3}{B_{i+1}^2}}{B_{t+1}} &\leq \frac{\frac{9}{2}(t+1)}{\frac{(t+1)(t+2)(t+3)}{6}} \leq \frac{27}{(t+2)(t+3)}, \\ \frac{\sum_{i=0}^t \frac{b_{i+1}^2}{B_{i+1}}}{B_{t+1}} &\leq \frac{\sum_{i=0}^t \frac{3}{2}(i+1)}{\frac{(t+1)(t+2)(t+3)}{6}} = \frac{\frac{3}{4}(t+2)(t+1)}{\frac{(t+1)(t+2)(t+3)}{6}} = \frac{9}{2(t+3)}. \end{aligned}$$

Shifting from $t+1$ tp t gives the desired result,

$$(f(x_t) - f^\star) \leq 6\frac{f(x_t) - f^\star}{t(t+1)(t+2)} + \frac{1}{(t+1)(t+2)}\frac{L(3R)^3}{2} + \frac{1}{t+2}\frac{C_2(3R)^2}{4}.$$

□

Theorem 6. *Assume f satisfy Assumptions 1, 2 and 4. Let Requirements 1a, 2 and 3 hold. Then, in expectation over the matrices D_i, algorithm 3 starting at x_0 with M_0 achieves, for $t \geq 1$,*

$$\mathbb{E}_{D_t}[f(x_t) - f^\star] \leq \frac{1}{1 + \frac{1}{4}\left[\frac{N}{d}t\right]^3}(f(x_0) - f^\star) + \frac{1}{\left[\frac{N}{d}t\right]^2}\frac{L(3R)^3}{2} + \frac{1}{\left[\frac{N}{d}t\right]}\frac{C_3(3R)^2}{2},$$

where $C_3 \overset{def}{=} \delta L\frac{\kappa+2\kappa^2}{2} + \frac{(d-N)}{d}\max_{i\in[0,t]}\|\nabla^2 f(x_i)\|$.

Proof. The proof technique is similar to [35]. Starting from proposition 10 with $x = x_t$,

$$\begin{aligned} \mathbb{E}f(x_{t+1}) \leq &\left(1 - \frac{N}{d}\right) f(x_t) + \frac{N}{d}f(y) + \frac{N}{d}\frac{(M_{t+1}+L)}{6}\|y - x_t\|^3 \\ &+ \frac{N}{d}\frac{\|y-x_t\|^2}{2}\left(\delta\frac{L\kappa + M_{t+1}\kappa^2}{2} + \frac{(d-N)}{d}\|\nabla^2 f(x_t)\|\right), \end{aligned}$$

where the expectation is taken with $D_0, \ldots, D_{t-1}$ fixed. Using the inequality $M_{t+1} \leq 2L$ gives

$$\mathbb{E}f(x_{t+1}) \leq \left(1 - \frac{N}{d}\right) f(x_t) + \frac{N}{d}\left(f(y) + \frac{\|y-x_t\|^2}{2}C_3 + \frac{L}{2}\|y - x_t\|^3\right)$$

where

$$C_3 \stackrel{\text{def}}{=} \left(\delta L \frac{\kappa + 2\kappa^2}{2} + \frac{(d-N)}{d} \max_{i\in[0,t]} \|\nabla^2 f(x_i)\|\right).$$

Let $y = \beta_t x^\star + (1-\beta_t)x_t$, $\beta_t \in [0,1]$. After using Assumption 4 and Assumption 2,

$$\begin{aligned}
\mathbb{E}f(x_{t+1}) &\leq \left(1 - \frac{N}{d}\right) f(x_t) + \frac{N}{d}\left(f\Big(\beta_t x^\star + (1-\beta_t)x_t\Big) + \beta_t^2 \frac{C_3 R^2}{2} + \beta_t^3 \frac{LR^3}{2}\right) \\
&\leq \left(1 - \frac{N}{d}\right) f(x_t) + \frac{N}{d}\left(\beta_t f(x^\star) + (1-\beta_t) f(x_t) + \beta_t^2 \frac{C_3 R^2}{2} + \beta_t^3 \frac{LR^3}{2}\right) \\
&= \left(1 - \frac{N}{d}\right) f(x_t) + \frac{N}{d}\left(\beta_t f(x^\star) + (1-\beta_t) f(x_t) + \beta_t^2 \frac{C_3 R^2}{2} + \beta_t^3 \frac{LR^3}{2}\right), \\
&= \left(1 - \beta_t \frac{N}{d}\right) f(x_t) + \frac{N}{d}\left(\beta_t f(x^\star) + \beta_t^2 \frac{C_3 R^2}{2} + \beta_t^3 \frac{LR^3}{2}\right).
\end{aligned}$$

Hence, the recursion

$$(\mathbb{E}f(x_{t+1}) - f^\star) \leq \left(1 - \beta_t \frac{N}{d}\right)(f(x_t) - f^\star) + \frac{N}{d}\left(\beta_t^2 \frac{C_3 R^2}{2} + \beta_t^3 \frac{LR^3}{2}\right).$$

Now, define

$$\begin{aligned}
&b_t = t^2, \\
&B_t = B_0 + \sum_{i=0}^{t} b_i, \;\; B_0 = \frac{4}{3}\left(\frac{d}{N}\right)^3 \\
&\beta_t = \frac{d}{N}\frac{b_{t+1}}{B_{t+1}} \;\; \Rightarrow \; 1 - \frac{N}{d}\beta_t = \frac{B_t}{B_{t+1}}.
\end{aligned}$$

Replacing those relations in the recursion gives

$$\begin{aligned}
&B_{t+1}\left(\mathbb{E}f(x_{t+1}) - f^\star\right) \\
\leq &B_t(f(x_t) - f^\star) + \frac{N}{dB_{t+1}}\left(\left(\frac{d}{N}\frac{b_{t+1}}{B_{t+1}}\right)^2 \frac{C_3 R^2}{2} + \left(\frac{d}{N}\frac{b_{t+1}}{B_{t+1}}\right)^3 \frac{LR^3}{2}\right) \\
= &B_t(f(x_t) - f^\star) + \frac{d}{N}\frac{b_{t+1}^2}{B_{t+1}}\frac{C_3 R^2}{2} + \frac{d^2}{N^2}\frac{b_{t+1}^3}{B_{t+1}^2}\frac{LR^3}{2}
\end{aligned}$$

Expanding the inequality gives

$$B_{t+1}\left(\mathbb{E}f(x_{t+1}) - f^\star\right) \leq B_0(f(x_0) - f^\star) + \frac{d}{N}\sum_{t=0}^{t+1}\frac{b_{i+1}^2}{B_{i+1}}\frac{C_3 R^2}{2} + \frac{d^2}{N^2}\sum_{t=0}^{t+1}\frac{b_{i+1}^3}{B_{i+1}^2}\frac{LR^3}{2}$$

Since

$$\begin{aligned}
B_t = B_0 + \sum_{i=1}^{t} \geq B_0 + \int_0^t x^2 \,\mathrm{d}x = B_0 + \frac{t^3}{3} \\
\sum_{i=0}^{t}\frac{b_t^2}{B_t} \leq \sum_{i=0}^{t}\frac{i^4}{B_0 + i^3/3} \leq 3t^2, \\
\sum_{i=0}^{t}\frac{b_t^3}{B_t^2} \leq \sum_{i=0}^{t}\frac{i^6}{(B_0 + i^3/3)^2} \leq 9t,
\end{aligned}$$

the bound becomes

$$B_{t+1}\left(\mathbb{E}f(x_{t+1}) - f^\star\right) \leq B_0(f(x_0) - f^\star) + \frac{d}{N}3t^2\frac{C_3 R^2}{2} + \frac{d^2}{N^2}9t\frac{LR^3}{2}$$

Dividing both sides by B_{t+1} gives

$$\mathbb{E}f(x_{t+1}) - f^\star \leq \frac{B_0}{B_0 + \frac{(t+1)^3}{3}}(f(x_0) - f^\star) + \frac{d}{N}\frac{3(t+1)^2}{B_0 + \frac{(t+1)^3}{3}}\frac{C_3R^2}{2} + \frac{d^2}{N^2}\frac{9(t+1)}{B_0 + \frac{(t+1)^3}{3}}\frac{LR^3}{2}.$$

After the following simplifications,

$$\frac{B_0}{B_0 + (t+1)^3/3} = \frac{1}{1 + \frac{(t+1)^3}{3B_0}} = \frac{1}{1 + \frac{1}{4}\left(\frac{N}{d}(t+1)\right)^3},$$

$$\frac{3(t+1)^2}{B_0 + (t+1)^3/3} = \frac{3}{B_0}\frac{(t+1)^3}{1 + \frac{(t+1)^3}{3B_0}}\frac{1}{t+1} \leq \frac{3}{B_0}3B_0\frac{1}{t+1} = \frac{9}{t+1},$$

$$\frac{9(t+1)}{B_0 + \frac{(t+1)^3}{3}} = \frac{9}{B_0}\frac{(t+1)^3}{\frac{(t+1)^3}{3B_0}}\frac{1}{(t+1)^2} \leq \frac{9}{B_0}3B_0\frac{1}{(t+1)^2} = \frac{27}{(t+1)^2},$$

the inequality finally becomes (after shifting from $t+1$ to t),

$$\mathbb{E}f(x_t) - f^\star \leq \frac{1}{1 + \frac{1}{4}\left[\frac{N}{d}t\right]^3}(f(x_0) - f^\star) + \frac{1}{\left[\frac{N}{d}t\right]^2}\frac{L(3R)^3}{2} + \frac{1}{\left[\frac{N}{d}t\right]}\frac{C_3(3R)^2}{2}.$$

□

F.4 Missing proofs from Section 4

Notations The following functions define the estimate sequence,

$$\ell_t(x) = \sum_{i=2}^{t} b_{i-1}\left(f(x_i) + \nabla f(x_i)(x - x_i)\right), \tag{55}$$

$$\phi_t(x) = f(x_1) + \ell_t(x) + \frac{\lambda_t^{(1)}}{2}\|x - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|x - x_0\|^3 \tag{56}$$

$$\Phi_t(x) = \frac{\phi_t(x)}{B_t}, \tag{57}$$

where $\lambda_t^{(1,2)}$ are non-negative and increasing, and the sequences b_t, B_t are

$$B_t = \frac{k(t+1)(t+2)}{6} = \sum_{i=1}^{t} b_i, \tag{58}$$

$$b_t = \frac{(t+1)(t+2)}{2} = B_{t+1} - B_t. \tag{59}$$

$$\tag{60}$$

Moreover, the following quantities will be important later,

$$v_t = \arg\min_x \phi_t(x) = \arg\min_x \Phi_t(x), \tag{61}$$

$$\beta_t = \frac{b_t}{B_{t+1}}, \tag{62}$$

$$y_t = (1 - \beta_t)x_t + \beta_t v_t. \tag{63}$$

F.4.1 Technical results

Lemma 1. *From [44, Lemma 4]. The Bregman divergence of the function $\|x\|^i$ satisfies, for $i \geq 2$,*

$$\|x\|^i - \|y\|^i - \nabla(\|y\|^i)(x - y) \geq \frac{1}{2^{i-2}}\|x - y\|^i.$$

Proposition 12. *The function ϕ_t is lower-bounded by*

$$\phi_t \geq \underbrace{\phi_t(v_t)}_{=\phi_t^\star} + \frac{\lambda_t^{(1)}}{2}\|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12}\|x - v_t\|^3 \tag{64}$$

where $v_t = \arg\min_x \phi_t(x)$.

Proof. The first order condition on ϕ_t reads,

$$\ell_t' + \nabla\left(\frac{\lambda_t^{(1)}}{2}\|v_t - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v_t - x_0\|^3\right) = 0.$$

Multiplying both sides by $(x - v_t)$ gives

$$\ell_t'(x - v_t) + \nabla\left(\frac{\lambda_t^{(1)}}{2}\|v_t - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v_t - x_0\|^3\right)(x - v_t) = 0.$$

Note that, since ℓ_t is an affine function, $\ell_t'(x - v_t) = \ell_t(x) - \ell_t(v_t)$. Hence,

$$\ell_t(x) - \ell_t(v_t) + \nabla\left(\frac{\lambda_t^{(1)}}{2}\|v_t - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v_t - x_0\|^3\right)(x - v_t) = 0.$$

Finally, adding $\frac{\lambda_t^{(1)}}{2}\|x - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|x - x_0\|^3$ on both sides and after reorganizing the terms,

$$\phi_t(x) = \ell_t(v_t) + \frac{\lambda_t^{(1)}}{2}\|x - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|x - x_0\|^3 - \nabla\left(\frac{\lambda_t^{(1)}}{2}\|v_t - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v_t - x_0\|^3\right)(x - v_t). \tag{65}$$

From lemma 1 with $x = x - x_0$, $y = v_t - x_0$, and after reorganizing the terms,

$$\|x - x_0\|^i - \nabla(\|v_t - x_0\|^i)(x - v_t) \geq \frac{1}{2^{i-2}}\|x - v_t\|^i + \|v_t - x_0\|^i.$$

Therefore, using the previous inequality with $i = 2$ and $i = 3$, (65) becomes

$$\phi_t(x) \geq \ell_t(v_t) + \frac{\lambda_t^{(1)}}{2}\|v_t - x_0\|^2 + \frac{\lambda_t^{(2)}}{6}\|v_t - x_0\|^3 + \frac{\lambda_t^{(2)}}{2}\|v_t - x\|^2 + \frac{\lambda_t^{(3)}}{12}\|v_t - x\|^3$$

By definition of $\phi_t^\star = \phi_t(v_t)$,

$$\phi_t(x) \geq \phi_t^\star + \frac{\lambda_t^{(1)}}{2}\|v_t - x\|^2 + \frac{\lambda_t^{(2)}}{12}\|v_t - x\|^3.$$

□

Proposition 13. *Under the assumptions of proposition 11 the condition*

$$\frac{\|f(x_+)\|^2}{M\left(\gamma + \frac{\|D\alpha\|}{2}\right)} \leq -\nabla f(x)^T D\alpha$$

is guaranteed as long as γ and M are sufficiently big,

$$\gamma \geq \frac{1}{2}\frac{\|\varepsilon\|}{\|D\|}\frac{1 + \kappa_D^2}{2},$$
$$M \geq \frac{1}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I - P)\nabla^2 f(x)P\|\right).$$

Proof. Elevating to the square the inequality of proposition 11 gives

$$\left(M\left(\gamma + \frac{\|D\alpha\|}{2}\right)\right)^2\|D\alpha\|^2 + \|\nabla f(x_+)\|^2 + \left(M\left(\gamma + \frac{\|D\alpha\|}{2}\right)\right)\nabla f(x_+)^T D\alpha$$
$$\leq \|D\alpha\|^2\left(\frac{L}{2}\|D\alpha\| + \frac{L}{2}\frac{\|\varepsilon\|}{\|D\|}\kappa_D + \|(I - P)\nabla^2 f(x)P\| + M\left(\gamma - \frac{\|\varepsilon\|}{2\|D\|}\right)\right)^2.$$

The desired result holds if the following condition is satisfied,

$$\left(M\left(\gamma + \frac{\|D\alpha\|}{2}\right)\right)^2\|D\alpha\|^2$$
$$\geq \|D\alpha\|^2\left(\frac{L}{2}\|D\alpha\| + \frac{L}{2}\frac{\|\varepsilon\|}{\|D\|}\kappa_D + \|(I - P)\nabla^2 f(x)P\| + M\left(\gamma - \frac{\|\varepsilon\|}{2\|D\|}\right)\right)^2.$$

After simplification of the squares and γ,

$$M\frac{\|D\alpha\|}{2} \geq \frac{L}{2}\|D\alpha\| + \frac{L}{2}\frac{\|\varepsilon\|}{\|D\|}\kappa_D + \|(I-P)\nabla^2 f(x)P\| - M\frac{\|\varepsilon\|}{2\|D\|}.$$

Hence, the following condition is sufficient,

$$M \geq \frac{1}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right).$$

□

Proposition 14 (Guarantees that algorithm 4 terminates)**.** *Let f satisfies Assumption 1. Then, under Requirements 1b to 3, if $M_0 < M$, the output of algorithm 4 guarantees that*

$$\begin{aligned} M &\leq 2\frac{1}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right) \\ &\leq L\kappa_D + \frac{2\|(I-P)\nabla^2 f(x)P\|}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}} \\ M\left(\gamma + \frac{\|D\alpha\|}{2}\right) &\leq (1+\kappa_D^2)\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right). \end{aligned}$$

Proof. Assume $\Delta = \infty$, so that the algorithm can only terminates with `ExitFlag` equals to `SmallStep`. Either the algorithm terminates at $M = M_0$, or $M_0 < M$. In the second case, algorithm 4 multiplies M by a factor 2 while

$$\frac{\|f(x_+)\|^2}{M\left(\gamma + \frac{\|D\alpha\|}{2}\right)} \geq -\nabla f(x)^T D\alpha,$$

then, by proposition 13, once

$$M \geq \frac{1}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right)$$

holds, the condition is met and the algorithm terminates. In the worst case, M is at most two times larger than the bound:

$$M \leq 2\frac{1}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right) \tag{66}$$

Since, for all $c_1 > 0$, $c_2 > 0$, $c_3 > 1$,

$$\frac{c_1 + c_2 c_3}{c_1 + c_2} = 1 + \frac{c_2(c_3-1)}{c_1+c_2} = 1 + \frac{c_3 - 1}{\frac{c_1}{c_2} + 1} \leq c_3, \tag{67}$$

the bound becomes

$$M \leq \left(L\kappa_D + \frac{2\|(I-P)\nabla^2 f(x)P\|}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\right).$$

Going back to (66), and after multiplying both sides by $\gamma + \frac{\|D\alpha\|}{2}$ with $\gamma = \frac{1}{2}\frac{\|\varepsilon\|}{\|D\|}\frac{1+\kappa_D^2}{2}$ gives

$$M\left(\gamma + \frac{\|D\alpha\|}{2}\right) \leq 2\frac{\frac{\|D\alpha\|}{2} + \frac{1}{2}\frac{\|\varepsilon\|}{\|D\|}\frac{1+\kappa_D^2}{2}}{\frac{\|D\alpha\|}{2} + \frac{\|\varepsilon\|}{2\|D\|}}\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right).$$

Once again, using (67) gives

$$M\left(\gamma + \frac{\|D\alpha\|}{2}\right) \leq (1+\kappa_D^2)\left(\frac{L}{2}\left(\|D\alpha\| + \frac{\|\varepsilon\|}{\|D\|}\kappa_D\right) + \|(I-P)\nabla^2 f(x)P\|\right).$$

□

Proposition 15. *Assume f satisfies Assumption 1. Then, under Requirements 1b to 3, if*

$$\|D\alpha\| \geq 2\left(\delta\kappa_D^2 + 2\frac{\|(I-P)\nabla^2 f(x)P\|}{\frac{1}{\sqrt{3}-1}L}\right).$$

then the condition

$$\frac{2}{3^{3/4}}\frac{\|\nabla f(x_+)\|^{3/2}}{\sqrt{M}} \leq -\nabla f(x_+)^T D\alpha$$

is guaranteed as long as M is sufficiently big,

$$\frac{2}{\sqrt{3}-1}L \leq M.$$

Proof. Starting from proposition 7,

$$\begin{aligned}&\left\|\frac{M\|D\alpha\|}{2}D\alpha + \nabla f(x_+)\right\| \\ \leq &\frac{L}{2}\|D\alpha\|^2 + \|D\alpha\|\left(\frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|\right).\end{aligned}$$

Assuming

$$\frac{\frac{L}{\kappa_D}+M}{4}\|D\alpha\| \geq \frac{\|\varepsilon\|}{\|D\|}\left(\frac{L+M\kappa_D}{2}\right)\kappa_D + \|(I-P)\nabla^2 f(x)P\|,$$

or equivalently,

$$\|D\alpha\| \geq 4\left(\frac{\|\varepsilon\|}{2\|D\|}\kappa_D^2 + \frac{\|(I-P)\nabla^2 f(x)P\|}{\frac{L}{\kappa_D}+M}\right), \tag{68}$$

gives the simpler bound

$$\left\|\frac{M\|D\alpha\|}{2}D\alpha + \nabla f(x_+)\right\| \leq \frac{L+\frac{L}{\kappa_D}+M}{4}\|D\alpha\|^2.$$

Elevating both sides to the square give

$$\frac{M^2}{4}\|D\alpha\|^4 + M\|D\alpha\|D\alpha^T\nabla f(x_+) + \|\nabla f(x_+)\|^2 \leq \frac{(L(1+\frac{1}{\kappa_D})+M)^2}{16}\|D\alpha\|^4,$$

hence, and using the fact that $\kappa_D \geq 1$,

$$M\|D\alpha\|D\alpha^T\nabla f(x_+) + \|\nabla f(x_+)\|^2 \leq \frac{(2L+M)^2 - 4M^2}{16}\|D\alpha\|^4.$$

Assuming $\frac{2}{\sqrt{3}-1}L \leq M$,

$$M\|D\alpha\|D\alpha^T\nabla f(x_+) + \|\nabla f(x_+)\|^2 \leq \frac{-M^2}{16}\|D\alpha\|^4.$$

After reorganization, and writing $r = \|D\alpha\|$,

$$\frac{M}{16}r^3 + \frac{\|\nabla f(x_+)\|^2}{Mr} \leq -D\alpha^T\nabla f(x_+).$$

Using

$$\frac{c_1}{r} + c_2 r^3 \geq 4c_2^{1/4}\left(\frac{c_1}{3}\right)^{3/4},$$

the inequality becomes

$$\begin{aligned}-D\alpha^T\nabla f(x_+) &\geq \frac{M^{1/4}}{2}\frac{\|\nabla f(x_+)\|^{3/2}}{M^{3/4}}\frac{4}{3^{3/4}} \\ &= \frac{2}{3^{3/4}}\frac{\|\nabla f(x_+)\|^{3/2}}{\sqrt{M}}.\end{aligned}$$

Finally, the condition on $\|D\alpha\|$ in (68) is made stronger by replacing M with its lower bound, by using Requirements 1b to 3 and because $\kappa_D \geq 1$, i.e.,

$$\|D\alpha\| \geq 4\left(\frac{\delta\kappa_D^2}{2} + \frac{\|(I-P)\nabla^2 f(x)P\|}{\frac{2}{\sqrt{3}-1}L}\right).$$

□

Proposition 16. *If $\lambda_t^{(1)}$ and $\lambda_t^{(2)}$ satisfy*

$$\lambda_t^{(1)} \geq \frac{b_{t+1}^2}{B_t} M_{t+1} \left(\gamma_t + \frac{\|D_t \alpha_t\|}{2} \right), \quad \lambda_t^{(2)} \geq \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} M_{t+1}$$

Then, the function ϕ satisfies

$$B_t f(x_t) \leq \phi_t(x), \qquad \phi_t(x) \leq B_t f(x) + \frac{\lambda_t^{(1)} + \tilde{\lambda}^{(1)}}{2} \|x - x_0\|^2 + \frac{\lambda_t^{(2)} + \tilde{\lambda}^{(2)}}{6} \|x - x_0\|^3,$$

where

$$\tilde{\lambda}^{(1)} = \|\nabla f(x_0) - P_0 \nabla f(x_0) P_0\| + \delta \left(\frac{L\kappa + M_1 \kappa^2}{2} \right), \quad \tilde{\lambda}^{(2)} = M_1 + L.$$

Proof. The result is proven by recursion. At $t = 1$, the condition $B_t f(x_t) \leq \phi_t(x)$ is obviously satisfied since

$$f(x_1) \leq \min_v \phi_1(v) = f(x_1).$$

On the other hand, by proposition 9,

$$\begin{aligned} f(x_1) &\leq \min_x f(x) + \frac{\tilde{\lambda}^{(2)}}{6} \|x - x_0\|^3 + \frac{\tilde{\lambda}^{(1)}}{2} \|x - x_0\|^2 \\ &\leq f(x) + \frac{\tilde{\lambda}^{(2)}}{6} \|x - x_0\|^3 + \frac{\tilde{\lambda}^{(1)}}{2} \|x - x_0\|^2. \end{aligned}$$

Therefore, the second condition holds by definition of ϕ,

$$\begin{aligned} \phi_t &= f(x_1) + \frac{\lambda_t^{(1)}}{2} \|x - x_0\|^2 + \frac{\lambda_t^{(2)}}{6} \|x - x_0\|^3 \\ &\leq \frac{\lambda_1^{(1)} + \tilde{\lambda}^{(1)}}{2} \|x - x_0\|^2 + \frac{\lambda_1^{(2)} + \tilde{\lambda}^{(2)}}{6} \|x - x_0\|^3. \end{aligned}$$

Now, assume $t > 1$, and $B_t f(x_t) \leq \phi_t(x)$. Hence,

$$\begin{aligned} &\min_x \phi_{t+1}(x) \\ &= \min_x \ell_t(x) + b_t \left[f(x_{t+1}) + \nabla f(x_t)(x - x_{t+1}) \right] + \frac{\lambda_{t+1}^{(1)}}{2} \|x - x_0\|^2 + \frac{\lambda_{t+1}^{(2)}}{6} \|x - x_0\|^3 \\ &= \min_x \phi_t(x) + b_t \left[f(x_{t+1}) + \nabla f(x_{t+1})(x - x_{t+1}) \right] \\ &\quad + \frac{\lambda_{t+1}^{(1)} - \lambda_t^{(1)}}{2} \|x - x_0\|^2 + \frac{\lambda_{t+1}^{(2)} - \lambda_t^{(2)}}{6} \|x - x_0\|^3 \\ &\geq \min_x \phi_t(x) + b_t \left[f(x_{t+1}) + \nabla f(x_{t+1})(x - x_{t+1}) \right] \\ &\overset{(64)}{\geq} \min_x \phi_t^\star + \frac{\lambda_t^{(1)}}{2} \|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12} \|x - v_t\|^3 + b_t \left[f(x_{t+1}) + \nabla f(x_{t+1})(x - x_{t+1}) \right] \\ &\geq \min_x B_t f(x_t) + \frac{\lambda_t^{(1)}}{2} \|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12} \|x - v_t\|^3 + b_t \left[f(x_{t+1}) + \nabla f(x_{t+1})(x - x_{t+1}) \right] \\ &\overset{\text{A.4}}{\geq} \min_x B_t f(x_{t+1}) + \nabla f(x_{t+1})(x_t - x_{t+1}) + b_t \left[f(x_{t+1}) + \nabla f(x_{t+1})(x - x_{t+1}) \right] \\ &\quad + \frac{\lambda_t^{(1)}}{2} \|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12} \|x - v_t\|^3 \\ &= \min_x B_{t+1} f(x_{t+1}) + \nabla f(x_{t+1})(B_t x_t + b_t x - B_{t+1} x_{t+1}) + \frac{\lambda_t^{(1)}}{2} \|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12} \|x - v_t\|^3 \\ &\overset{(63)}{=} \min_x B_{t+1} f(x_{t+1}) + B_{t+1} \nabla f(x_{t+1})(y_t - x_{t+1}) \\ &\quad + b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(1)}}{2} \|x - v_t\|^2 + \frac{\lambda_t^{(2)}}{12} \|x - v_t\|^3 \end{aligned}$$

The inequality is satisfied if either

$$\textbf{(a)} \quad 0 \leq B_{t+1}\nabla f(x_{t+1})(y_t - x_{t+1}) + b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(2)}}{12}\|x - v_t\|^3, \quad \text{or}$$

$$\textbf{(b)} \quad 0 \leq B_{t+1}\nabla f(x_{t+1})(y_t - x_{t+1}) + b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(1)}}{2}\|x - v_t\|^2.$$

It remains now to find *sufficient condition* such that one of the previous inequalities hold.

Define x_{t+1} to be the output of algorithm 4 starting from y_t, hence $y_t - x_{t+1} = -D_t\alpha_t$. The algorithm guarantees that

$$\textbf{(b)} \quad -\nabla f(x_{t+1})^T D_t \alpha_t \geq \frac{\|f(x_{t+1})\|^2}{M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right)}, \quad \text{or} \tag{69}$$

$$\textbf{(a)} \quad -\nabla f(x_{t+1})^T D_t \alpha_t \geq \frac{2}{3^{3/4}} \frac{\|\nabla f(x_{t+1})\|^{3/2}}{\sqrt{M_{t+1}}} \quad \text{and} \quad \|D\alpha\| \geq \Delta. \tag{70}$$

Combining the expressions **(a)** and **(b)** leads to the following sufficient conditions:

$$0 \leq B_{t+1}\frac{2}{3^{3/4}} \frac{\|\nabla f(x_{t+1})\|^{3/2}}{\sqrt{M_{t+1}}} + b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(2)}}{12}\|x - v_t\|^3, \tag{71}$$

$$0 \leq B_{t+1}\frac{\|f(x_{t+1})\|^2}{M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right)} + b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(1)}}{2}\|x - v_t\|^2. \tag{72}$$

Case 1: equation (71). Starting from the first order condition of the minimum of (71) over x,

$$b_t \nabla f(x_{t+1}) + \frac{\lambda_t^{(2)}}{4}\|x - v_t\|(x - v_t) = 0. \tag{73}$$

Multiplying (73) by $(x - v_t)$ gives

$$b_t \nabla f(x_{t+1})(x - v_t) = -\frac{\lambda_t^{(2)}}{4}\|x - v_t\|^3$$

Hence, when x satisfies (73),

$$b_t \nabla f(x_{t+1})(x - v_t) + \frac{\lambda_t^{(2)}}{12}\|x - v_t\|^3 = -\frac{\lambda_t^{(2)}}{6}\|x - v_t\|^3. \tag{74}$$

Going back to (73), after isolating $x - v_t$,

$$(x - v_t) = -\frac{4b_t}{\lambda_t^{(2)}}\nabla f(x_{t+1})\frac{1}{\|x - v_t\|}$$

Therefore, after taking the norm and changing the power,

$$\begin{aligned}
\|x - v_t\|^3 &= \left(\frac{4b_t}{\lambda_t^{(2)}}\|\nabla f(x_{t+1})\|\right)^{3/2}, \\
\Leftrightarrow \frac{\lambda_t^{(2)}}{6}\|x - v_t\|^3 &= \frac{\lambda_t^{(2)}}{6}\left(\frac{4b_t}{\lambda_t^{(2)}}\|\nabla f(x_{t+1})\|\right)^{3/2} \\
&= \frac{4}{3\sqrt{\lambda_t^{(2)}}}\left(b_t\|\nabla f(x_{t+1})\|\right)^{3/2}.
\end{aligned}$$

After using (74) and injecting the minimal value makes the condition (71) stronger:

$$0 \leq B_{t+1}\frac{2}{3^{3/4}} \frac{\|\nabla f(x_{t+1})\|^{3/2}}{\sqrt{M_{t+1}}} - \frac{4}{3\sqrt{\lambda_t^{(2)}}}\left(b_t\|\nabla f(x_{t+1})\|\right)^{3/2}.$$

Hence, if $\lambda_t^{(2)}$ satisfies

$$B_{t+1}\frac{2}{3^{3/4}\sqrt{M_{t+1}}} \geq \frac{4}{3\sqrt{\lambda_t^{(2)}}}b_t^{(3/2)} \quad \Leftrightarrow \quad \lambda_t^{(2)} \geq \frac{4}{\sqrt{3}}\frac{b_t^3}{B_{t+1}^2}M_{t+1}, \tag{75}$$

then (71) is satisfied.

Case 2: equation (72). Starting from the first order condition of the minimum of (72) over x,

$$b_{t+1}\nabla f(x_{t+1}) + \lambda_t^{(1)}(x - v_t). \tag{76}$$

Hence,

$$(x - v_t) = -\frac{b_t \nabla f(x_{t+1})}{\lambda_t^{(1)}}.$$

Injecting the value back in (72) gives

$$B_{t+1}\frac{\|f(x_{t+1})\|^2}{M\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right)} - b_t^2\frac{\|\nabla f(x_{t+1})\|^2}{\lambda_t^{(1)}} + \frac{1}{2}b_t^2\frac{\|\nabla f(x_{t+1})\|^2}{\lambda_t^{(1)}}.$$

Therefore, if the following condition holds,

$$\frac{B_{t+1}}{2M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right)} \geq \frac{b_t^2}{\lambda_t^{(1)}} \quad \Leftrightarrow \quad \lambda_t^{(1)} \geq \frac{b_t^2}{2B_{t+1}}M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right),$$

then (72) is satisfied.

□

Proposition 17. *Let f satisfies Assumption 1. Then, under Requirements 1b to 3, using the re-scaling technique from algorithm 6*

$$(M_0)_{t+1} \leftarrow M_{t+1}\left(\frac{\|\varepsilon_t\|}{2\|D_t\|} + \frac{\|D_t\alpha_t\|}{2}\right),$$

makes $(M_0)_{t+1}$ bounded as follow:

$$(M_0)_{t+1} \leq \frac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1)\max_{0\leq i\leq t}\|(I - P_i)\nabla^2 f(x_i)P_i\|. \tag{77}$$

Proof. By proposition 14, for all Δ, if $M_0 \leq M_{t+1}$,

$$M_{t+1} \leq L\kappa_{D_t} + \frac{2\|(I - P_t)\nabla^2 f(x)P_t\|}{\frac{\|D_t\alpha_t\|}{2} + \frac{\|\varepsilon_t\|}{2\|D_t\|}},$$

where $(M_0)_t$ is the initial smoothness parameter in algorithm 4. The desired result comes immediately after multiplying by $\left(\frac{\|\varepsilon_t\|}{2\|D_t\|} + \frac{\|D_t\alpha_t\|}{2}\right)$, using Requirements 2 and 3 and because the re-scaling technique requires $M_0 < M_{t+1}$.

In addition, if $\|D_t\alpha_t\|$ is sufficiently large, i.e., if

$$\|D_t\alpha_t\| \geq \max\left\{\Delta\ ;\ 2\kappa_D^2\delta + 2\max_{0\leq i\leq t}\frac{\|(I - P_i)\nabla^2 f(x_i)P_i\|}{\frac{1}{\sqrt{3}-1}L}\right\},$$

then by proposition 15 the algorithm terminates when $M_{t+1} \geq \frac{2}{\sqrt{3}-1}L$. For simplicity, consider the stronger condition

$$\|D_t\alpha_t\| \geq \Delta + 2\kappa_D^2\delta + 2\max_{0\leq i\leq t}\frac{\|(I - P_i)\nabla^2 f(x_i)P_i\|}{\frac{1}{\sqrt{3}-1}L}. \tag{78}$$

Hence, $(M_0)_{t+1}$ cannot be larger than

$$\begin{aligned}
&(M_0)_{t+1} \\
&\leq \frac{L}{2}\left(\Delta + 2\kappa^2\delta + 2\max_{0\leq i\leq t}\frac{\|(I - P_i)\nabla^2 f(x_i)P_i\|}{\frac{1}{\sqrt{3}-1}L} + \delta\kappa\right) + \max_i\|(I - P_i)\nabla^2 f(x_i)P_i\|, \\
&= \frac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1)\max_{0\leq i\leq t}\|(I - P_i)\nabla^2 f(x_i)P_i\|.
\end{aligned}$$

□

Proposition 18. *Let f satisfies Assumption 1. Then, under Requirements 1b to 3, $\lambda_t^{(1)}$ and $\lambda_t^{(2)}$ in algorithm 6 are bounded by*

$$\lambda_t^{(1)} \leq 2 \cdot \frac{b_{t+1}^2}{B_t} \frac{(M_0)_{\max}^2}{L}, \tag{79}$$

$$\lambda_t^{(2)} \leq 2 \cdot \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} \max\left\{1 \; ; \; \frac{2}{\Delta}\right\} (M_0)_{\max}. \tag{80}$$

where

$$(M_0)_{\max} \overset{def}{=} \frac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1) \max_{0 \leq i \leq t} \|(I - P_i)\nabla^2 f(x_i) P_i\|. \tag{81}$$

Proof. Since algorithm 6 doubles $\lambda_t^{(1)}$, $\lambda_t^{(2)}$ until $\phi_t^\star \geq f(x_{t+1})$, then by proposition 16, both $\lambda_t^{(1)}$, $\lambda_t^{(2)}$ achieves at most

$$\lambda_t^{(1)} \leq 2 \cdot \frac{b_{t+1}^2}{B_t} M_{t+1} \left(\gamma_t + \frac{\|D_t \alpha_t\|}{2}\right), \quad \lambda_t^{(2)} \leq 2 \cdot \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} M_{t+1}.$$

There are three cases to distinguish.

First case. When $(M_0)_t = M_{t+1}$, whatever the value of `ExitFlag`, by proposition 17, and by construction of algorithm 6, $(M_0)_t$ is bounded as follow:

$$(M_0)_t \leq \max\left\{\frac{(M_0)_{t-1}}{2} \; ; \; \frac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1) \max_{0 \leq i \leq t} \|(I - P_i)\nabla^2 f(x_i) P_i\|. \right\}.$$

In the worst case, the maximum is attained in the right hand side. For simplicity, let $(M_0)_{\max}$ be defined as

$$(M_0)_{\max} \overset{\text{def}}{=} \frac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1) \max_{0 \leq i \leq t} \|(I - P_i)\nabla^2 f(x_i) P_i\|.$$

In this case, $\lambda_1^{(t)}$ and $\lambda_t^{(2)}$ are bounded by

$$\begin{aligned} \lambda_t^{(1)} &\leq 2 \cdot \frac{b_{t+1}^2}{B_t} (M_0)_{\max} \left(\gamma_t + \frac{\|D_t \alpha_t\|}{2}\right) \\ \lambda_t^{(2)} &\leq 2 \cdot \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} (M_0)_{\max}. \end{aligned} \tag{82}$$

By Requirements 2 and 3, γ_t is bounded by

$$\gamma_t = \frac{1}{2} \frac{\|\varepsilon_t\|}{\|D_t\|} \frac{1 + \kappa_{D_t}^2}{2} \leq \frac{1 + \kappa^2}{4} \delta.$$

Moreover, under the condition (78),

$$\|D_t \alpha_t\| \geq \Delta + 2\kappa_D^2 \delta + 2 \max_{0 \leq i \leq t} \frac{\|(I - P_i)\nabla^2 f(x_i) P_i\|}{\frac{1}{\sqrt{3}-1} L},$$

the algorithm algorithm 4 terminates with `ExitFlag` equals to `LargeStep`. Hence, to update $\lambda_t^{(1)}$,

$$\|D_t \alpha_t\| \leq \Delta + 2\kappa_D^2 \delta + 2 \max_{0 \leq i \leq t} \frac{\|(I - P_i)\nabla^2 f(x_i) P_i\|}{\frac{1}{\sqrt{3}-1} L} \tag{83}$$

Therefore,

$$\begin{aligned} \gamma_t + \frac{\|D_t \alpha_t\|}{2} &\leq \frac{1 + \kappa^2}{4} \delta + \frac{1}{2} \left(\Delta + 2\kappa^2 \delta + 2 \max_{0 \leq i \leq t} \frac{\|(I - P_i)\nabla^2 f(x_i) P_i\|}{\frac{1}{\sqrt{3}-1} L}\right) \\ &\leq \frac{1}{L} \left(\frac{L}{2} \left(\frac{1 + 5\kappa^2}{2} \delta + \Delta\right) + (\sqrt{3} - 1) \max_{0 \leq i \leq t} \|(I - P_i)\nabla^2 f(x_i) P_i\|\right) \\ &\leq \frac{(M_0)_{\max}}{L}. \end{aligned}$$

By consequence,

$$\begin{aligned}\lambda_t^{(1)} \leq & 2 \cdot \frac{b_{t+1}^2}{B_t}(M_0)_{\max}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right) \\ & \leq 2 \cdot \frac{b_{t+1}^2}{B_t}\frac{(M_0)_{\max}^2}{L}.\end{aligned}$$

Second case. When $M_0 \leq M_t$ and `ExitFlag` equals `SmallStep`, only λ_1 is updated. Therefore,

$$\lambda_t^{(1)} \leq 2 \cdot \frac{b_{t+1}^2}{B_t} M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right).$$

Since $M_0 \leq M_{t+1}$, by proposition 14, then by Requirements 2 and 3,

$$\begin{aligned}M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right) & \leq (1+\kappa_{D_t}^2)\left(\frac{L}{2}\left(\|D_t\alpha_t\| + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}\right) + \|(I-P_t)\nabla^2 f(x_t)P_t\|\right), \\ & \leq (1+\kappa^2)\left(\frac{L}{2}\left(\|D_t\alpha_t\| + \delta\kappa\right) + \max_{0\leq i\leq t}\|(I-P_i)\nabla^2 f(x_i)P_i\|\right)\end{aligned}$$

Because `ExitFlag` is `SmallStep`, $\|D\alpha\|$ is bounded by (83). Hence,

$$\begin{aligned}M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right) \leq & (1+\kappa^2)\left[\frac{L}{2}\left(\Delta + 2\kappa_D^2\delta + 2\max_{0\leq i\leq t}\frac{\|(I-P_i)\nabla^2 f(x_i)P_i\|}{\frac{1}{\sqrt{3}-1}L} + \delta\kappa\right)\right. \\ & \left. + \max_{0\leq i\leq t}\|(I-P_i)\nabla^2 f(x_i)P_i\|\right] \\ = & (1+\kappa^2)\left[\frac{L}{2}\left(\Delta + (2\kappa_D^2+\kappa)\delta\right) + \sqrt{3}\max_{0\leq i\leq t}\|(I-P_i)\nabla^2 f(x_i)P_i\|\right] \\ \leq & (1+\kappa^2)(M_0)_{\max}.\end{aligned}$$

Since $(1+\kappa^2) \leq \frac{(M_0)_{\max}}{L}$,

$$M_{t+1}\left(\gamma_t + \frac{\|D_t\alpha_t\|}{2}\right) \leq \frac{(M_0)_{\max}^2}{L}.$$

Hence,

$$\lambda_t^{(1)} \leq 2 \cdot \frac{b_{t+1}^2}{B_t}\frac{(M_0)_{\max}^2}{L}.$$

Third case. It remains to bound $\lambda_t^{(2)}$, when $(M_0)_t \leq M_{t+1}$ and `ExitFlag` equals `LargeStep`. In such a case, by proposition 14,

$$M_{t+1} \leq 4\frac{1}{\|D_t\alpha_t\| + \frac{\|\varepsilon_t\|}{\|D_t\|}}\left(\frac{L}{2}\left(\|D_t\alpha_t\| + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}\right) + \|(I-P_t)\nabla^2 f(x_t)P_t\|\right)$$

Note that, for all a, b, the function $\frac{x+a}{x+b}$ is decreasing as long as $b < a$. Hence, since `ExitFlag` equals `LargeStep`, $\|D_t\alpha_t\| \geq \Delta$ and

$$\frac{\|D_t\alpha_t\| + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}}{\|D_t\alpha_t\| + \frac{\|\varepsilon_t\|}{\|D_t\|}} \leq \frac{\Delta + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}}{\Delta + \frac{\|\varepsilon_t\|}{\|D_t\|}},$$

and therefore, using Requirements 2 and 3 leads to

$$\begin{aligned}M_{t+1} & \leq 4\frac{1}{\Delta + \frac{\|\varepsilon_t\|}{\|D_t\|}}\left(\frac{L}{2}\left(\Delta + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}\right) + \|(I-P_t)\nabla^2 f(x_t)P_t\|\right) \\ & \leq 4\frac{1}{\Delta}\left(\frac{L}{2}\left(\Delta + \frac{\|\varepsilon_t\|}{\|D_t\|}\kappa_{D_t}\right) + \|(I-P_t)\nabla^2 f(x_t)P_t\|\right) \\ & \leq 4\frac{1}{\Delta}\left(\frac{L}{2}\left(\Delta + \delta\kappa\right) + \max_{0\leq i\leq t}\|(I-P_i)\nabla^2 f(x_i)P_i\|\right) \\ & \leq 2\frac{(M_0)_t}{\Delta}.\end{aligned}$$

Therefore, after combining this inequality with (82),

$$\lambda_t^{(2)} \leq 2 \cdot \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} \max\left\{1\ ;\ \frac{2}{\Delta}\right\} (M_0)_{\max}.$$

□

Theorem 7. *Assume f satisfy Assumptions 1, 2 and 4. Let Requirements 1b to 3 hold. Then, algorithm 5 starting at x_0 with M_0 achieves, for all $\Delta > 0$ and for $t \geq 1$,*

$$f(x_t) - f^\star \leq \frac{(M_0)_{\max}^2}{L} \left(\frac{3R}{t+3}\right)^2 + \frac{4(M_0)_{\max}}{3\sqrt{3}} \max\left\{1\ ;\ \tfrac{2}{\Delta}\right\} \left(\frac{3R}{t+3}\right)^3 + \frac{\frac{\tilde{\lambda}^{(1)}R^2}{2} + \frac{\tilde{\lambda}^{(2)}R^3}{6}}{(t+1)^3}.$$

where $\tilde{\lambda}^{(1)} = 0.5 \cdot \delta\left(L\kappa + M_1\kappa^2\right) + \|\nabla f(x_0) - P_0 \nabla f(x_0) P_0\|, \qquad \tilde{\lambda}^{(2)} = M_1 + L,$

$$(M_0)_{\max} = \tfrac{L}{2}(2\Delta + (2\kappa^2 + \kappa)\delta) + (2\sqrt{3} - 1)\max_{0 \leq i \leq t} \|(I - P_i)\nabla^2 f(x_i) P_i\|.$$

Proof. By construction of $\phi_t(x)$, from proposition 16 and Assumption 2,

$$\begin{aligned}
B_t f(x_t) &\leq \min_x \phi_t(x) && (84)\\
&\leq \phi_t(x^\star) && (85)\\
&\leq B_t f(x^\star) + \frac{\lambda_t^{(1)} + \tilde{\lambda}^{(1)}}{2}\|x^\star - x_0\|^2 + \frac{\lambda_t^{(2)} + \tilde{\lambda}^{(2)}}{6}\|x^\star - x_0\|^3 && (86)\\
&\leq B_t f(x^\star) + \frac{\lambda_t^{(1)} + \tilde{\lambda}^{(1)}}{2}R^2 + \frac{\lambda_t^{(2)} + \tilde{\lambda}^{(2)}}{6}R^3 && (87)\\
\Rightarrow f(x_t) - f^\star &\leq \frac{\lambda_t^{(1)} + \tilde{\lambda}^{(1)}}{2B_t}R^2 + \frac{\lambda_t^{(2)} + \tilde{\lambda}^{(2)}}{6B_t}R^3. && (88)
\end{aligned}$$

By proposition 18, the following bounds holds:

$$\begin{aligned}
\lambda_t^{(1)} &\leq 2 \cdot \frac{b_{t+1}^2}{B_t} \frac{(M_0)_{\max}^2}{L},\\
\lambda_t^{(2)} &\leq 2 \cdot \frac{4}{\sqrt{3}} \frac{b_{t+1}^3}{B_t^2} \max\left\{1\ ;\ \frac{2}{\Delta}\right\} (M_0)_{\max}.
\end{aligned}$$

Hence,

$$f(x_t) - f^\star \leq \frac{2 \cdot \frac{b_{t+1}^2}{B_t}\frac{(M_0)_{\max}^2}{L} + \tilde{\lambda}^{(1)}}{2B_t}R^2 + \frac{2 \cdot \frac{4}{\sqrt{3}}\frac{b_{t+1}^3}{B_t^2}\max\left\{1\ ;\ \frac{2}{\Delta}\right\}(M_0)_{\max} + \tilde{\lambda}^{(2)}}{6B_t}R^3.$$

Since $\frac{b_{t+1}}{B_t} = \frac{3}{(t+3)}$,

$$\begin{aligned}
\frac{b_{t+1}^3}{B_t^3} &= \frac{3^3}{(t+3)^3}, && (89)\\
\frac{b_{t+1}^2}{B_t^2} &= \frac{3^2}{(t+3)^2}. && (90)\\
& && (91)
\end{aligned}$$

Therefore,

$$f(x_t) - f^\star \leq \frac{(M_0)_{\max}^2}{L} \left(\frac{3R}{t+3}\right)^2 + \frac{4(M_0)_{\max}}{3\sqrt{3}} \max\left\{1\ ;\ \frac{2}{\Delta}\right\} \left(\frac{3R}{t+3}\right)^3 + \frac{\frac{\tilde{\lambda}^{(1)}R^2}{2} + \frac{\tilde{\lambda}^{(2)}R^3}{6}}{(t+1)^3}.$$

□