A Fast Anderson-Chebyshev Acceleration for Nonlinear Optimization

Zhize Li
King Abdullah University of Science and Technology

Jian Li
Tsinghua University

Abstract

Anderson acceleration (or Anderson mixing) is an efficient acceleration method for fixed point iterations $x_{t+1} = G(x_t)$, e.g., gradient descent can be viewed as iteratively applying the operation $G(x) \triangleq x - \alpha\nabla f(x)$. It is known that Anderson acceleration is quite efficient in practice and can be viewed as an extension of Krylov subspace methods for nonlinear problems. In this paper, we show that Anderson acceleration with Chebyshev polynomial can achieve the optimal convergence rate $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$, which improves the previous result $O(\kappa\ln\frac{1}{\epsilon})$ provided by (Toth and Kelley, 2015) for quadratic functions. Moreover, we provide a convergence analysis for minimizing general nonlinear problems. Besides, if the hyperparameters (e.g., the Lipschitz smooth parameter L) are not available, we propose a *guessing algorithm* for guessing them dynamically and also prove a similar convergence rate. Finally, the experimental results demonstrate that the proposed Anderson-Chebyshev acceleration method converges significantly faster than other algorithms, e.g., vanilla gradient descent (GD), Nesterov's Accelerated GD. Also, these algorithms combined with the proposed guessing algorithm (guessing the hyperparameters dynamically) achieve much better performance.

1 Introduction

Machine learning problems are usually modeled as optimization problems, ranging from convex optimization to highly nonconvex optimization such as deep neural networks, e.g., (Nesterov, 2004; Bubeck, 2015; LeCun et al., 2015; Lei et al., 2017; Li and Li, 2018; Fang et al., 2018; Zhou et al., 2018; Li et al., 2019; Ge et al., 2019; Li, 2019). To solve an optimization problem $\min_x f(x)$, the classical method is gradient descent, i.e., $x_{t+1} = x_t - \alpha_t\nabla f(x_t)$. There exist several techniques to accelerate the standard gradient descent, e.g., momentum (Nesterov, 2004; Allen-Zhu, 2017; Lan and Zhou, 2018; Lan et al., 2019). There are also various vector sequence acceleration methods developed in the numerical analysis literature, e.g., (Brezinski, 2000; Sidi et al., 1986; Smith et al., 1987; Brezinski and Redivo Zaglia, 1991; Brezinski et al., 2018). Roughly speaking, if a vector sequence converges very slowly to its limit, then one may apply such methods to accelerate the convergence of this sequence. Taking gradient descent as an example, the vector sequence are generated by $x_{t+1} = G(x_t) \triangleq x_t - \alpha_t\nabla f(x_t)$, where the limit is the fixed-point $G(x^*) = x^*$ (i.e. $\nabla f(x^*) = 0$). One notable advantage of such acceleration methods is that they usually do not require to know how the vector sequence is actually generated. Thus the applicability of those methods is very wide.

Recently, Scieur et al. (2016) used the minimal polynomial extrapolation (MPE) method (Smith et al., 1987) for convergence acceleration. This is a nice example of using sequence acceleration methods to optimization problems. In this paper, we are interested in another classical sequence acceleration method called *Anderson acceleration* (or *Anderson mixing*), which was proposed by Anderson in 1965 (Anderson, 1965). The method is known to be quite efficient in a variety of applications (Capehart, 1989; Pratapa et al., 2016; Higham and Strabić, 2016; Loffeld and Woodward, 2016). The idea of Anderson acceleration is to maintain m recent iterations for determining the next iteration point, where m is a parameter (typically a very small constant). Thus, it can be viewed as an extension of the existing momentum methods which usually use the last and current points to determine the next iteration point. Anderson acceleration with slight modifications is described in Algorithm 1.

Proceedings of the 23[rd]International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

Algorithm 1: Anderson Acceleration(m)

1 **input:** x_0, T, λ, β_t
2 Define $G(x) \triangleq x + F \triangleq x - \lambda \nabla f(x)$;
3 $x_1 = G(x_0)$, $F_0 = G(x_0) - x_0$;
4 **for** $t = 1, 2, \dots T$ **do**
5 $\quad m_t = \min\{m, t\}$;
6 $\quad F_t \triangleq G(x_t) - x_t$;
7 $\quad$ Solve $\min_{\alpha^t = (\alpha_0^t, \dots, \alpha_{m_t}^t)^T} \|\sum_{i=0}^{m_t} \alpha_i^t F_{t-i}\|_2$ subject to $\sum_{i=0}^{m_t} \alpha_i^t = 1$;
8 $\quad x_{t+1} = (1-\beta_t) \sum_{i=0}^{m_t} \alpha_i^t x_{t-i} + \beta_t \sum_{i=0}^{m_t} \alpha_i^t G(x_{t-i})$;
9 **return** x_T

Note that the step in Line 7 of Algorithm 1 can be transformed to an equivalent unconstrained least-squares problem:

$$\min_{(\alpha_1^t, \dots, \alpha_{m_t}^t)^T} \left\| F_t - \sum_{i=1}^{m_t} \alpha_i^t (F_t - F_{t-i}) \right\|_2, \qquad (1)$$

then let $\alpha_0^t = 1 - \sum_{i=1}^{m_t} \alpha_i^t$. Using QR decomposition, (1) can be solved in time $2m_t^2 d$, where d is the dimension. Moreover, the QR decomposition of (1) at iteration t can be efficiently obtained from that of at iteration $t-1$ in $O(m_t d)$ (see, e.g. (Golub and Van Loan, 1996)). The constant $m_t \leq m$ is usually very small. We use $m = 3$ and 5 for the numerical experiments in Section 5. Hence, each iteration of Anderson acceleration can be implemented quite efficiently.

Many studies showed the relations between Anderson acceleration and other optimization methods. In particular, for the quadratic case (linear problems), Walker and Ni (2011) showed that it is related to the well-known Krylov subspace method GMRES (generalized minimal residual algorithm) (Saad and Schultz, 1986). Furthermore, Potra and Engler (2013) showed that GMRES is equivalent to Anderson acceleration with any mixing parameters under $m = \infty$ (see Line 5 of Algorithm 1) for linear problems. Concretely, Toth and Kelley (2015) proved the first linear convergence rate $O(\kappa \ln \frac{1}{\epsilon})$ for linear problems with fixed parameter β, where κ is the condition number. Besides, Eyert (1996), and Fang and Saad (2009) showed that Anderson acceleration is related to the multisecant quasi-Newton methods (more concretely, the generalized Broyden's second method). Despite the above results, the convergence results for this efficient method are still limited (especially for general nonlinear problems and the case where m is small). In this paper, we analyze the convergence for small m which is the typical case in practice and also provide the convergence analysis for general nonlinear problems.

1.1 Our Contributions

There has been a growing number of applications of Anderson acceleration method (Pratapa et al., 2016; Higham and Strabić, 2016; Loffeld and Woodward, 2016; Scieur et al., 2018). Towards a better understanding of this efficient method, we make the following technical contributions:

1. We prove the optimal $O(\sqrt{\kappa} \ln \frac{1}{\epsilon})$ convergence rate of the proposed Anderson-Chebyshev acceleration (i.e., Anderson acceleration with Chebyshev polynomial) for minimizing quadratic functions (see Theorem 1). Our result improves the previous result $O(\kappa \ln \frac{1}{\epsilon})$ given by (Toth and Kelley, 2015) and matches the lower bound $\Omega(\sqrt{\kappa} \ln \frac{1}{\epsilon})$ provided by (Nesterov, 2004). Note that for ill-conditioned problems, the condition number κ can be very large.

2. Then, we prove the linear-quadratic convergence of Anderson acceleration for minimizing general nonlinear problems under some standard assumptions (see Theorem 2). Compared with Newton-like methods, it is more attractive since it does not require to compute (or approximate) Hessians, or Hessian-vector products.

3. Besides, we propose a *guessing algorithm* (Algorithm 2) for the case when the hyperparameters (e.g., μ, L) are not available. We prove that it achieves a similar convergence rate $O(\sqrt{\kappa} \ln \frac{1}{\epsilon} + \sqrt{\kappa}(\ln \kappa \ln B)^2)$ (see Theorem 3). This guessing algorithm can also be combined with other algorithms, e.g., Gradient Descent (GD), Nesterov's Accelerated GD (NAGD). The experimental results (see Section 5.1) show that these algorithms combined with the proposed guessing algorithm achieve much better performance.

4. Finally, the experimental results on the real-world UCI datasets and synthetic datasets demonstrate that Anderson acceleration methods converge significantly faster than other algorithms (see Section 5). Combined with our theoretical results, the experiments validate that Anderson acceleration methods (especially Anderson-Chebyshev acceleration) are efficient both in theory and practice.

1.2 Related Work

As aforementioned, Anderson acceleration can be viewed as the extension of the momentum methods (e.g., NAGD) and the potential extension of Krylov subspace methods (e.g., GMRES) for nonlinear problems. In particular, GD is the special case of Anderson

acceleration with $m = 0$, and to some extent NAGD can be viewed as $m = 1$. We also review the equivalence of GMRES and Anderson acceleration without truncation (i.e., $m = \infty$) in Appendix A[1]. Besides, Eyert (1996), and Fang and Saad (2009) showed that Anderson acceleration is related to the multisecant quasi-Newton methods. Note that Anderson acceleration has the advantage over the Newton-like methods since it does not require the computation of Hessians or approximation of Hessians or Hessian-vector products.

There are many sequence acceleration methods in the numerical analysis literatures. In particular, the well-known Aitken's Δ^2 process (Aitken, 1926) accelerated the convergence of a sequence that is converging linearly. Shanks generalized the Aitken extrapolation which was known as Shanks transformation (Shanks, 1955). Recently, Brezinski et al. (2018) proposed a general framework for Shanks sequence transformations which includes many vector sequence acceleration methods. One fundamental difference between Anderson acceleration and other sequence acceleration methods (such as MPE, RRE (reduced rank extrapolation) (Sidi et al., 1986; Smith et al., 1987), etc.) is that Anderson acceleration is a fully dynamic method (Capehart, 1989). Here *dynamic* means all iterations are in the same sequence, and it does not require to restart the procedure. It can be seen from Algorithm 1 that all iterations are applied to the same sequence $\{x_t\}$. In fact, in Capehart's PhD thesis (Capehart, 1989), several experiments were conducted to demonstrate the superior performance of Anderson acceleration over other semi-dynamic methods such as MPE, RRE (semi-dynamic means that the algorithm maintains more than one sequences or needs to restart several times). More recently, Anderson acceleration with different variants and/or under different assumptions are widely studied (see e.g., (Zhang et al., 2018; Evans et al., 2018; Scieur et al., 2019)).

2 The Quadratic Case

In this section, we consider the problem of minimizing a quadratic function (also called least squares, or ridge regression (Boyd and Vandenberghe, 2004; Hoerl and Kennard, 1970)). The formulation of the problem is

$$\min_{x\in\mathbb{R}^d} f(x) = \frac{1}{2}x^T Ax - b^T x, \tag{2}$$

where $\mu I_d \preceq \nabla^2 f = A \preceq L I_d$. Note that μ and L are usually called the strongly convex parameter and Lipschitz continuous gradient parameter, respectively (e.g. (Nesterov, 2004; Allen-Zhu, 2017; Lan et al., 2019)). There are many algorithms for optimizing this type of functions. See e.g. (Bubeck, 2015) for more details. We analyze the problem of minimizing a more general function $f(x)$ in the next Section 3.

We prove that Anderson acceleration with Chebyshev polynomial parameters $\{\beta_t\}$ achieves the optimal convergence rate, i.e., it obtains an ϵ-approximate solution using $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$ iterations. The convergence result is stated in the following Theorem 1.

Theorem 1 *The Anderson-Chebyshev acceleration method achieves the optimal convergence rate $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$ for obtaining an ϵ-approximate solution of problem (2) for any $0 \le m \le k$, where $\kappa = L/\mu$ is the condition number, k is defined in Definition 1 and this method combines Anderson acceleration (Algorithm 1) with the Chebyshev polynomial parameters $\beta_t = 1/\big(\frac{L+\mu}{2} + \frac{L-\mu}{2}\cos(\frac{(2t-1)\pi}{2T})\big)$, for $t = 1, 2, \ldots, T$.*

Remark: In this quadratic case, we mention that Toth and Kelley (2015) proved the first convergence rate $O(\kappa\ln\frac{1}{\epsilon})$ for fixed parameter β. Here we use the Chebyshev polynomials to improve the result to the optimal $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$ which matches the lower bound $\Omega(\sqrt{\kappa}\ln\frac{1}{\epsilon})$. Note that for ill-conditioned problems, the condition number κ can be very large. Also note that in practice the constant m is usually very small. Particularly, $m = 3$ has already achieved a remarkable performance from our experimental results (see Figures 2–5 in Section 5).

Before proving Theorem 1, we first define k and then briefly review some properties of the Chebyshev polynomials. We refer to (Rivlin, 1974; Olshanskii and Tyrtyshnikov, 2014; Hageman and Young, 2012) for more details of Chebyshev polynomials.

Definition 1 *Let v_i's be the unit eigenvectors of A, where A is defined in (2). Consider a unit vector $c \triangleq \sum_{i=1}^d c_i v_i$ and let $c' \triangleq \mathrm{Proj}_{B_k^\perp} c = \sum_{i=1}^d c_i' v_i$, where $\mathrm{Proj}_{B_k^\perp}$ denotes the projection to the orthogonal complement of the column space of $B_k \triangleq A[x_{t-k} - x_t, \ldots, x_{t-1} - x_t] \in \mathbb{R}^{d\times k}$. Define k to be the maximum integer such that $c_i' \le (1 + \frac{1}{\sqrt{\kappa}+1})c_i$ for any $i \in [d]$.*

Obviously, $k \ge 0$ since $c' = c$ due to $B_0 = 0$ and $\mathrm{Proj}_{B_0^\perp} = I$.

Now we review the Chebyshev polynomials. The *Chebyshev polynomials* are polynomials $P_k(x)$, where $k \ge 0$, $\deg(P_k) = k$, which is defined by the recursive relation:

$$\begin{aligned} &P_0(x) = 1, \\ &P_1(x) = x, \\ &P_{k+1}(x) = 2xP_k(x) - P_{k-1}(x). \end{aligned} \tag{3}$$

[1] All appendices are provided in the Supplementary Material.

The key property is that $P_k(x)$ has minimal deviation from 0 on $[-1,1]$ among all polynomials Q_k with $\deg(Q_k)=k$ and coefficient $\alpha_k=2^{k-1}$ for the largest degree term x^k, i.e.,

$$\max_{x\in[-1,1]}|P_k(x)|\le\max_{x\in[-1,1]}|Q_k(x)|\quad\text{for all } Q_k. \tag{4}$$

In particular, for $|x|\le 1$, Chebyshev polynomials can be written in an equivalent way:

$$P_k(x)=\cos(k\arccos x). \tag{5}$$

In our proof, we use this equivalent form (5) instead of (3). The equivalence can be verified as follows:

$$\begin{aligned}P_k(x)&=2x\cos((k-1)\arccos x)-\cos((k-2)\arccos x)\\&=2\cos\theta\cos((k-1)\theta)-\cos((k-2)\theta) &(6)\\&=\cos(k\theta)+\cos((k-2)\theta)-\cos((k-2)\theta)\\&=\cos(k\arccos x), &(7)\end{aligned}$$

where (6) and (7) use the transformation $x=\cos\theta$ due to $|x|\le 1$. According to (5), $\max_{x\in[-1,1]}|P_k(x)|=1$ and the k roots of P_k are as follows:

$$x_i=\cos\Big(\frac{(2i-1)\pi}{2k}\Big),\ i=1,2,\dots,k. \tag{8}$$

To demonstrate it more clearly, we provide an example for $P_4(x)$ (W-shape curve) in Figure 1. Since $k=4$ in this polynomial $P_4(x)$, the first root $x_1=\cos\left(\frac{(2i-1)\pi}{2k}\right)=\cos\left(\frac{\pi}{8}\right)\approx 0.92$. The remaining three roots for $P_4(x)$ can be easily computed too.

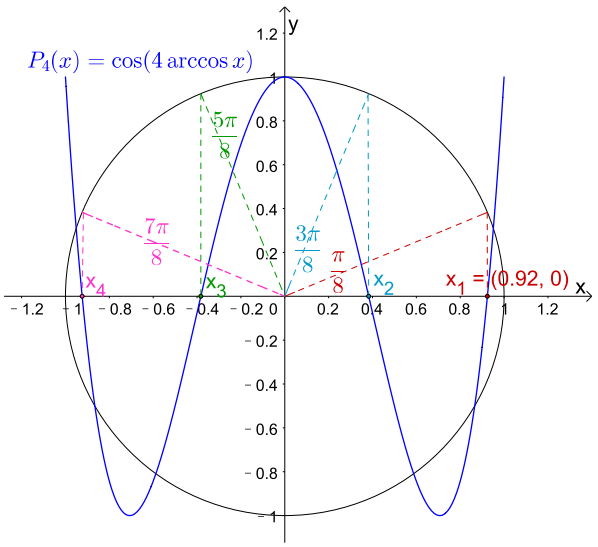


Figure 1: The Chebyshev polynomial $P_4(x)$

Proof of Theorem 1. For iteration $t+1$, the residual $F_{t+1}\triangleq-\lambda\nabla f(x_{t+1})=-(Ax_{t+1}-b)$ (let $\lambda=1$) can be deduced as follows:

$$\begin{aligned}F_{t+1}&=b-Ax_{t+1}\\&=b-A\Big[(1-\beta_t)\sum_{i=0}^{m_t}\alpha_i^t x_{t-i}+\beta_t\sum_{i=0}^{m_t}\alpha_i^t G(x_{t-i})\Big]\\&=b-A\Big[\sum_{i=0}^{m_t}\alpha_i^t x_{t-i}+\beta_t\sum_{i=0}^{m_t}\alpha_i^t\left(b-Ax_{t-i}\right)\Big] &(9)\\&=b-\beta_tAb-A\Big[\sum_{i=0}^{m_t}\alpha_i^t\left((I-\beta_tA)x_{t-i}\right)\Big]\\&=(I-\beta_tA)\sum_{i=0}^{m_t}\alpha_i^t\left(b-Ax_{t-i}\right)\\&=(I-\beta_tA)\sum_{i=0}^{m_t}\alpha_i^t F_{t-i}, &(10)\end{aligned}$$

where (9) uses $G(x_t)=x_t+F_t$.

To bound $\|F_{t+1}\|_2$ (i.e., $\|\nabla f(x_{t+1})\|_2$), we first obtain the following lemma by using Singular Value Decomposition (SVD) to solve the least squares problem (1) and then using several transformations. We defer the proof of Lemma 1 to Appendix B.2.

Lemma 1 *Let $F_1=b-Ax_1$ and $F_{t+1}=b-Ax_{t+1}$, then*

$$\|F_{t+1}\|_2/\|F_1\|_2\le\sqrt{2\min_{\beta}\max_{\lambda\in[\mu,L]}|H_t(\lambda)|} \tag{11}$$

where $H_t(\lambda)=(1-\beta_t\lambda)\cdots(1-\beta_1\lambda)$ is a degree t polynomial.

According to Lemma 1, to bound $\|F_{t+1}\|_2$, it is sufficient to bound the right-hand-side (RHS) of (11) (i.e., $\min_\beta\max_{\lambda\in[\mu,L]}|H_t(\lambda)|$). So we want to choose parameter β in order to make $\max_{\lambda\in[\mu,L]}|H_t(\lambda)|$ as small as possible. According to (4) (the minimal deviation property of standard Chebyshev polynomials), hence a natural idea is to choose β such that $H_t(\lambda)=(1-\beta_t\lambda)\cdots(1-\beta_1\lambda)$ is a kind of modified Chebyshev polynomials. In order to do this, we first transform $[\mu,L]$ into $[-1,1]$, i.e., let $\lambda=\frac{L+\mu}{2}+\frac{L-\mu}{2}x$, where $x\in[-1,1]$. Also note that polynomial $H_t(\lambda)=(1-\beta_t\lambda)\cdots(1-\beta_1\lambda)$ has (only) one constraint, i.e., $H_t(0)=1$. Thus we choose β such that

$$\begin{aligned}H_t(\lambda)&=P_t\Big(\frac{2\lambda-(L+\mu)}{L-\mu}\Big)\Big/P_t\Big(-\frac{L+\mu}{L-\mu}\Big)\\&=P_t(x)/P_t\Big(-\frac{L+\mu}{L-\mu}\Big), &(12)\end{aligned}$$

where $P_t(\cdot)$ is the standard Chebyshev polynomials.

Now, the RHS of (11) can be bounded as follows:

$$\begin{aligned}&\min_{\beta}\max_{\lambda\in[\mu,L]}|H_t(\lambda)|\\ &\le \max_{x\in[-1,1]}\Big|P_t(x)/P_t\Big(-\frac{L+\mu}{L-\mu}\Big)\Big| \qquad (13)\\ &\le 1\Big/\Big|P_t\Big(-\frac{L+\mu}{L-\mu}\Big)\Big|, \qquad (14)\end{aligned}$$

where (13) uses (12), and (14) uses $\max_{x\in[-1,1]}|P_t(x)| = 1$ (see (5)). According to (8), it is not hard to see that $H_t(\lambda)$ is defined by the mixing parameters $\beta_i = 1\Big/\Big(\frac{L+\mu}{2}+\frac{L-\mu}{2}\cos\big(\frac{(2i-1)\pi}{2t}\big)\Big)$ according to $\lambda = \frac{L+\mu}{2}+\frac{L-\mu}{2}x$, where $i = 1, 2, \ldots, t$. Note that the roots of standard Chebyshev polynomials (i.e., (8)) can be found from many textbooks, e.g., Section 1.2 of (Rivlin, 1974). Now, we only need to bound $\big|P_t\big(-\frac{L+\mu}{L-\mu}\big)\big|$. First, we need to transform the form (5) of Chebyshev polynomials $P_t(x)$ as follows:

$$\begin{aligned}P_t(x) &= \cos(t\arccos x)\\ &= \cos(t\theta) \qquad \text{Define } x \triangleq \cos\theta\\ &= \big(e^{i\theta t}+e^{-i\theta t}\big)/2\\ &= \big((\cos\theta+i\sin\theta)^t+(\cos\theta-i\sin\theta)^t\big)/2\\ &= \Big(\big(x+\sqrt{x^2-1}\big)^t+\big(x-\sqrt{x^2-1}\big)^t\Big)/2.\end{aligned}$$

Let $x = -\frac{L+\mu}{L-\mu}$, we get $\sqrt{x^2-1} = \sqrt{\frac{(L+\mu)^2-(L-\mu)^2}{(L-\mu)^2}} = \sqrt{\frac{4L\mu}{(L-\mu)^2}} = \frac{2\sqrt{L\mu}}{L-\mu}$. So we have

$$\begin{aligned}\Big|P_t\Big(-\frac{L+\mu}{L-\mu}\Big)\Big| &\ge \frac{1}{2}\Big(\frac{L+\mu}{L-\mu}+\frac{2\sqrt{L\mu}}{L-\mu}\Big)^t\\ &\ge \frac{1}{2}\Big(\frac{\sqrt{L}+\sqrt{\mu}}{\sqrt{L}-\sqrt{\mu}}\Big)^t. \qquad (15)\end{aligned}$$

Now, the RHS of (11) can be bounded as

$$\begin{aligned}\sqrt{2\min_{\beta}\max_{\lambda\in[\mu,L]}|H_t(\lambda)|} &\le \sqrt{2\Big/\Big|P_t\Big(-\frac{L+\mu}{L-\mu}\Big)\Big|} \qquad (16)\\ &\le 2\Big(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\Big)^{t/2}, \qquad (17)\end{aligned}$$

where (16) follows from (14), and (17) follows from (15). Then, according to (11), the gradient norm is bounded as $\|\nabla f(x_{t+1})\|_2 = \|F_{t+1}\|_2 \le 2\big(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\big)^{t/2}\|F_1\|_2 = 2\big(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\big)^{t/2}\|\nabla f(x_1)\|_2$, where $\kappa = L/\mu$. Note that if the number of iterations $t = (\sqrt{\kappa}+1)\ln\frac{1}{\epsilon}$, then

$$\big(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\big)^{t/2} = \big(1-\frac{2}{\sqrt{\kappa}+1}\big)^{t/2} \le \epsilon.$$

Thus the Anderson-Chebyshev acceleration method achieves the optimal convergence rate $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$ for obtaining an ϵ-approximate solution. □

3 The General Case

In this section, we analyze the Anderson Acceleration (Algorithm 1) in the general nonlinear case:

$$\min_{x\in\mathbb{R}^d} f(x). \qquad (18)$$

We prove that Anderson acceleration method achieves the linear-quadratic convergence rate under the following standard Assumptions 1 and 2, where $\|\cdot\|$ denotes the Euclidean norm. Let $\mathcal{B}_t$ denote the small matrix of the least-square problem in Line 7 of Algorithm 1, i.e., $\mathcal{B}_t \triangleq [F_t - F_{t-1}, \ldots, F_t - F_{t-m}] \in \mathbb{R}^{d\times m}$ (see problem (1)). Then, we define its condition number $\kappa_t \triangleq \|\nabla f(x_t)\|/\tilde{\mu}_t$ and $\tilde{\kappa} \triangleq \max_t\{\kappa_t\}$, where $\tilde{\mu}_t$ denotes the least non-zero singular value of $\mathcal{B}_t$.

Assumption 1 *The Hessian $\nabla^2 f$ satisfies $\mu \le \|\nabla^2 f\| \le L$, where $0 \le \mu \le L$.*

Assumption 2 *The Hessian $\nabla^2 f$ is γ-Lipschitz continuous, i.e.,*

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \le \gamma\|x-y\|. \qquad (19)$$

Theorem 2 *Suppose Assumption 1 and 2 hold. Let step-size $\lambda = \frac{2}{L+\mu}$. The convergence rate of Anderson Acceleration(m) (Algorithm 1) is linear-quadratic for problem (18), i.e.,*

$$\|\nabla f(x_{t+1})\| \le c_1\Delta_t^2 + c_2\Delta_t\|\nabla f(x_t)\| + (1-c_3)\|\nabla f(x_t)\|, \qquad (20)$$

where $c_1 = \frac{3\tilde{\kappa}^2\gamma m}{(L+\mu)^2}$, $c_2 = \frac{2\tilde{\kappa}\beta_t\gamma\sqrt{m}}{(L+\mu)^2}$, $c_3 = \beta_t\frac{2\mu}{L+\mu}$ and $\Delta_t \triangleq \max_{i\in[m]}\|x_t - x_{t-i}\|$.

Remark:

1. The constant $m \ge 0$ is usually very small. Particularly, we use $m = 3$ and 5 for the numerical experiments in Section 5. Hence Δ_t is very small and also decreases as the algorithm converges.

2. Besides, one can also use $\|\nabla f(x_t)\|$ instead of Δ_t in (20) according to the property of f (Assumption 1), i.e., $\mu\|x_t - x^*\| \le \|\nabla f(x_t) - \nabla f(x^*)\| = \|\nabla f(x_t)\|$, and $\|x_t - x_{t-i}\| = \|x_t - x^* + x^* - x_{t-i}\| \le \|x_t - x^*\| + \|x_{t-i} - x^*\|$.

3. Note that the first two terms in RHS of (20) converge quadratically and the last term converges linearly. Due to the fully dynamic property of Anderson acceleration as we discussed in Section 1.2, it turns out the exact convergence rate of Anderson acceleration in the general case is not easy to obtain. But we note that the convergence rate is roughly linear, i.e., $O(\frac{1}{c_3}\log\frac{1}{\epsilon})$ since the first two quadratic terms converge much faster

than the last linear term in some neighborhood of optimum. In particular, if f is a quadratic function, then $\gamma = 0$ (Assumption 2) and thus $c_1 = c_2 = 0$ in (20). Only the last linear term remained, thus it converges linearly (see the following corollary).

Corollary 1 *If f is a quadratic function, let step-size $\lambda = \frac{2}{L+\mu}$ and $\beta_t = 1$. Then the convergence rate of Anderson Acceleration is linear, i.e., $O(\kappa \ln \frac{1}{\epsilon})$, where $\kappa = L/\mu$ is the condition number.*

Note that this corollary recovers the previous result (i.e., $O(\kappa \ln \frac{1}{\epsilon})$) obtained by (Toth and Kelley, 2015), and we use *Chebyshev polyniomial* to improve this result to the optimal convergence rate $O(\sqrt{\kappa} \ln \frac{1}{\epsilon})$ in our previous Section 2 (see Theorem 1). Concretely, we transfer the weight of step-size λ to the parameters β_t's and use Chebyshev polynomial parameters β_t's in our Theorem 1 instead of using fixed parameter $\beta \equiv 1$.

Now, we provide a proof sketch for Theorem 2. The detailed proof can be found in Appendix B.1.

Proof Sketch of Theorem 2. Consider the iteration $t+1$, we have $F_t = -\frac{2}{L+\mu}\nabla f(x_t)$ according to $\lambda = \frac{2}{L+\mu}$. First, we need to demonstrate several useful forms of x_{t+1} as follows:

$$\begin{aligned} x_{t+1} &= (1-\beta_t)\sum_{i=0}^{m_t}\alpha_i^t x_{t-i} + \beta_t \sum_{i=0}^{m_t}\alpha_i^t G(x_{t-i}) \\ &= \sum_{i=0}^{m_t}\alpha_i^t x_{t-i} + \beta_t \sum_{i=0}^{m_t}\alpha_i^t \Big(G(x_{t-i}) - x_{t-i}\Big) \\ &= \sum_{i=0}^{m_t}\alpha_i^t x_{t-i} + \beta_t \sum_{i=0}^{m_t}\alpha_i^t F_{t-i} \qquad (21) \\ &= x_t - \sum_{i=1}^{m_t}\alpha_i^t (x_t - x_{t-i}) \\ &\quad + \beta_t\Big(F_t - \sum_{i=1}^{m_t}\alpha_i^t (F_t - F_{t-i})\Big), \qquad (22) \end{aligned}$$

where (21) holds due to the definition $G_t = G(x_t) = x_t + F_t$, and (22) holds since $\sum_{i=0}^{m_t}\alpha_i^t = 1$.

Then, to bound $\|F_{t+1}\|_2$ (i.e., $\|\nabla f(x_{t+1})\|_2$), we deduce F_{t+1} as follows:

$$\begin{aligned} F_{t+1} &= G_{t+1} - x_{t+1} \\ &= G_{t+1} - \sum_{i=0}^{m_t}\alpha_i^t x_{t-i} - \beta_t \sum_{i=0}^{m_t}\alpha_i^t F_{t-i} \\ &= G_{t+1} - \sum_{i=0}^{m_t}\alpha_i^t (G_{t-i} - F_{t-i}) - \beta_t \sum_{i=0}^{m_t}\alpha_i^t F_{t-i} \\ &= G_{t+1} - \sum_{i=0}^{m_t}\alpha_i^t G_{t-i} + (1-\beta_t)\mathcal{F}, \qquad (23) \end{aligned}$$

where (23) uses the definition $\mathcal{F} \triangleq \sum_{i=0}^{m_t}\alpha_i^t F_{t-i}$. Now, we bound the first two terms of (23) as follows:

$$\begin{aligned} & G_{t+1} - \sum_{i=0}^{m_t}\alpha_i^t G_{t-i} \\ &= G_{t+1} - \big(G_t - \sum_{i=1}^{m_t}\alpha_i^t (G_t - G_{t-i})\big) \\ &= \int_0^1 G'\Big(x_t + u(x_{t+1} - x_t)\Big)(x_{t+1} - x_t)\, du \\ &\quad - \sum_{i=1}^{m_t}\alpha_i^t \int_0^1 G'\Big(x_t + u(x_{t-i} - x_t)\Big)(x_{t-i} - x_t)\, du \\ &= \sum_{i=1}^{m_t}\alpha_i^t \int_0^1 G'\Big(x_t + u(x_{t+1} - x_t)\Big)(x_{t-i} - x_t)\, du \\ &\quad + \int_0^1 G'\Big(x_t + u(x_{t+1} - x_t)\Big)\beta_t \mathcal{F}\, du \\ &\quad - \sum_{i=1}^{m_t}\alpha_i^t \int_0^1 G'\Big(x_t + u(x_{t-i} - x_t)\Big)(x_{t-i} - x_t)\, du, \end{aligned} \qquad (24)$$

where (24) is obtained by using (22) to replace x_{t+1}. To bound (24), we use Assumptions 1, 2, and the equation

$$G_t' = I + F_t' = I - \frac{2}{L+\mu}\nabla^2 f(x_t).$$

After some non-trivial calculations (details can be found in Appendix B.1), we obtain

$$\begin{aligned} \|F_{t+1}\| \le \frac{\gamma(m\|\alpha\|^2 + \sqrt{m}\|\alpha\|)\Delta_t^2}{L+\mu} + \frac{\gamma\sqrt{m}\|\alpha\|\beta_t\Delta_t\|\mathcal{F}\|}{L+\mu} \\ + \Big(1 - \frac{2\mu}{L+\mu}\beta_t\Big)\|\mathcal{F}\|, \end{aligned}$$

where $\|\alpha\|$ denotes the Euclidean norm of $\alpha = (\alpha_1^t, \ldots, \alpha_{m_t}^t)^T$. Then, according to the problem (1) and the definition of $\mathcal{F} \triangleq \sum_{i=0}^{m_t}\alpha_i^t F_{t-i}$, we have $\|\mathcal{F}\| \le \|F_t\|$. Finally, we bound $\|\alpha\| \le \frac{2\tilde{\kappa}}{L+\mu}$ using QR decomposition of problem (1) and recall $F_t = -\frac{2}{L+\mu}\nabla f(x_t)$ to finish the proof of Theorem 2. □

4 Guessing Algorithm

In this section, we provide a *Guessing Algorithm* (described in Algorithm 2) which guesses the parameters (e.g., μ, L) dynamically. Intuitively, we guess the parameter μ and the condition number κ in a doubling way. Note that in general these parameters are not available, since the time for computing these parameters is almost the same as (or even longer than) solving the original problem. Also note that the condition in Line 14 of Algorithm 2 depends on the algorithm used in Line 12.

Algorithm 2: Guessing Algorithm

input: x_0, T, δ, B
Let $t = 0$;
for $i = 1, 2, \ldots$ **do**
  $\kappa_i = e^{i+2}$;
  for $j = 1, 2, \ldots, \ln B$ **do**
    $\mu_i = e^j \delta, L_i = \mu_i \kappa_i, t_i = 1$;
    do
      $t_i = \lfloor e t_i \rfloor$;
      if $t + t_i > T$ **then**
        break;
      $x_{t-1} = x_t$;
      $x =$Anderson Acceleration(x_t, t_i, μ_i, L_i) //can be replaced by other algorithms;
      $t = t + t_i, x_t = x$;
    while $\frac{\|\nabla f(x_t)\|}{\|\nabla f(x_{t-1})\|} \le 2\left(\frac{\sqrt{\kappa_i}-1}{\sqrt{\kappa_i}+1}\right)^{t_i}$;
    if $\|\nabla f(x_t)\| > \|\nabla f(x_{t-1})\|$ **then**
      $x_t = x_{t-1}$;
return x_t

The convergence result of our Algorithm 2 is stated in the following Theorem 3. The detailed proof is deferred to Appendix B.3. Note that we only prove the quadratic case for Theorem 3, but it is possible to extend it to the general case.

Theorem 3 *Without knowing the parameters μ and L, Algorithm 2 achieves $O(\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)^2)$ convergence rate for obtaining an ϵ-approximate solution of problem (2), where $\kappa = L/\mu$, and B can be any number as long as the eigenvalue spectrum belongs to $[\delta, B\delta]$.*

Remark: We provide a simple example to show why this guessing algorithm is useful. Note that algorithms usually need the (exact) parameters μ and L to set the step size. Without knowing the exact values μ and L, one needs to approximate these parameters once at the beginning. Let $\mu' = \frac{1}{c_1}\mu$ and $L' = c_2 L$ denote the approximated values, where $c_1, c_2 \ge 1$. Without guessing them dynamically, one fixes μ' and L' all the time in its algorithm. According to the lower bound $\Omega(\sqrt{\kappa}\ln\frac{1}{\epsilon})$, we know that its convergence rate cannot be better than $O(\sqrt{\kappa'}\ln\frac{1}{\epsilon}) = O(\sqrt{c_1 c_2 \kappa}\ln\frac{1}{\epsilon})$, where $\kappa' = L'/\mu'$. However, if one combines with our Algorithm 2 (guessing the parameters dynamically), the convergence rate can be improved to $O(\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln(c_1c_2\kappa))^2)$ according to our Theorem 3 by letting $\delta = \mu'$ and $B\delta = L'$ (hence $B = c_1c_2\kappa$). Note that there is no ϵ (accuracy) in the second term $\sqrt{\kappa}(\ln\kappa\ln(c_1c_2\kappa))^2$. Thus the rate turns to the optimal $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$ when $\epsilon \to 0$. To achieve an ϵ-approximate solution, our guessing algorithm can improve the convergence a lot especially for an imprecise estimate at the beginning (i.e., c_1 and c_2 are very large). The corresponding experimental results in Section 5.1 (see Figure 6) indeed validate our theoretical results.

5 Experiments

In this section, we conduct the numerical experiments on the real-world UCI datasets[2] and synthetic datasets. We compare the performance among these five algorithms: Anderson Acceleration (AA), Anderson-Chebyshev acceleration (AA-Cheby), vanilla Gradient Descent (GD), Nesterov's Accelerated Gradient Descent (NAGD) (Nesterov, 2004) and Regularized Minimal Polynomial Extrapolation (RMPE) with $k = 5$ (same as (Scieur et al., 2016)).

Regarding the hyperparameters, we directly set them from their corresponding theoretical results. See Proposition 1 of (Lessard et al., 2016) for GD and NAGD. For RMPE5, we follow the same setting as in (Scieur et al., 2016). For our AA/AA-Cheby, we set them according to our Theorem 1 and 2.

Figure 2 demonstrates the convergence performance of these algorithms in general nonlinear case and Figures 3–5 demonstrate the convergence performance in quadratic case. The last Figure 6 demonstrates the convergence performance of these algorithms combined with our guessing algorithm (Algorithm 2). The values of m in the caption of figures denote the mixing parameter of Anderson acceleration algorithms (see Line 5 of Algorithm 1).

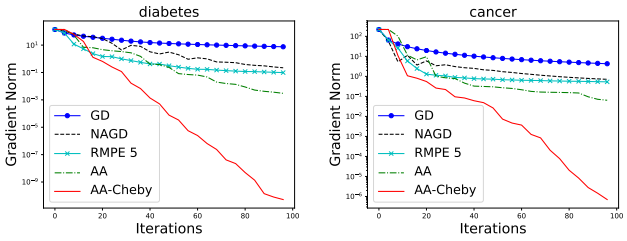


Figure 2: Logistic regression, $m = 3$

In Figure 2, we use the negative log-likelihood as the loss function f (logistic regression), i.e., $f(\theta) = -\sum_{i=1}^{n}(y_i \log\phi(\theta^T x_i) + (1-y_i)\log(1-\phi(\theta^T x_i)))$, where $\phi(z) = 1/(1+\exp(-z))$. We run these five algorithms on real-world *diabetes* and *cancer* datasets which are standard UCI datasets. The x-axis and y-axis represent the number of iterations and the norm of the gradient of loss function respectively.

[2]The UCI datasets can be downloaded from `https://archive.ics.uci.edu/ml`

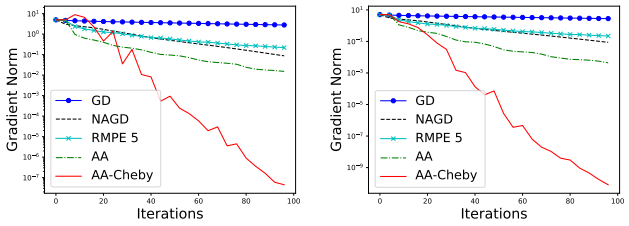


Figure 3: $\kappa \in [0, 500]$; $m = 3$ (left), $m = 5$ (right)

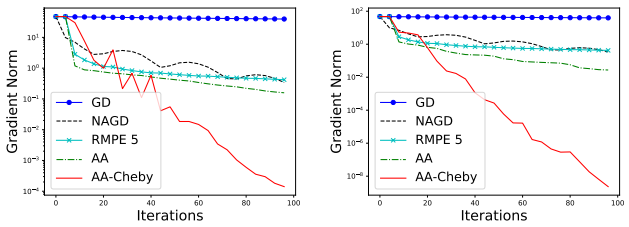


Figure 4: $\kappa \in [500, 2000]$; $m = 3$ (left), $m = 5$ (right)

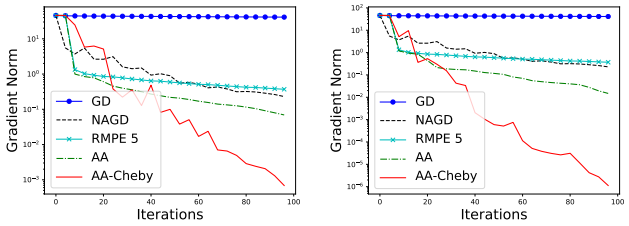


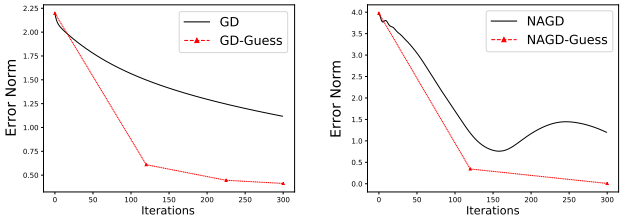
Figure 5: $\kappa \in [2000, 5000]$; $m = 3$ (left), $m = 5$ (right)

Figures 3–5 demonstrate the convergence performance for the quadratic case, where $f(x) = \frac{1}{2}x^T Ax - b^T x$. Concretely, we compared the convergence performance among these algorithms when the condition number $\kappa(A)$ and the mixing parameter m are varied, e.g., the left figure in Figure 3 is the case $\kappa \in [0, 500]$ and $m = 3$. Recall that m is the mixing parameter for Anderson acceleration algorithms (see Line 5 of Algorithm 1). We run these five algorithms on the synthetic datasets in which we randomly generate the A and b for the loss function f. Note that for randomly generated A satisfying the property of $A \in \mathcal{S}^d_{++}$, we randomly generate B instead and let $A \triangleq B^T B$.

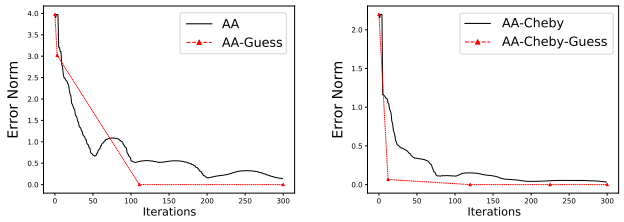
In conclusion, Anderson acceleration methods converge the fastest no matter it is a quadratic function or general function in all of our experiments. The efficient Anderson acceleration methods can be viewed as the extension of momentum methods (e.g., NAGD) since GD is the special case of Anderson Acceleration with $m = 0$, and to some extent NAGD can be viewed as $m = 1$. Combined with our theoretical results (i.e., optimal convergence rate in quadratic case and linear-quadratic convergence in general case), the experimental results validate that Anderson acceleration methods are efficient both in theory and practice.

5.1 Experiments for Guessing Algorithm

In this section, we conduct the experiments for guessing the hyperparameters (i.e., μ, L) dynamically using our Algorithm 2.



(a) Gradient Descent (b) Nesterov's AGD



(c) Anderson Acceleration (d) Anderson-Chebyshev

Figure 6: Algorithms with/without guessing algorithm

In Figure 6, we separately consider these algorithms. For each of them, we compare its convergence performance between its original version and the one combined with our guessing algorithm (Algorithm 2). The experimental results show that all these four algorithms combined with our guessing algorithm achieve much better performance than their original versions. Thus it validates our theoretical results (see Theorem 3 and its following Remark).

6 Conclusion

In this paper, we prove that Anderson acceleration with Chebyshev polynomial can achieve the optimal convergence rate $O(\sqrt{\kappa}\ln\frac{1}{\epsilon})$, which improves the previous result $O(\kappa\ln\frac{1}{\epsilon})$ provided by (Toth and Kelley, 2015). Thus it can deal with ill-conditioned problems (condition number κ is large) more efficiently. Furthermore, we also prove the linear-quadratic convergence of Anderson acceleration for minimizing general nonlinear problems. Besides, if the hyperparameters (e.g., the Lipschitz smooth parameter L) are not available, we propose a guessing algorithm for guessing them dynamically and also prove a similar convergence rate. Finally, the experimental results demonstrate that the efficient Anderson acceleration methods converge significantly faster than other algorithms. This validates that Anderson-Chebyshev acceleration is efficient both in theory and practice.

Acknowledgements

Zhize was supported by the Office of Sponsored Research of KAUST, through the Baseline Research Fund of Prof. Peter Richtárik. Jian was supported in part by the National Natural Science Foundation of China Grant 61822203, 61772297, 61632016, 61761146003, and the Zhongguancun Haihua Institute for Frontier Information Technology and Turing AI Institute of Nanjing. The authors also would like to thank Francis Bach, Claude Brezinski, Rong Ge, Damien Scieur, Le Zhang and anonymous reviewers for useful discussions and suggestions.

References

A Aitken. On bernoulli's numerical solution of algebraic equations. *Proceedings of the Royal Society of Edinburgh*, 46:289–305, 1926.

Zeyuan Allen-Zhu. Katyusha: the first direct acceleration of stochastic gradient methods. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1200–1205. ACM, 2017.

Donald G Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM*, 12(4):547–560, 1965.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Claude Brezinski. Convergence acceleration during the 20th century. *Journal of Computational and Applied Mathematics*, 122:1–21, 2000.

Claude Brezinski and M Redivo Zaglia. Extrapolation methods: theory and practice. 1991.

Claude Brezinski, Michela Redivo-Zaglia, and Yousef Saad. Shanks sequence transformations and anderson acceleration. *SIAM Review*, 60(3):646–669, 2018.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Steven Russell Capehart. *Techniques for accelerating iterative methods for the solution of mathematical problems*. PhD thesis, Oklahoma State University, 1989.

Claire Evans, Sara Pollock, Leo G Rebholz, and Mengying Xiao. A proof that anderson acceleration increases the convergence rate in linearly converging fixed point methods (but not in quadratically converging ones). *arXiv preprint arXiv:1810.08455*, 2018.

V Eyert. A comparative study on methods for convergence acceleration of iterative vector sequences. *Journal of Computational Physics*, 124(2):271–285, 1996.

Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pages 687–697, 2018.

Haw-ren Fang and Yousef Saad. Two classes of multisecant methods for nonlinear acceleration. *Numerical Linear Algebra with Applications*, 16(3):197–221, 2009.

Rong Ge, Zhize Li, Weiyao Wang, and Xiang Wang. Stabilized svrg: Simple variance reduction for nonconvex optimization. In *Conference on Learning Theory*, 2019.

GH Golub and CF Van Loan. Matrix computations. *3rd ed., The John Hopkins University Press, Baltimore, MD*, 1996.

Louis A Hageman and David M Young. *Applied iterative methods*. Courier Corporation, 2012.

Nicholas J Higham and Nataša Strabić. Anderson acceleration of the alternating projections method for computing the nearest correlation matrix. *Numerical Algorithms*, 72(4):1021–1042, 2016.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Guanghui Lan and Yi Zhou. Random gradient extrapolation for distributed and stochastic optimization. *SIAM Journal on Optimization*, 28(4):2753–2782, 2018.

Guanghui Lan, Zhize Li, and Yi Zhou. A unified variance-reduced accelerated gradient method for convex optimization. In *Advances in Neural Information Processing Systems*, 2019.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex finite-sum optimization via scsg methods. In *Advances in Neural Information Processing Systems*, pages 2348–2358, 2017.

Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.

Zhize Li. Ssrgd: Simple stochastic recursive gradient descent for escaping saddle points. In *Advances in Neural Information Processing Systems*, 2019.

Zhize Li and Jian Li. A simple proximal stochastic gradient method for nonsmooth nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 5569–5579, 2018.

Zhize Li, Tianyi Zhang, Shuyu Cheng, Jun Zhu, and Jian Li. Stochastic gradient hamiltonian monte carlo with variance reduction for bayesian inference. *Machine Learning*, 108(8-9):1701–1727, 2019.

John Loffeld and Carol S Woodward. Considerations on the implementation and use of anderson acceleration on distributed memory and gpu-based parallel computers. *Advances in the Mathematical Sciences*, page 417, 2016.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer, 2004.

Maxim A Olshanskii and Eugene E Tyrtyshnikov. *Iterative methods for linear systems: theory and applications*. SIAM, 2014.

Florian A Potra and Hans Engler. A characterization of the behavior of the anderson acceleration on linear problems. *Linear Algebra and its Applications*, 438(3):1002–1011, 2013.

Phanisri P Pratapa, Phanish Suryanarayana, and John E Pask. Anderson acceleration of the jacobi iterative method: An efficient alternative to krylov methods for large, sparse linear systems. *Journal of Computational Physics*, 306:43–54, 2016.

Theodore J Rivlin. *The Chebyshev polynomials*. Wiley, 1974.

Youcef Saad and Martin H Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

Damien Scieur, Alexandre d'Aspremont, and Francis Bach. Regularized nonlinear acceleration. In *Advances in Neural Information Processing Systems*, pages 712–720, 2016.

Damien Scieur, Edouard Oyallon, Alexandre d'Aspremont, and Francis Bach. Nonlinear acceleration of deep neural networks. *arXiv preprint arXiv:1805.09639v1*, 2018.

Damien Scieur, Edouard Oyallon, Alexandre d'Aspremont, and Francis Bach. Online regularized nonlinear acceleration. *arXiv preprint arXiv:1805.09639v2*, 2019.

Daniel Shanks. Non-linear transformations of divergent and slowly convergent sequences. *Studies in Applied Mathematics*, 34(1-4):1–42, 1955.

Avram Sidi, William F Ford, and David A Smith. Acceleration of convergence of vector sequences. *SIAM Journal on Numerical Analysis*, 23(1):178–196, 1986.

David A Smith, William F Ford, and Avram Sidi. Extrapolation methods for vector sequences. *SIAM review*, 29(2):199–233, 1987.

Alex Toth and CT Kelley. Convergence analysis for anderson acceleration. *SIAM Journal on Numerical Analysis*, 53(2):805–819, 2015.

Homer F Walker and Peng Ni. Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis*, 49(4):1715–1735, 2011.

Junzi Zhang, Brendan O'Donoghue, and Stephen Boyd. Globally convergent type-i anderson acceleration for non-smooth fixed-point iterations. *arXiv preprint arXiv:1808.03971*, 2018.

Dongruo Zhou, Pan Xu, and Quanquan Gu. Stochastic nested variance reduction for nonconvex optimization. *arXiv preprint arXiv:1806.07811*, 2018.

A GMRES vs. Anderson Acceleration($m = \infty$)

In this appendix, in order to better understand the efficient Anderson acceleration method, we review the equivalence between the well-known Krylov subspace method GMRES (Saad and Schultz, 1986) and Anderson acceleration without truncation (i.e., $m = \infty$ or large enough in Line 5 of Algorithm 1) in linear case. We emphasize that in this paper we focus on the more general hard cases where m is small (since m usually is finite and not very large in practice) and also general nonlinear case.

Consider the problem of solving the linear system $Ax = b$, with a nonsingular matrix A. This is equivalent to solving the fixed point $x = G(x) = x - \nabla f(x)$, where $\nabla f(x) = Ax - b$. Let r_i denote the residual in the point x_i, i.e., $r_i = b - Ax_i$. The GMRES method is an effective iterative method for linear system which has the property of minimizing the norm of the residual vector over a Krylov subspace at every step.

$$x_t^{\mathsf{GMRES}} = \arg\min\{\|b - Ax\|_2 : x = x_0 + y, y \in \mathcal{K}_t\} \tag{25}$$

Note that the Krylov space $\mathcal{K}_t$ is the linear span of the first t gradients and $\mathcal{K}_n$ can span the whole space $\mathbb{R}^n$. Hence the method arrives the exact solution after n iteration. It is also theoretically equivalent to the Generalized Conjugate Residual method (GCR).

Now we show that $x_{t+1}^{\mathsf{AA}} = G(x_t^{\mathsf{GMRES}})$ to indicate the equivalence, under the assumption $0 < \|r_i\|_2 < \|r_{i-1}\|_2$ for $1 \le i \le t$. x_t^{GMRES} and x_{t+1}^{AA} denote the t-th GMRES iterative point and $t+1$-th Anderson Acceleration iterative point, respectively. Let mixing parameters $\beta_t = 1$ for all t. Then, we deduce the x_{t+1}^{AA} as follows:

$$\begin{aligned} x_{t+1}^{\mathsf{AA}} &= \sum_{i=0}^{t} \alpha_i^t G(x_i) \qquad \because m_t = t && (26)\\ &= \sum_{i=0}^{t} \alpha_i^t x_i + \sum_{i=0}^{t} \alpha_i^t (G(x_i) - x_i) && (27)\\ &= \sum_{i=0}^{t} \alpha_i^t x_i + \sum_{i=0}^{t} \alpha_i^t F_i && (28) \end{aligned}$$

Note that the second term in (28) is the same as we minimized in Line 7 of Algorithm 1. This step also can be transformed to an unconstrained version as follows:

$$\min_{(\alpha_1^t,\ldots,\alpha_t^t)^T} \|F_0 - \sum_{i=1}^{t} \alpha_i^t (F_0 - F_i)\|_2 \tag{29}$$

The α_0^t equals to $1 - \sum_{i=1}^{t} \alpha_i^t$. Note that $F_0 - F_i = b - Ax_0 - (b - Ax_i) = A(x_i - x_0)$ and $F_0 = r_0 = b - Ax_0$. Replacing these equations into (29), we have

$$\begin{aligned} &\min_{(\alpha_1^t,\ldots,\alpha_t^t)^T} \|F_0 - \sum_{i=1}^{t} \alpha_i^t (F_0 - F_i)\|_2 && (30)\\ =&\min_{(\alpha_1^t,\ldots,\alpha_t^t)^T} \|b - Ax_0 - \sum_{i=1}^{t} \alpha_i^t A(x_i - x_0)\|_2 && (31)\\ =&\min_{(\alpha_1^t,\ldots,\alpha_t^t)^T} \|b - A\Big(x_0 + \sum_{i=1}^{t} \alpha_i^t (x_i - x_0)\Big)\|_2 && (32) \end{aligned}$$

Comparing (32) with (25), if $\{y_i = (x_i - x_0) : 1 \le i \le t\}$ form a basis of Krylov subspace $\mathcal{K}_t$, then we have the following equations (easily from (30)-(25)). Note that the Krylov subspaces $\mathcal{K}$ are defined by (r_0, A), i.e.,

$\mathcal{K}_i = \text{span}\{r_0, Ar_0, \ldots, A^{i-1}r_0\}$.

$$x_t^{\mathsf{GMRES}} = x_0 + \sum_{i=1}^{t} \alpha_i^t (x_i - x_0) \tag{33}$$

$$r_t^{\mathsf{GMRES}} = b - Ax_t^{\mathsf{GMRES}} = F_0 - \sum_{i=1}^{t} \alpha_i^t (F_0 - F_i) \tag{34}$$

Now we continue to deduce the x_{t+1}^{AA} from (28) to finish the proof of equivalence.

$$\begin{aligned} x_{t+1}^{\mathsf{AA}} &= \sum_{i=0}^{t} \alpha_i^t x_i + \sum_{i=0}^{t} \alpha_i^t F_i && (35)\\ &= x_0 + \sum_{i=1}^{t} \alpha_i^t (x_i - x_0) + F_0 - \sum_{i=1}^{t} \alpha_i^t (F_0 - F_i) && (36)\\ &= x_t^{\mathsf{GMRES}} + b - Ax_t^{\mathsf{GMRES}} && (37)\\ &= x_t^{\mathsf{GMRES}} - \nabla f(x_t^{\mathsf{GMRES}}) && (38)\\ &= G(x_t^{\mathsf{GMRES}}) && (39) \end{aligned}$$

Now the only remaining thing is to show that $\{y_i = (x_i - x_0) : 1 \le i \le t\}$ form the basis of Krylov subspace $\mathcal{K}_t$. This can be proved by induction. For $t = 1$, $y_1 = x_1 - x_0 = G(x_0) - x_0 = x_0 - (Ax_0 - b) - x_0 = r_0$. Now, assuming that $\{y_i = (x_i - x_0) : 1 \le i \le t\}$ form the basis of $\mathcal{K}_t$, we show that

$$\begin{aligned} y_{t+1} &= x_{t+1} - x_0 \\ &= \sum_{i=1}^{t} \alpha_i^t (x_i - x_0) + F_0 - \sum_{i=1}^{t} \alpha_i^t (F_0 - F_i) && (40)\\ &= \sum_{i=1}^{t} \alpha_i^t y_i + r_t^{\mathsf{GMRES}}, && (41) \end{aligned}$$

where (40) follows from (36), and (41) follows from (34). The first term in (41) belongs to $\mathcal{K}_t$ by induction. The second term $r_t^{\mathsf{GMRES}} \in \mathcal{K}_{t+1}$ (From (25)) and $r_t^{\mathsf{GMRES}} \notin \mathcal{K}_t$ since the assumption $0 < \|r_i\|_2 < \|r_{i-1}\|_2$ for $1 \le i \le t$. Hence $y_{t+1} \in \mathcal{K}_{t+1}$.

B Missing Proofs

In this appendix, we provide the proof details for Theorem 2 (Appendix B.1), Lemma 1 (Appendix B.2) and Theorem 3 (Appendix B.3).

B.1 Proof of Theorem 2

For the iteration $t+1$, we have $F_t = F(x_t) = -\frac{2}{L+\mu}\nabla f(x_t)$ according to $\lambda = \frac{2}{L+\mu}$, where μ and L are defined in Assumption 1. First, we recall the form of F_{t+1} (i.e. (23)) as

$$F_{t+1} = G_{t+1} - \sum_{i=0}^{m_t} \alpha_i^t G_{t-i} + (1 - \beta_t)\mathcal{F}, \tag{42}$$

and the definition of $\mathcal{F}$ as

$$\mathcal{F} \triangleq \sum_{i=0}^{m_t} \alpha_i^t F_{t-i}. \tag{43}$$

Now we bound the first two terms in RHS of (42) by combining the first and third term of (24) as follows:

$$\begin{aligned}
&G_{t+1} - \sum_{i=0}^{m_t} \alpha_i^t G_{t-i} \\
&= \sum_{i=1}^{m_t} \alpha_i^t \int_0^1 \Big(G'\Big(x_t + u(x_{t+1} - x_t)\Big) - G'\Big(x_t + u(x_{t-i} - x_t)\Big)\Big)(x_{t-i} - x_t)\, du \\
&\quad + \int_0^1 G'\Big(x_t + u(x_{t+1} - x_t)\Big)\beta_t \mathcal{F}\, du. \qquad (44)
\end{aligned}$$

To bound the Equation (44), we recall that $G_t = G(x_t) = x_t + F_t$ and $F_t = -\frac{2}{L+\mu}\nabla f(x_t)$. Hence

$$G_t' = I + F_t' = I - \frac{2}{L+\mu}\nabla^2 f(x_t). \qquad (45)$$

Due to the Lipschitz continuity of Hessian $\nabla^2 f$ (see (19)), we have

$$\begin{aligned}
\|G'(x) - G'(y)\| &= \frac{2}{L+\mu}\|\nabla^2 f(x) - \nabla^2 f(y)\| \\
&\le \frac{2\gamma}{L+\mu}\|x - y\|. \qquad (46)
\end{aligned}$$

Now the first term in (44) can be bounded as follows:

$$\begin{aligned}
&\sum_{i=1}^{m_t} \alpha_i^t \int_0^1 \Big(G'\Big(x_t + u(x_{t+1} - x_t)\Big) - G'\Big(x_t + u(x_{t-i} - x_t)\Big)\Big)(x_{t-i} - x_t)\, du \\
&\le \sum_{i=1}^{m_t} \alpha_i^t \frac{\gamma\|x_{t+1} - x_t - (x_{t-i} - x_t)\|\|x_{t-i} - x_t\|}{L+\mu}. \qquad (47)
\end{aligned}$$

Using (22) to replace x_{t+1} and combining with (43), we have

$$\begin{aligned}
&\|x_{t+1} - x_t - (x_{t-i} - x_t)\| \\
&= \|\sum_{i=1}^{m_t} \alpha_i^t (x_{t-i} - x_t) + \beta_t \mathcal{F} - (x_{t-i} - x_t)\| \\
&\le (\sqrt{m}\|\alpha\| + 1) \max_{i\in[1,m]} \|x_t - x_{t-i}\| + \beta_t\|\mathcal{F}\| \qquad (48) \\
&= (\sqrt{m}\|\alpha\| + 1)\Delta_t + \beta_t\|\mathcal{F}\|, \qquad (49)
\end{aligned}$$

where (48) uses triangle inequality and Cauchy–Schwarz inequality. Now, pluging (49) into (47), we get

$$(47) \le \frac{\sqrt{m}\|\alpha\|\gamma((\sqrt{m}\|\alpha\| + 1)\Delta_t + \beta_t\|\mathcal{F}\|)\Delta_t}{L+\mu}, \qquad (50)$$

where (50) uses Cauchy–Schwarz inequality. Then, we bound the second term in (44) as follows:

$$\begin{aligned}
&\int_0^1 G'\Big(x_t + u(x_{t+1} - x_t)\Big)\beta_t \mathcal{F}\, du \\
&= \int_0^1 \Big(I - \frac{2}{L+\mu}\nabla^2 f\Big(x_t + u(x_{t+1} - x_t)\Big)\Big)\beta_t \mathcal{F}\, du \\
&\le \Big(1 - \frac{2\mu}{L+\mu}\Big)\beta_t\|\mathcal{F}\|. \qquad (51)
\end{aligned}$$

Now, we recall F_{t+1} here:

$$F_{t+1} = G_{t+1} - \sum_{i=0}^{m_t} \alpha_i^t G_{t-i} + (1-\beta_t)\mathcal{F} \quad \text{same as (42)}$$

Then, according to (44), (47), (50) and (51), we have

$$\begin{aligned}\|F_{t+1}\| &\le \frac{\sqrt{m}\|\alpha\|\gamma((\sqrt{m}\|\alpha\|+1)\Delta_t+\beta_t\|\mathcal{F}\|)\Delta_t}{L+\mu}+\Big(1-\frac{2\mu}{L+\mu}\Big)\beta_t\|\mathcal{F}\|+(1-\beta_t)\|\mathcal{F}\|\\ &= \frac{\gamma(m\|\alpha\|^2+\sqrt{m}\|\alpha\|)\Delta_t^2}{L+\mu}+\frac{\gamma\sqrt{m}\|\alpha\|\beta_t\Delta_t\|\mathcal{F}\|}{L+\mu}+\Big(1-\frac{2\mu}{L+\mu}\beta_t\Big)\|\mathcal{F}\|.\end{aligned}$$

Recall that $F_t=-\frac{2}{L+\mu}\nabla f(x_t)$. According to (43) and (1), we have $\|\mathcal{F}\|\le\|F_t\|$. Thus, we have

$$\begin{aligned}\|\nabla f(x_{t+1})\| \le{}& \frac{\gamma(m\|\alpha\|^2+\sqrt{m}\|\alpha\|)\Delta_t^2}{2}+\frac{\gamma\sqrt{m}\|\alpha\|\beta_t\Delta_t\|\nabla f(x_t)\|}{L+\mu}\\ &+\Big(1-\frac{2\mu}{L+\mu}\beta_t\Big)\|\nabla f(x_t)\|.\end{aligned}\tag{52}$$

Now, we bound $\|\alpha\|\le\frac{2\tilde{\kappa}}{L+\mu}$ to finish the proof for Theorem 2. First we recall that the α satisfies problem (1), i.e., $\alpha=\arg\min_\alpha\|F_t-\mathcal{B}\alpha\|_2$. We use the QR decomposition for $\mathcal{B}$ and let $\mathcal{B}=QR$, where $Q^TQ=I$ and R is an upper triangular matrix. Then we let $\tilde{R}$ denote the upper nonzeros of R, and $\tilde{Q}$ is the matrix with the corresponding columns of Q. Then $\tilde{R}\alpha=\tilde{Q}^TF_t$ and $\alpha=\tilde{R}^{-1}\tilde{Q}^TF_t$. Hence, we have

$$\|\alpha\|=\|\tilde{R}^{-1}\tilde{Q}^TF_t\|\le\|\tilde{R}^{-1}\|\|\tilde{Q}^TF_t\|\le\|\tilde{R}^{-1}\|\|Q^TF_t\|\le 2\kappa/(L+\mu),\tag{53}$$

where (53) uses $F_t=-\frac{2}{L+\mu}\nabla f(x_t)$ and $\tilde{\kappa}=\|\nabla f(x_t)\|/\tilde{\mu}$ (where $\tilde{\mu}$ denotes the least non-zero singular value of $\tilde{R}$). The proof for Theorem 2 is finished by plugging (53) into (52). □

B.2 Proof of Lemma 1

First, we obtain the relation between F_{t+1} and F_1 by using the Singular Value Decomposition (SVD) for the small matrix B_t for all t.

Concretely, Let $\alpha_0^t=1-\sum_{i=1}^{m_t}\alpha_i^t$. Recall that B_t denotes $[F_t-F_{t-1},\ldots,F_t-F_{t-m_t}]$, i.e. a matrix with column vectors are F_t-F_{t-i} for $1\le i\le m_t$. Then we adopt the SVD of B_t as $\tilde{U}_t\Sigma_t\tilde{V}_t^T$, where $\tilde{U}_t^T\tilde{U}_t=I$, $\tilde{V}_t^T\tilde{V}_t=I$ and $\Sigma_t=\mathbf{diag}(\sigma_1,\ldots,\sigma_r)$, $r=\mathbf{rank}(B_t)$. Then $B_t^\dagger=\tilde{V}_t\Sigma_t^{-1}\tilde{U}_t^T$. Although B_t may have dependent columns, one solution for (1) is that $\alpha^*=B_t^\dagger F_t$, where $\alpha^*=(\alpha_1^t,\ldots,\alpha_{m_t}^t)^T$ is the vector of coefficients in (1). Therefore, $\sum_{i=0}^{m_t}\alpha_i^tF_{t-i}$ can be represented as $F_t-B_tB_t^\dagger F_t=F_t-\tilde{U}_t\tilde{U}_t^TF_t$.

Let $P_t=I-\tilde{U}_t\tilde{U}_t^T$. The matrix P_t is a projection matrix since $P_tP_t=(I-\tilde{U}_t\tilde{U}_t^T)(I-\tilde{U}_t\tilde{U}_t^T)=I-\tilde{U}_t\tilde{U}_t^T$. Also P_t is symmetric. So we finally have $F_{t+1}=(I-\beta_tA)P_tF_t$. Expanding F_t recursively, we get the following relation

$$F_{t+1}=(I-\beta_tA)P_t\cdots(I-\beta_1A)P_1F_1.\tag{54}$$

We can further have $\|P_j\|_2\le 1$, for $1\le j\le t$. This is due to the following fact

$$\|P_jx\|_2^2=(P_jx)^T(P_jx)=x^TP_j^TP_jx=x^TP_jx\le\|x\|_2\|P_jx\|_2.$$

As $A\in\mathcal{S}_{++}^d$, $A=Q\Lambda Q^T$, where $Q^TQ=I$, and $\Lambda=\mathbf{diag}(\lambda_1,\ldots,\lambda_n)$ (λ_j's are the real eigenvalues of A).

Now, we need to bound $\|F_{t+1}\|_2$. According to (54), we have

$$\begin{aligned}\|F_{t+1}\|_2 &= \|(I-\beta_tA)P_t\cdots(I-\beta_1A)P_1F_1\|_2\\ &\le \|(I-\beta_tA)P_t\cdots(I-\beta_1A)P_1\|_2\|F_1\|_2.\end{aligned}\tag{55}$$

It is sufficient to bound $\|(I-\beta_tA)P_t\cdots(I-\beta_1A)P_1\|_2$, which is

$$\sup_{\|x\|_2=1}\|(I-\beta_tA)P_t\cdots(I-\beta_1A)P_1x\|_2.\tag{56}$$

Denote the column vectors of Q as $v_1,\ldots,v_d$ (they are the eigenvectors of A). The vector x can be represented as $\sum_{j=1}^d c_{0,j}v_j$, for some $c_{0,j}$'s with $\sum_{j=1}^d c_{0,j}^2=1$. Hence P_1x can be represented as $P_1x=\sum_{j=1}^n c_{1,j}v_j$. As

$\|P_1\|_2 \le 1$, the $c_{1,j}$'s satisfy $\sum_{j=1}^d c_{1,j}^2 \le 1$. With $P_1 x$, we know $(I-\beta_1 A)P_1 x = \sum_{j=1}^d c_{1,j}(1-\beta_1\lambda_j)v_j$, where $\sum_j c_{1,j}^2 \le 1$. Iteratively expanding, we get $(I-\beta_t A)P_t \cdots (I-\beta_1 A)P_1 x = \sum_{j=1}^d c_{t,j}(1-\beta_t\lambda_j)\cdots(1-\beta_1\lambda_j)v_j$, where $\sum_j c_{t,j}^2 \le (1+\frac{1}{\sqrt{\kappa}+1})^t$. Hence we have

$$(56) \le \Big(1+\frac{1}{\sqrt{\kappa}+1}\Big)^t \min_{\beta} \max_{\lambda\in \mathrm{sp}(A)} |H_t(\lambda)|, \tag{57}$$

where $H_t(\lambda) = (1-\beta_t\lambda)\cdots(1-\beta_1\lambda)$ is a degree t polynomial and the $\mathrm{sp}(A)$ is the eigenvalue spectrum of A. As in general, the eigenvalues of A is unknown. We look for the bound of the following form (58) instead of (57),

$$(57) \le \Big(1+\frac{1}{\sqrt{\kappa}+1}\Big)^t \min_{\beta} \max_{\lambda\in[\mu,L]} |H_t(\lambda)|. \tag{58}$$

Finally, combining (55), (56), (57), (58), (14), (15) and the fact

$$\Big(1+\frac{1}{\sqrt{\kappa}+1}\Big)^t \Big(1-\frac{2}{\sqrt{\kappa}+1}\Big)^{t/2} \le 1,$$

we finish the proof, i.e.,

$$\|F_{t+1}\|_2/\|F_1\|_2 \le \sqrt{2\min_{\beta}\max_{\lambda\in[\mu,L]}|H_t(\lambda)|}.$$

□

B.3 Proof of Theorem 3

Before to prove Theorem 3, we need the following three lemmas.

Lemma 2 *If $\sum_{j=1}^k e^{i_j} = e^{i_1} + e^{i_2} + \ldots + e^{i_k} = T$, then $\sum_{j=1}^k i_j \le k\ln\frac{T}{k}$.*

Proof. Let $g(x) = e^x$. Note that $g(x)$ is a convex function. According to Jensen's inequality, the following holds.

$$g(\mathbb{E}[x]) = \exp\Big(\frac{1}{k}\sum_{j=1}^k i_j\Big) \le \mathbb{E}[g(x)] = \frac{1}{k}\sum_{j=1}^k \exp(i_j)$$

We obtain $\sum_{j=1}^k i_j \le k\ln\frac{T}{k}$ by taking log for both sides. □

Lemma 3 *Let $T = c\big(\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)^2\big)$, where $c > 2$, then $\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{T}{\ln\kappa\ln B} \le T$ is satisfied.*

Proof. We divide this proof into three cases.

1. $\ln\frac{1}{\epsilon} \le \ln\kappa\ln B$.
   The left-hand side (LHS) of the constraint inequality in this lemma is deduced as follows:

$$\begin{aligned}
&\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{T}{\ln\kappa\ln B} \\
\le\ &\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{2c\sqrt{\kappa}(\ln\kappa\ln B)^2}{\ln\kappa\ln B} \\
=\ &\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)\big(\ln\sqrt{\kappa} + \ln(\ln\kappa\ln B) + \ln 2c\big)
\end{aligned}$$

   Hence $c \ge 2$ is enough for satisfying LHS $\le T$.

2. $\ln\frac{1}{\epsilon} > \ln\kappa\ln B > \ln\ln\frac{1}{\epsilon}$.

We also deduce the LHS of the constraint inequality as follows:

$$\begin{aligned}
&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{T}{\ln\kappa\ln B}\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{c(\sqrt{\kappa}\ln\frac{1}{\epsilon})\,(1+\ln\kappa\ln B)}{\ln\kappa\ln B}\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln\left(2c\sqrt{\kappa}\ln\frac{1}{\epsilon}\right)\\
=&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\Big(\frac{1}{2}\ln\kappa+\ln\ln\frac{1}{\epsilon}+\ln 2c\Big)\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\Big(\frac{1}{2}\ln\kappa+\ln\kappa\ln B+\ln 2c\Big)
\end{aligned}$$

Hence $c\ge 2$ is also enough for satisfying LHS $\le T=c\left(\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)^2\right)$.

3. $\ln\kappa\ln B\le\ln\ln\frac{1}{\epsilon}$.
We deduce the LHS of the constraint inequality as follows:

$$\begin{aligned}
&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{T}{\ln\kappa\ln B}\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{c\sqrt{\kappa}\left(\ln\frac{1}{\epsilon}+(\ln\ln\frac{1}{\epsilon})^2\right)}{\ln\kappa\ln B}\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\ln c\sqrt{\kappa}\Big(\ln\frac{1}{\epsilon}+\big(\ln\ln\frac{1}{\epsilon}\big)^2\Big)\\
\le&\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)\left(\frac{1}{2}\ln\kappa\right)+\sqrt{\kappa}\ln\ln\frac{1}{\epsilon}\left(\ln\ln\frac{1}{\epsilon}+\ln c+2\ln\left(\ln\ln\frac{1}{\epsilon}\right)\right)
\end{aligned}$$

Since $\ln(1/\epsilon)>(\ln\ln\frac{1}{\epsilon})^2$ if $(1/\epsilon)>e^2$. Hence it shows that $c\ge 2$ is enough for satisfying LHS $\le T=c\left(\sqrt{\kappa}\ln\frac{1}{\epsilon}+\sqrt{\kappa}(\ln\kappa\ln B)^2\right)$.

□

Lemma 4 *The condition number κ_i (in Line 4 of Algorithm 2) is always less than $e^2\kappa$, where κ is the true condition number. Equivalently, i (in Line 3 of Algorithm 2) is always less than $\ln\kappa$.*

Proof. Without loss of generality, let $e^c\le\mu\le e^{c+1}$ and $e^d\le L\le e^{d+1}$. When $\kappa_i=e^2\kappa$ and $j=e^c$, then $[\mu,L]\subset[\mu_i,L_i]$. According to inequality $\|\nabla f(x_{t+1})\|_2\le 2\big(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\big)^t\|\nabla f(x_1)\|_2$ (see the end of the proof of Theorem 1), the condition of do-while loop in Line 7–14 of Algorithm 2 always hold. The only way to break the loop is that the iteration $t>T$, i.e., the end of the algorithm. □

Proof of Theorem 3. According to Lemma 4, i (in Line 3 of Algorithm 2) is less than $\ln\kappa$ and κ_i is less than $e^2\kappa$. The inner loop j (in Line 5) is obviously less than $\ln B$. Let $k=\ln\kappa\ln B$ and i_j denote the times of the execution of do-while loop (Line 7–14) in each loop iteration (Line 5–16). Thus, the total number of iterations (corresponding to t) is e^{i_j} in each loop iteration. These i_j iterations satisfy the do-while condition, i.e., $\frac{\|\nabla f(x_t)\|}{\|\nabla f(x_{t-1})\|}\le 2\left(\frac{\sqrt{\kappa_i}-1}{\sqrt{\kappa_i}+1}\right)^{t_i}$.

We combine the condition together to obtain $\|\nabla f(x_t)\|\le 2^{i_j}\left(\frac{\sqrt{\kappa_i}-1}{\sqrt{\kappa_i}+1}\right)^{e^{i_j}}\|\nabla f(x_{t-e^{i_j}})\|$. Finally, this guessing algorithm satisfied the following Inequality (59).

Note that the Line 15 and 16 of Algorithm 2 ignore the failed iterations. Also this ignored step can be executed at most once in each loop iteration (Line 5–16). Let T denote the total number of iterations of Algorithm 2. Then $\sum_{j=1}^k e^{i_j}\le T\le 2\sum_{j=1}^k e^{i_j}+e\ln\kappa\ln B$.

$$\|\nabla f(x_t)\|\le 2^{\sum_{j=1}^k i_j}\left(\frac{\sqrt{e^2\kappa}-1}{\sqrt{e^2\kappa}+1}\right)^{\sum_{j=1}^k e^{i_j}}\|\nabla f(x_0)\| \tag{59}$$

As κ_i is less than $e^2\kappa$ and $k=\ln\kappa\ln B$. In order to prove the convergence rate, we need the RHS of (59) $\le\epsilon$, it

is sufficient to satisfy the following inequality

$$\sum_{j=1}^{k} i_j \le \frac{2}{\sqrt{e^2\kappa}+1}\left(\sum_{j=1}^{k} e^{i_j} - \frac{e\sqrt{\kappa}+1}{2}\ln\frac{1}{\epsilon}\right),$$

i.e.,

$$\frac{e\sqrt{\kappa}+1}{2}\ln\frac{1}{\epsilon} + \frac{e\sqrt{\kappa}+1}{2}\sum_{j=1}^{k} i_j \le \sum_{j=1}^{k} e^{i_j}. \tag{60}$$

By applying Lemma 2 and ignoring the constant, we can transform (60) to (61). Recall that $\sum_{j=1}^{k} e^{i_j} \le T \le 2\sum_{j=1}^{k} e^{i_j} + e\ln\kappa\ln B$ and $k = \ln\kappa\ln B$.

$$\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)\ln\frac{T}{\ln\kappa\ln B} \le T. \tag{61}$$

This is exactly the same as Lemma 3. Thus the proof is finished by using Lemma 3, i.e., T is bounded by $O\left(\sqrt{\kappa}\ln\frac{1}{\epsilon} + \sqrt{\kappa}(\ln\kappa\ln B)^2\right)$. □