# A Damped Newton Method Achieves Global $\mathcal{O}\left(\frac{1}{k^2}\right)$ and Local Quadratic Convergence Rate

**Slavomír Hanzely**
MBZUAI
KAUST

**Dmitry Kamzolov**
MBZUAI

**Dmitry Pasechnyuk**
MBZUAI
MIPT
ISP RAS

**Alexander Gasnikov**
MIPT
ISP RAS
HSE University

**Peter Richtárik**
KAUST

**Martin Takáč**
MBZUAI

## Abstract

In this paper, we present the first stepsize schedule for Newton method resulting in fast global and local convergence guarantees. In particular, a) we prove an $\mathcal{O}\left(\frac{1}{k^2}\right)$ global rate, which matches the state-of-the-art global rate of cubically regularized Newton method of Polyak and Nesterov (2006) and of regularized Newton method of Mishchenko (2021) and Doikov and Nesterov (2021), b) we prove a local quadratic rate, which matches the best-known local rate of second-order methods, and c) our stepsize formula is simple, explicit, and does not require solving any subproblem. Our convergence proofs hold under affine-invariance assumptions closely related to the notion of self-concordance. Finally, our method has competitive performance when compared to existing baselines, which share the same fast global convergence guarantees.

## 1 Introduction

Second-order optimization methods are the backbone of much of industrial and scientific computing. With origins that can be tracked back several centuries to the pioneering works of Newton [Newton, 1687], Raphson [Raphson, 1697] and Simpson [Simpson, 1740], they were extensively studied, generalized, modified, and improved in the last century [Kantorovich, 1948, Moré, 1978, Griewank, 1981a]. For a review of the historical development of the classical Newton-Raphson method, we refer the reader to the work of Ypma [1995]. The number of extensions and applications of second-order optimization methods is enormous; for example, the survey of Conn et al. [2000] on trust-region and quasi-Newton methods cited over a thousand papers.

### 1.1 Second-order methods and modern machine learning

Despite the rich history of the field, research on second-order methods has been flourishing up to this day. Some of the most recent development in the area was motivated by the needs of modern machine learning. Data-oriented machine learning depends on large datasets (both in number of features and number of datapoints), which are often stored in distributed/decentalized fashion. Consequently, there is a need for scalable algorithms.

To tackle large number of features, Qu et al. [2016], Gower et al. [2019], Doikov and Richtárik [2018] and Hanzely et al. [2020] proposed variants of Newton method operating in random low-dimensional subspaces. On the other hand, Pilanci and Wainwright [2017], Xu et al. [2020] and Kovalev et al. [2019] developed subsampled Newton methods for solving empirical risk minimization (ERM)

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

problems with large training datasets. Additionally, Bordes et al. [2009], Mokhtari and Ribeiro [2015], Gower et al. [2016], Byrd et al. [2016] and Kovalev et al. [2020] proposed stochastic variants of quasi-Newton methods. To tackle non-centralized nature of datasets, Shamir et al. [2014], Reddi et al. [2016], Wang et al. [2018] and Crane and Roosta [2019] considered distributed variants of Newton method, with improvements under various data/function similarity assumptions. Islamov et al. [2021], Safaryan et al. [2022], Qian et al. [2022], Islamov et al. [2022] and Agafonov et al. [2022] developed communication-efficient distributed variants of Newton method using the idea of communication compression and error compensation, without the need for any similarity assumptions.

We highlight two main research directions throughout of history of second-order methods: globally convergent methods under additional second-order smoothness (4) [Nesterov and Polyak, 2006] and local methods for self-concordant problems (12) [Nesterov and Nemirovski, 1989]. Former approach lead to various improvements such as acceleration Nesterov [2008], Monteiro and Svaiter [2013], usage of inexact information Ghadimi et al. [2017], Agafonov et al. [2020], generalization to tensor methods and their acceleration Nesterov [2021a], Gasnikov et al. [2019], Kovalev and Gasnikov [2022], superfast second-order methods under higher smoothness Nesterov [2021c,b], Kamzolov and Gasnikov [2020]. Latter approach was a breakthrough in 1990s, it lead to interior-point methods. Summary of the results can be found in books Nesterov and Nemirovski [1994], Nesterov [2018]. This direction is still popular up to this day Dvurechensky and Nesterov [2018], Hildebrand [2020], Doikov and Nesterov [2022b], Nesterov [2022].

As easy-to-scale alternative to second-order methods, first-order algorithms attracted a lot of attention. Many of their aspects have been explored, including strong results in variance reduction (Roux et al. [2012],Gower et al. [2020], Johnson and Zhang [2013], Nguyen et al. [2021, 2017]), preconditioning Jahani et al. [2022], acceleration (Nesterov [2013], d'Aspremont et al. [2021]) distributed/federated computation (Konečný et al. [2016], Chen et al. [2022], Berahas et al. [2016], Takáč et al. [2015], Richtárik and Takáč [2016], Kairouz et al. [2021]), and decentralized computation [Koloskova et al., 2020, Sadiev et al., 2021, Borodich et al., 2021]. However, the convergence of first-order methods always depends on the conditioning of the underlying problem. Improving conditioning is fundamentally impossible without using higher-order information. Removing this conditioning dependence is possible by incorporating information about the Hessian. This results in second-order methods. Their most compelling advantage is that they can converge extremely quickly, usually in just a few iterations.

### 1.2 Newton method: benefits and limitations

One of the most famous algorithms in optimization, Newton method, takes iterates of form

$$x_{k+1} = x_k - \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k). \tag{1}$$

Its iterates satisfy the recursion $\|\nabla f(x_{k+1})\|_2 \leq c \|\nabla f(x_k)\|_2^2$ (for a constant $c > 0$), which means that Newton method converges locally quadratically. However, convergence of Newton method is limited to only to the neighborhood of the solution. It is well-known that when initialized far from optimum, Newton can diverge, both in theory and practice (Jarre and Toint [2016], Mascarenhas [2007]). We can explain intuition why this happens. Update rule of Newton (1) was chosen to minimize right hand side of Taylor approximation

$$f(y) \approx Q_f(y; x) \stackrel{\text{def}}{=} f(x) + \langle \nabla f(x), y - x \rangle + \frac{1}{2} \left\langle \nabla^2 f(x)(y - x), y - x \right\rangle. \tag{2}$$

The main problem is that Taylor approximation is not an upper bound, and therefore, global convergence of Newton method is not guaranteed.

### 1.3 Towards a fast globally convergent Newton method

Even though second-order algorithms with superlinear local convergence rates are very common, global convergence guarantees of any form are surprisingly rare. Many papers proposed globalization strategies, essentially all of them require some combination of the following: line-search, trust regions, damping/truncation, regularization. Some popular globalization strategies show non-increase of functional value during the training. However, this turned out to be insufficient for convergence to the optimum. Jarre and Toint [2016], Mascarenhas [2007] designed simple functions (strictly

convex with compact level sets) so that Newton method with Armijo stepsizes does not converge to the optimum. To this day, virtually all known global convergence guarantees are for regularized Newton methods with, which can be written as

$$x_{k+1} = x_k - \alpha_k \left(\nabla^2 f(x_k) + \lambda_k \mathbf{I}\right)^{-1} \nabla f(x_k), \tag{3}$$

where $\lambda_k \geq 0$. Parameter $\lambda_k$ is also known as Levenberg-Marquardt regularization [Moré, 1978], which was first introduced for a nonlinear least-squares objective. For simplicity, we disregard differences in the objectives for the literature comparison. Motivation behind update (3) is to replace Taylor approximation in (2) by an upper bound. The first method with proven global convergence rate $\mathcal{O}\left(k^{-2}\right)$ is Cubic Newton method [Nesterov and Polyak, 2006] for function $f$ with Lipschitz-continuous Hessian,

$$\|\nabla^2 f(x) - \nabla^2 f(y)\|_2 \leq L_2 \|x - y\|_2. \tag{4}$$

Under this condition, one can upper bound of Taylor approximation eq. (2) as

$$f(y) \leq Q_f(y; x) + \frac{L_2}{6}\|y - x\|_2^3. \tag{5}$$

Next iterate of Cubic Newton can be written as a minimizer of right hand side of (5)[1]

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{E}} \left\{ Q_f(y; x_k) + \frac{L_2}{6}\|y - x_k\|_2^3 \right\}. \tag{6}$$

For our newly-proposed algorithm AICN (Algorithm 1), we are using almost identical step[1] [2]

$$x_{k+1} = \operatorname*{argmin}_{y \in \mathbb{E}} \left\{ Q_f(y; x_k) + \frac{L_{\text{semi}}}{6}\|y - x_k\|_{x_k}^3 \right\}. \tag{7}$$

The difference between the update of Cubic Newton and AICN is that we measure the cubic regularization term in the local Hessian norms. This seemingly negligible perturbation turned out to be of a great significance for two reasons **a)** model in (7) is affine-invariant, **b)** surprisingly, the next iterate of (7) lies in the direction of Newton method step and is obtainable without regularizer $\lambda_k$ (AICN just needs to set stepsize $\alpha_k$). We elaborate on both of these points later in the paper.

Cubic Newton method (6) can be equivalently expressed in form (3) with $\alpha_k = 1$ and $\lambda_k = L_2\|x_k - x_{k+1}\|_2$. However, since such $\lambda_k$ depends on $x_{k+1}$, resulting algorithm requires additional subroutine for solving its subproblem each iteration. Next work showing convergence rate of regularized Newton method [Polyak, 2009] avoided implicit steps by choosing $\lambda_k \propto \|\nabla f(x_k)\|_2$. However, this came with a trade-off for slower convergence rate, $\mathcal{O}\left(k^{-1/4}\right)$. Finally, Mishchenko [4/2021] (see also the work of Doikov and Nesterov [12/2021]) improved upon both of these works by using explicit regularization $\alpha_k = 1$, $\lambda_k \propto \sqrt{L_2\|\nabla f(x_k)\|_2}$, and proving global rate $\mathcal{O}\left(k^{-2}\right)$.

## 2 Contributions

### 2.1 AICN as a damped Newton method

In this work, we investigate global convergence for most basic globalization strategy, stepsized Newton method without any regularizer ($\lambda_k = 0$). This algorithm is also referred as damped (or truncated) Newton method; it can be written as

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1} \nabla f(x_k).$$

Resulting algorithm was investigated in detail as an interior-point method. Nesterov [2018] shows quadratic local convergence for stepsizes $\alpha_1 \stackrel{\text{def}}{=} \frac{1}{1+G_1}, \alpha_2 \stackrel{\text{def}}{=} \frac{1+G_1}{1+G_1+G_1^2}$, where[3][4]

[1] Where $\mathbb{E}$ a $d$-dimensional Euclidean space, defined in Section 2.3.

[2] Function $f$ is $L_{\text{semi}}$-semi-strongly self-concordant (Definition 3). Instead of $L_{\text{semi}}$, we will use its upper bound $L_{\text{est}}$, $L_{\text{est}} \geq L_{\text{semi}}$.

[3] Function $f$ is $L_{\text{sc}}$-self-concordant (Definition 1).

[4] Dual norm $\|\nabla f(x_k)\|_{x_k}^* = \left\langle \nabla f(x_k), \nabla^2 f(x_k)^{-1} \nabla f(x_k) \right\rangle$ is defined in Section 2.3.

Table 1: A summary of regularized Newton methods with global convergence guarantees. We consider algorithms with updates of form $x_{k+1} = x_k - \alpha_k \left(\nabla^2 f(x_k) + \lambda_k \mathbf{I}\right)^{-1} \nabla f(x_k)$. For simplicity of comparison, we disregard differences in objectives and assumptions. We assume $L_2$-smoothness of Hessian, $L_{\text{semi}}$-semi-strong self-concordance, convexity (Definition 3), $\mu$-strong convexity locally and bounded level sets. For regularization parameter holds $\lambda_k \geq 0$ and stepsize satisfy $0 < \alpha_k \leq 1$. We highlight the best know rates in blue.

| **Algorithm** | **Regularizer** $\lambda_k \propto$ | **Stepsize** $\alpha_k =$ | **Affine**[(1)] **invariant? (alg., ass., rate)** | **Avoids line search?** | **Global convergence rate** | **Local**[(1)] **convergence exponent** | **Reference** |
|---|---|---|---|---|---|---|---|
| Newton | 0 | 1 | (✓, ✗, ✗) | ✓ | ✗ | 2 | Kantorovich [1948] |
| Newton | 0 | 1 | (✓,✓,✓) | ✓ | ✗ | 2 | Nesterov and Nemirovski [1994] |
| Damped Newton B | 0 | $\frac{1}{1+G_1}$ [(4)] | (✓,✓,✓) | ✓ | $\mathcal{O}\left(k^{-\frac{1}{2}}\right)$ | 2 | Nesterov [2018, (5.1.28)] |
| Damped Newton C | 0 | $\frac{1+G_1}{1+G_1+G_1^2}$ [(4)] | (✓,✓,✓) | ✓ | ✗ | 2 | Nesterov [2018, $(5.2.1)_C$] |
| Cubic Newton | $L_2\|x_{k+1} - x_k\|_2$ | 1 | (✗, ✗, ✗) | ✗ | $\mathcal{O}\left(k^{-2}\right)$ | 2 | Nesterov and Polyak [2006], Griewank [1981b], Doikov and Nesterov [2022a] |
| Locally Reg. Newton | $\|\nabla f(x_k)\|_2$ | 1 | (✗, ✗, ✗) | ✓ | ✗ | 2 | Polyak [2009] |
| Globally Reg. Newton | $\|\nabla f(x_k)\|_2$ | $\frac{\mu+\|\nabla f(x_k)\|_2}{L_1}$ [(3)] | (✗, ✗, ✗) | ✗ | $\mathcal{O}\left(k^{-\frac{1}{4}}\right)$ | 2 | Polyak [2009] |
| Globally Reg. Newton | $\sqrt{L_2\|\nabla f(x_k)\|_2}$ | 1 | (✗, ✗, ✗) | ✓ | $\mathcal{O}\left(k^{-2}\right)$ | $\frac{3}{2}$ | Mishchenko [4/2021] Doikov and Nesterov [12/2021] |
| **AIC Newton** (Algorithm 1) | 0 | $\frac{-1+\sqrt{1+2G}}{G}$ [(4)] | (✓, ✓, ✓) | ✓ | $\mathcal{O}\left(k^{-2}\right)$ | 2 | **This work** |

[(1)] In triplets, we report whether algorithm, used assumptions, convergence rate are affine-invariant, respectively.
[(2)] For a Lyapunov function $\Phi^k$ and a constant $c$, we report exponent $\beta$ of $\Phi(x_{k+1}) \leq c\Phi(x_k)^\beta$.
[(3)] $f$ has $L_1$-Lipschitz continuous gradient.
[(4)] For simplicity, we denote $G_1 \stackrel{\text{def}}{=} L_{\text{sc}}\|\nabla f(x_k)\|^*_{x_k}$ and $G \stackrel{\text{def}}{=} L_{\text{semi}}\|\nabla f(x_k)\|^*_{x_k}$ (for $L_{\text{est}} \leftarrow L_{\text{semi}}$).

$G_1 \stackrel{\text{def}}{=} L_{\text{sc}}\|\nabla f(x_k)\|^*_{x_k}$. Our algorithm AICN is also damped Newton method with stepsize $\alpha = \frac{-1+\sqrt{1+2G}}{G}$, where[24] $G \stackrel{\text{def}}{=} L_{\text{semi}}\|\nabla f(x_k)\|^*_{x_k}$. Mentioned stepsizes $\alpha_1, \alpha_2, \alpha$ share two characteristics. Firstly, all of them depends on gradient computed in the dual norm and scaled by a smoothness constant ($G_1$ or $G$). Secondly, all of these stepsizes converge to 1 from below (for $\hat{\alpha} \in \{\alpha_1, \alpha_2, \alpha\}$ holds $0 < \hat{\alpha} \leq 1$ and $\lim_{x \to x_*} \hat{\alpha} = 1$). Our algorithm uses stepsize bigger by orders of magnitude (see Figure 3 in Appendix A for detailed comparison). The main difference between already established stepsizes $\alpha_1, \alpha_2$ and our stepsize $\alpha$ are resulting global convergence rates. While stepsize $\alpha_2$ does not lead to a global convergence rate, and $\alpha_1$ leads to rate $\mathcal{O}\left(k^{-1/2}\right)$, our stepsize $\alpha$ leads to a significantly faster, $\mathcal{O}\left(k^{-2}\right)$ rate. Our rate matches best known global rates for regularized Newton methods. We manage to achieve these results by carefully choosing assumptions. While rates for $\alpha_1$ and $\alpha_2$ follows from standard self-concordance, our assumptions are a consequence of a slightly stronger version of self-concordance. We will discuss this difference in detail later.

We summarize important properties of regularized Newton methods with fast global convergence guarantees and damped Newton methods in Table 1.

### 2.2 Summary of contributions

To summarize novelty in our work, we present a novel algorithm AICN. Our algorithm can be interpreted in two viewpoints **a)** as a regularized Newton method (version of Cubic Newton method), **b)** as a damped Newton method. AICN enjoys the best properties of these two worlds:

- **Fast global convergence**: AICN converges globally with rate $\mathcal{O}\left(k^{-2}\right)$ (Theorem 2, 4), which matches state-of-the-art global rate for all regularized Newton methods. Furthermore, it is the first such rate for Damped Newton method.
- **Fast local convergence:** In addition to the fast global rate, AICN decreases gradient norms locally in quadratic rate (Theorem 3). This result matches the best-known rates for both regularized Newton algorithms and damped Newton algorithms.

- **Simplicity:** Previous works on Newton regularizations can be viewed as a popular global-convergence fix for the Newton method. We propose an even simpler fix in the form of a stepsize schedule (Section 3).
- **Implementability:** Step of AICN depends on a smoothness constant $L_{\text{semi}}$ (Definition 3). Given this constant, next iterate of AICN can be computed directly.

  This is improvement over Cubic Newton [Nesterov and Polyak, 2006], which for a given constant $L_2$ needs to run **line-search** subroutine each iteration to solve its subproblem
- **Improvement:** Avoiding latter subroutine yields theoretical improvements. If we compute matrix inverses naively, iteration cost of AICN is $\mathcal{O}(d^3)$ (where $d$ is a dimension of the problem), which is improvement over $\mathcal{O}(d^3 \log \varepsilon^{-1})$ iteration cost of Cubic Newton [Nesterov and Polyak, 2006].
- **Practical performance:** We show that in practice, AICN outperforms all algorithms sharing same convergence guarantees: Cubic Newton [Nesterov and Polyak, 2006] and Globally Regularized Newton [Mishchenko, 4/2021] and Doikov and Nesterov [12/2021], and fixed stepsize Damped Newton method (Section 5).
- **Geometric properties:** We analyze AICN under more geometrically natural assumptions. Instead of smoothness, we use a version of self-concordance (Section 3.1), which is invariant to affine transformations and hence also to a choice of a basis. AICN preserves affine-invariance obtained from assumptions throughout the convergence. In contrast, Cubic Newton uses base-dependent $l_2$ norm and hence depends on a choice of a basis. This represents an extra layer of complexity.
- **Alternative analysis:** We also provide alternative analysis under weaker assumptions (Appendix C).

The rest of the paper is structured as follows. In Section 2.3 we introduce our notation. In Section 3, we discuss algorithm AICN, affine-invariant properties and self-concordance. In Sections 4.1 and 4.2 we show global and local convergence guarantees, respectively. In Section 5 we present an empirical comparison of AICN with other algorithms sharing fast global convergence.

### 2.3 Minimization problem & notation

In the paper, we consider a $d$-dimensional Euclidean space $\mathbb{E}$. Its dual space, $\mathbb{E}^*$, is composed of all linear functionals on $\mathbb{E}$. For a functional $g \in \mathbb{E}^*$, we denote by $\langle g, x \rangle$ its value at $x \in \mathbb{E}$.

We consider the following convex optimization problem:

$$\min_{x \in \mathbb{E}} f(x), \tag{8}$$

where $f(x) \in C^2$ is a convex function with continuous first and second derivatives and positive definite Hessian. We assume that the problem has a unique minimizer $x_* \in \operatorname{argmin}_{x \in \mathbb{E}} f(x)$. Note, that $\nabla f(x) \in \mathbb{E}^*, \quad \nabla^2 f(x) h \in \mathbb{E}^*$. Now, we introduce different norms for spaces $\mathbb{E}$ and $\mathbb{E}^*$. Denote $x, h \in \mathbb{E}, g \in \mathbb{E}^*$. For a self-adjoint positive-definite operator $\mathbf{H} : \mathbb{E} \to \mathbb{E}^*$, we can endow these spaces with conjugate Euclidean norms:

$$\|x\|_{\mathbf{H}} \stackrel{\text{def}}{=} \langle \mathbf{H}x, x \rangle^{1/2}, \; x \in \mathbb{E}, \qquad \|g\|^*_{\mathbf{H}} \stackrel{\text{def}}{=} \left\langle g, \mathbf{H}^{-1} g \right\rangle^{1/2}, \; g \in \mathbb{E}^*.$$

For identity $\mathbf{H} = \mathbf{I}$, we get classical Eucledian norm $\|x\|_{\mathbf{I}} = \langle x, x \rangle^{1/2}$. For local Hessian norm $\mathbf{H} = \nabla^2 f(x)$, we use shortened notation

$$\|h\|_x \stackrel{\text{def}}{=} \left\langle \nabla^2 f(x) h, h \right\rangle^{1/2}, \; h \in \mathbb{E}, \qquad \|g\|^*_x \stackrel{\text{def}}{=} \left\langle g, \nabla^2 f(x)^{-1} g \right\rangle^{1/2}, \; g \in \mathbb{E}^*. \tag{9}$$

Operator norm is defined by

$$\|\mathbf{H}\|_{op} \stackrel{\text{def}}{=} \sup_{v \in \mathbb{E}} \frac{\|\mathbf{H}v\|^*_x}{\|v\|_x}, \tag{10}$$

for $\mathbf{H} : \mathbb{E} \to \mathbb{E}^*$ and a fixed $x \in \mathbb{E}$. If we consider a specific case $\mathbb{E} \leftarrow \mathbb{R}^d$, then $\mathbf{H}$ is a symmetric positive definite matrix.

## 3 New Algorithm: Affine-Invariant Cubic Newton

Finally, we are ready to present algorithm AICN. It is damped Newton method with updates

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1} \nabla f(x_k), \tag{11}$$

with stepsize

$$\alpha_k \stackrel{\text{def}}{=} \frac{-1+\sqrt{1+2L_{\text{est}}\|\nabla f(x_k)\|^*_{x_k}}}{L_{\text{est}}\|\nabla f(x_k)\|^*_{x_k}},$$

as summarized in Algorithm 1. Stepsize satisfy $\alpha_k \leq 1$ (from AG inequality, (45)). Also $\lim_{x_k \to x_*} \alpha_k = 1$, hence (11) converges to Newton method. Next, we are going to discuss geometric properties of our algorithm.

**Algorithm 1** AICN: Affine-Invariant Cubic Newton

1: **Requires:** Initial point $x_0 \in \mathbb{E}$, constant $L_{\text{est}}$ s.t. $L_{\text{est}} \geq L_{\text{semi}} > 0$
2: **for** $k = 0, 1, 2 \dots$ **do**
3: $\quad \alpha_k = \frac{-1+\sqrt{1+2L_{\text{est}}\|\nabla f(x_k)\|^*_{x_k}}}{L_{\text{est}}\|\nabla f(x_k)\|^*_{x_k}}$
4: $\quad x_{k+1} = x_k - \alpha_k \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$ ▷ Note that $x_{k+1} \stackrel{(17)}{=} S_{f,L_{\text{est}}}(x_k)$.
5: **end for**

### 3.1 Geometric properties: affine invariance

One of the main geometric properties of the Newton method is *affine invariance*, invariance to affine transformations of variables. Let $\mathbf{A} : \mathbb{E} \to \mathbb{E}^*$ be a non-degenerate linear transformation. Consider function $\phi(y) = f(\mathbf{A}y)$. By affine transformation, we denote $f(x) \to \phi(y) = f(\mathbf{A}y), x \to \mathbf{A}^{-1}y$.

**Significance of norms:** Note that local Hessian norm $\|h\|_{\nabla f(x)}$ is affine-invariant because

$$\|z\|^2_{\nabla^2\phi(y)} = \left\langle \nabla^2\phi(y)z, z\right\rangle = \left\langle \mathbf{A}^\top \nabla^2 f(\mathbf{A}y)\mathbf{A}z, z\right\rangle = \left\langle \nabla^2 f(x)h, h\right\rangle = \|h\|^2_{\nabla^2 f(x)},$$

where $h = \mathbf{A}z$. On the other hand, induced norm $\|h\|_{\mathbf{I}}$ is not affine-invariant because

$$\|z\|^2_{\mathbf{I}} = \langle z, z\rangle = \left\langle \mathbf{A}^{-1}h, \mathbf{A}^{-1}h\right\rangle = \|\mathbf{A}^{-1}h\|^2_{\mathbf{I}}.$$

With respect to geometry, the most natural norm is local Hessian norm, $\|h\|_{\nabla f(x)}$. From affine invariance follows that for this norm, the level sets $\left\{y \in \mathbb{E} \mid \|y - x\|^2_x \leq c\right\}$ are balls centered around $x$ (all directions have the same scaling). In comparison, scaling of the $l_2$ norm is dependent on eigenvalues of the Hessian. In terms of convergence, one direction in $l_2$ can significantly dominate others and slow down an algorithm.

**Significance for algorithms:** Algorithms that are not affine-invariant can suffer from chosen coordinate system. This is the case for Cubic Newton, as its model (4) is bound to base-dependent $l_2$ norm. Same is true for any other method regularized with an induced norm $\|h\|_{\mathbf{I}}$. On the other hand, (damped) Newton methods have affine-invariant models, and hence as algorithms independent of the chosen coordinate system. We prove this claim in following lemma (note: $\alpha_k = 1$ and $\alpha_k$ from (11) are affine-invariant).

**Lemma 1** (Lemma 5.1.1 Nesterov [2018]). *Let the sequence $\{x_k\}$ be generated by a damped Newton method with affine-invariant stepsize $\alpha_k$, applied to the function $f$: $x_{k+1} = x_k - \alpha_k \left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$. For function $\phi(y)$, damped Newton method generates $\{y_k\}$: $y_{k+1} = y_k - \alpha_k \left[\nabla^2\phi(y_k)\right]^{-1} \nabla\phi(y_k)$, with $y_0 = \mathbf{A}^{-1}x_0$. Then $y_k = \mathbf{A}^{-1}x_k$.*

### 3.2 Significance in assumptions: self-concordance

We showed that damped Newton methods preserve affine-invariance through iterations. Hence it is more fitting to analyze them under affine-invariant assumptions. Affine-invariant version of smoothness, *self-concordance*, was introduced in Nesterov and Nemirovski [1994].

**Definition 1.** *Convex function $f \in C^3$ is called self-concordant if*

$$|D^3 f(x)[h]^3| \leq L_{sc} \|h\|_x^3, \quad \forall x, h \in \mathbb{E}, \tag{12}$$

*where for any integer $p \geq 1$, by $D^p f(x)[h]^p \stackrel{def}{=} D^p f(x)[h, \ldots, h]$ we denote the $p$-th order directional derivative[a] of $f$ at $x \in \mathbb{E}$ along direction $h \in \mathbb{E}$.*

[a]For example, $D^1 f(x)[h] = \langle \nabla f(x), h \rangle$ and $D^2 f(x)[h]^2 = \langle \nabla^2 f(x) h, h \rangle$.

Both sides of inequality are affine-invariant. This assumption corresponds to a big class of optimization methods called interior-point methods. Self-concordance implies uniqueness of the solution, as stated in following proposition.

**Proposition 1** (Theorem 5.1.16, Nesterov [2018])**.** *Let a self-concordant function $f$ be bounded below. Then it attains its minimum at a single point.*

Rodomanov and Nesterov [2021] introduced stronger version of self-concordance assumption.

**Definition 2.** *Convex function $f \in C^2$ is called strongly self-concordant if*

$$\nabla^2 f(y) - \nabla^2 f(x) \preceq L_{str} \|y - x\|_z \nabla^2 f(w), \quad \forall y, x, z, w \in \mathbb{E}. \tag{13}$$

For our analysis, we introduce definition for functions between self-concordant and strongly self-concordant.

**Definition 3.** *Convex function $f \in C^2$ is called semi-strongly self-concordant if*

$$\left\| \nabla^2 f(y) - \nabla^2 f(x) \right\|_{op} \leq L_{semi} \|y - x\|_x, \quad \forall y, x \in \mathbb{E}. \tag{14}$$

Note that all of the Definitions 1 - 3 are affine-invariant, their respective classes satisfy

*strong self-concordance $\subseteq$ semi-strong self-concordance $\subseteq$ self-concordance.*

Also, for a fixed strongly self-concordant function $f$ and smallest such $L_{\text{sc}}, L_{\text{semi}}, L_{\text{str}}$ holds $L_{\text{sc}} \leq L_{\text{semi}} \leq L_{\text{str}}$.

These notions are related to the convexity and smoothness; strong concordance follows from function $L_2$-Lipschitz continuous Hessian and strong convexity.

**Proposition 2** (Example 4.1 from [Rodomanov and Nesterov, 2021])**.** *Let $\mathbf{H} : \mathbb{E} \to \mathbb{E}^*$ be a self-adjoint positive definite operator. Suppose there exist $\mu > 0$ and $L_2 \geq 0$ such that the function $f$ is $\mu$-strongly convex and its Hessian is $L_2$-Lipschitz continuous* (4) *with respect to the norm $\| \cdot \|_{\mathbf{H}}$. Then $f$ is strongly self-concordant with constant $L_{str} = \frac{L_2}{\mu^{3/2}}$.*

### 3.3 From assumptions to algorithm

From semi-strong self-concordance we can get a second-order bounds on the function and model.

**Lemma 2.** *If $f$ is semi-strongly self-concordant, then*

$$|f(y) - Q_f(y; x)| \leq \frac{L_{semi}}{6} \|y - x\|_x^3, \quad \forall x, y \in \mathbb{E}. \tag{15}$$

*Consequently, we have upper bound for function value in form*

$$f(y) \leq Q_f(y; x) + \frac{L_{semi}}{6} \|y - x\|_x^3. \tag{16}$$

One can show that (16) is not valid for just self-concordant functions. For example, there is no such upper bound for $-\log(x)$. Hence, the semi-strongly self-concordance is significant as an assumption.

We can define iterates of optimization algorithm to be minimizers of the right hand side of (16),

$$S_{f,L_{\text{est}}}(x) \stackrel{\text{def}}{=} x + \operatorname*{argmin}_{h\in\mathbb{E}} \left\{ f(x) + \langle \nabla f(x), h\rangle + \tfrac{1}{2}\left\langle \nabla^2 f(x)h, h\right\rangle + \tfrac{L_{\text{est}}}{6}\|h\|_x^3 \right\}, \tag{17}$$

$$x_{k+1} = S_{f,L_{\text{est}}}(x_k), \tag{18}$$

for an estimate constant $L_{\text{est}} \geq L_{\text{semi}}$. It turns out that subproblem (17) is easy to solve. To get an explicit solution, we compute its gradient w.r.t. $h$. For solution $h^*$, it should be equal to zero,

$$\nabla f(x) + \nabla^2 f(x)h^* + \tfrac{L_{\text{est}}}{2}\|h^*\|_x \nabla^2 f(x)h^* = 0, \tag{19}$$

$$h^* = -\left[\nabla^2 f(x)\right]^{-1} \nabla f(x) \cdot \left(\tfrac{L_{\text{est}}}{2}\|h^*\|_x + 1\right)^{-1}. \tag{20}$$

We get that step (18) has the same direction as a Newton method and is scaled by $\alpha_k = \left(\frac{L_{\text{est}}}{2}\|h^*\|_x + 1\right)^{-1}$. Now, we substitute $h^*$ from (20) to (19)

$$\nabla f(x) - \nabla f(x)\alpha_k - \tfrac{L_{\text{est}}}{2}\left\langle \nabla f(x), \left[\nabla^2 f(x)\right]^{-1}\nabla f(x)\right\rangle^{1/2} \nabla f(x)\alpha_k^2 = 0,$$

$$\nabla f(x)\left(1 - \alpha_k - \tfrac{L_{\text{est}}}{2}\|\nabla f(x)\|_x^* \alpha_k^2\right) = 0. \tag{21}$$

We solve the quadratic equation (21) for $\alpha_k$, and obtain explicit formula for stepsizes of AICN, as (11). We formalize this connection in theorem, for further explanation see proof in Appendix B.

**Theorem 1.** *For $L_{est} \geq L_{semi}$, update of* AICN (11),

$$x_{k+1} = x_k - \alpha_k \nabla^2 f(x_k)^{-1}\nabla f(x_k), \qquad \textit{where} \qquad \alpha_k = \frac{-1+\sqrt{1+2L_{est}\|\nabla f(x_k)\|_{x_k}^*}}{L_{est}\|\nabla f(x_k)\|_{x_k}^*},$$

*is a minimizer of upper bound* (17), $x_{k+1} = S_{f,L_{est}}(x_k)$.

# 4 Convergence Results

## 4.1 Global convergence

Next, we focus on global convergence guarantees. We will utilize the following assumption:

**Assumption 1** (Bounded level sets)**.** *The objective function $f$ has a unique minimizer $x_*$. Also, the diameter of the level set $\mathcal{L}(x_0) \stackrel{def}{=} \{x \in \mathbb{E} : f(x) \leq f(x_0)\}$ is bounded by a constant $D_2$ as*[a], $\max_{x\in\mathcal{L}(x_0)} \|x - x_*\|_2 \leq D_2 < +\infty$.

[a]We state it in $l_2$ norm for easier verification. In proofs, we use its variant $D$ in Hessian norms, (23).

Our analysis proceeds as follows. Firstly, we show that one step of the algorithm decreases function value, and secondly, we use the technique from [Ghadimi et al., 2017] to show that multiple steps lead to $\mathcal{O}\left(k^{-2}\right)$ global convergence. We start with following lemma.

**Lemma 3** (One step globally)**.** *Let function $f$ be $L_{semi}$-semi-strongly self-concordant, convex with positive-definite Hessian and $L_{est} \geq L_{semi}$. Then for any $x \in \mathbb{E}$, we have*

$$S_{f,L_{est}}(x) \leq \min_{y\in\mathbb{E}} \left\{ f(y) + \tfrac{L_{est}}{3}\|y - x\|_x^3 \right\}. \tag{22}$$

This lemma implies that step (17) decreases function value (take $y \leftarrow x$). Using notation of Assumption 1, $x_k \in \mathcal{L}(x_0)$ for any $k \geq 0$. Also, setting $y \leftarrow x$ and $x \leftarrow x_*$ in (14) yields $\|x - x_*\|_x \leq \left(\|x - x_*\|_{x_*}^2 + L_{\text{est}}\|x - x_*\|_{x_*}^3\right)^{\frac{1}{2}}$. We denote those distances $D$ and $R$,

$$D \stackrel{\text{def}}{=} \max_{t\in[0;k+1]} \|x_t - x_*\|_{x_t} \qquad \text{and} \qquad R \stackrel{\text{def}}{=} \max_{x\in\mathcal{L}(x_0)} \left(\|x - x_*\|_{x_*}^2 + L_{\text{est}}\|x - x_*\|_{x_*}^3\right)^{\frac{1}{2}}. \tag{23}$$

They are both affine-invariant and $R$ upper bounds $D$. While $R$ depends only on the level set $\mathcal{L}(x_0)$, $D$ can be used to obtain more tight inequalities. We avoid using common distance $D_2$, as $l_2$ norm would ruin affine-invariant properties.

**Theorem 2.** *Let $f(x)$ be a $L_{semi}$-semi-strongly self-concordant convex function with positive-definite Hessian, constant $L_{est}$ satisfy $L_{est} \geq L_{semi}$ and Assumption 1 holds. Then, after $k+1$ iterations of Algorithm 1, we have the following convergence rate:*

$$f(x_{k+1}) - f(x_*) \leq O\left(\frac{L_{est}D^3}{k^2}\right) \leq O\left(\frac{L_{est}R^3}{k^2}\right). \tag{24}$$

Consequently, AICN converges globally with a fast rate $\mathcal{O}\left(k^{-2}\right)$. We can now present local analysis.

### 4.2 Local convergence

For local quadratic convergence are going to utilise following lemmas.

**Lemma 4.** *For convex $L_{semi}$-semi-strongly self-concordant function $f$ and for any $c < 1$ in the neighborhood of solution*

$$x_k \in \left\{x : \|\nabla f(x)\|_x^* \leq \frac{(2c+1)^2 - 1}{2L_{est}}\right\} \text{ holds } \nabla^2 f(x_{k+1})^{-1} \preceq (1-c)^{-2} \nabla^2 f(x_k)^{-1}. \tag{25}$$

Lemma 4 formalizes that a inverse hessians of a self-concordant function around the solution is non-degenerate. With this result, we can show one-step gradient norm decrease.

**Lemma 5** (One step decrease locally)**.** *Let function $f$ be $L_{semi}$-semi-strongly self-concordant and $L_{est} \geq L_{semi}$. If $x_k$ such that* (25) *holds, then for next iterate $x_{k+1}$ of* AICN *holds*

$$\|\nabla f(x_{k+1})\|_{x_k}^* \leq L_{est}\alpha_k^2 \|\nabla f(x_k)\|_{x_k}^{*2} < L_{est}\|\nabla f(x_k)\|_{x_k}^{*2}. \tag{26}$$

*Using Lemma 4, we shift the gradient bound to respective norms,*

$$\|\nabla f(x_{k+1})\|_{x_{k+1}}^* \leq \frac{L_{est}\alpha_k^2}{1-c}\|\nabla f(x_k)\|_{x_k}^{*2} < \frac{L_{est}\alpha_k^2}{1-c}\|\nabla f(x_k)\|_{x_k}^{*2}. \tag{27}$$

*Gradient norm decreases $\|\nabla f(x_{k+1})\|_{x_{k+1}}^* \leq \|\nabla f(x_k)\|_{x_k}^*$ for $\|\nabla f(x_k)\|_{x_k}^* \leq \frac{(2-c)^2-1}{2L_{est}}$.*

As a result, neighbourhood of the local convergence is $\left\{x : \|\nabla f(x)\|_x^* \leq \min\left[\frac{(2-c)^2-1}{2L_{\text{est}}}; \frac{(2c+1)^2-1}{2L_{\text{est}}}\right]\right\}$. Maximizing by $c$, we get $c = 1/3$ and neighrbourhood $\left\{x : \|\nabla f(x)\|_x^* \leq \frac{8}{9L_{\text{est}}}\right\}$. One step of AICN decreases gradient norm quadratically, multiple steps leads to following decrease.

**Theorem 3** (Local convergence rate)**.** *Let function $f$ be $L_{semi}$-semi-strongly self-concordant, $L_{est} \geq L_{semi}$ and starting point $x_0$ be in the neighborhood of the solution such that $\|\nabla f(x_0)\|_{x_0}^* \leq \frac{8}{9L_{est}}$. For $k \geq 0$, we have quadratic decrease of the gradient norms,*

$$\|\nabla f(x_k)\|_{x_k}^* \leq \left(\tfrac{3}{2}L_{est}\right)^k \left(\|\nabla f(x_0)\|_{x_0}^*\right)^{2^k}. \tag{28}$$

## 5 Numerical Experiments

In this section, we evaluate proposed AICN (Algorithm 1) algorithm on the logistic regression task and second-order lower bound function. We compare it with regularized Newton methods sharing fast global convergence guarantees: Cubic Newton method [Nesterov and Polyak, 2006], and Globally Regularized Newton method ([Mishchenko, 4/2021, Doikov and Nesterov, 12/2021]) with $L_2$-constant. Because AICN has a form of a damped Newton method, we also compare it with Damped Newton with fixed (tuned) stepsize $\alpha_k = \alpha$. We report decrease in function value $f(x_k)$, function value suboptimality $f(x_k) - f(x_*)$ with respect to iteration and time.

### 5.1 Logistic regression

For first part, we solve the following empirical risk minimization problem:

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) = \frac{1}{m}\sum_{i=1}^m \log\left(1 - e^{-b_i a_i^\top x}\right) + \frac{\mu}{2}\|x\|_2^2 \right\},$$

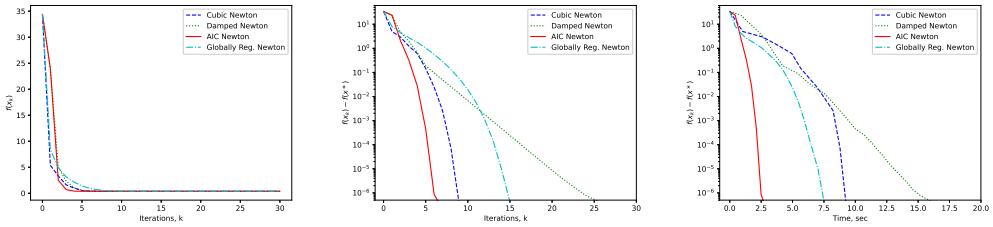


Figure 1: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on *a9a* dataset.

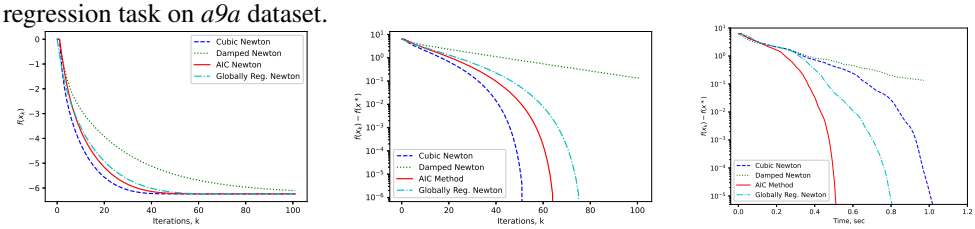


Figure 2: Comparison of regularized Newton methods and Damped Newton method for *second-order lower bound function*.

where $\{(a_i, b_i)\}_{i=1}^m$ are given data samples described by features $a_i$ and class $b_i \in \{-1, 1\}$.

In Figure 1, we consider task of classification images on dataset *a9a* [Chang and Lin, 2011]. Number of features for every data sample is $d = 123$, $m = 20000$. We take staring point $x_0 \stackrel{\text{def}}{=} 10[1, 1, \ldots, 1]^\top$ and $\mu = 10^{-3}$. Our choice is differ from $x_0 = 0$ (equal to all zeroes) to show globalisation properties of methods ($x_0 = 0$ is very close to the solution, as Newton method converges in 4 iterations). Parameters of all methods are fine-tuned, we choose parameters $L_{\text{est}}, L_2, \alpha$ (of AICN, Cubic Newton, Damped Newton, resp.) to largest values having monotone decrease in reported metrics. Fine-tuned values are $L_{\text{est}} = 0.97$ $L_2 = 0.000215$, $\alpha = 0.285$. Figure 1 demonstrates that AICN converges slightly faster than Cubic Newton method by iteration, notably faster than Globally Regularized Newton and significantly faster than Damped Newton. AICN outperforms every method by time.

### 5.2 Second-order lower bound function

For second part we solve the following minimization problem:

$$\min_{x \in \mathbb{R}^d} \left\{ f(x) = \frac{1}{d} \sum_{j=1}^{d} \left| [\mathbf{A}x]_j \right|^3 - x_1 + \frac{\mu}{2}\|x\|_2^2 \right\}, \quad \text{where} \quad \mathbf{A} = \begin{pmatrix} 1 & -1 & 0 & \ldots & 0 \\ 0 & 1 & -1 & \ldots & 0 \\ & & \ldots & \ldots & \\ 0 & 0 & \ldots & 0 & 1 \end{pmatrix}.$$

This function is a lower bound for a class of functions with Lipschitz continuous Hessian (4) with additional regularization [Nesterov and Polyak, 2006, Nesterov, 2021a]. In Figure 2, we take $d = 20$, $x_0 = 0$ (equal to all zeroes). Parameters $L_{\text{est}}, L_2, \alpha$ are fine-tuned to largest values having monotone decrease in reported metrics: $L_{\text{est}} = 662$ $L_2 = 0.662$, $\alpha = 0.0172$. Figure 2 demonstrates that AICN converges slightly slower than Cubic Newton method, slightly faster than Globally Regularized Newton, and significantly faster than Damped Newton. ain outperforms every method by time. More experiments are presented in Appendix A. Note, that the iteration of the Cubic Newton method needs an additional line-search, so one iteration of Cubic Newton is computationally harder than one iteration of AICN. More experiments are presented in Appendix A.


### Acknowledgement

The work of D. Pasechnyuk and A. Gasnikov was supported by a grant for research centers in the field of artificial intelligence, provided by the Analytical Center for the Government of the RF in accordance with the subsidy agreement (agreement identifier 000000D730321P5Q0002) and the agreement with the Ivannikov Institute for System Programming of the RAS dated November 2, 2021 No. 70-2021-00142.

## References


Artem Agafonov, Dmitry Kamzolov, Pavel Dvurechensky, and Alexander Gasnikov. Inexact tensor methods and their application to stochastic convex optimization. *arXiv preprint arXiv:2012.15636*, 2020.

Artem Agafonov, Dmitry Kamzolov, Rachael Tappenden, Alexander Gasnikov, and Martin Takáč. Flecs: A federated learning second-order framework via compression and sketching. *arXiv preprint arXiv:2206.02009*, 2022.

Albert S Berahas, Jorge Nocedal, and Martin Takáč. A multi-batch L-BFGS method for machine learning. In *The Thirtieth Annual Conference on Neural Information Processing Systems (NIPS)*, 2016.

Antoine Bordes, Léon Bottou, and Patrick Gallinari. SGD-QN: careful quasi-Newton stochastic gradient descent. *Journal of Machine Learning Research*, 10:1737–1754, 2009.

Ekaterina Borodich, Aleksandr Beznosikov, Abdurakhmon Sadiev, Vadim Sushko, Nikolay Savelyev, Martin Takáč, and Alexander Gasnikov. Decentralized personalized federated min-max problems. *arXiv preprint arXiv:2106.07289*, 2021.

Richard H Byrd, Jorge Nocedal, and Yoram Singer. A stochastic quasi-Newton method for large-scale optimization. *SIAM Journal on Optimization*, 26(2):1008–1031, 2016.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):1–27, 2011.

Yicheng Chen, Rick S Blum, Martin Takáč, and Brian M Sadler. Distributed learning with sparsified gradient differences. *IEEE Journal of Selected Topics in Signal Processing*, 2022.

Andrew R Conn, Nicholas IM Gould, and Philippe L Toint. *Trust Region Methods*. SIAM, 2000.

Rixon Crane and Fred Roosta. DINGO: Distributed Newton-type method for gradient-norm optimization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 9498–9508. Curran Associates, Inc., 2019. URL `https://proceedings.neurips.cc/paper/2019/file/9718db12cae6be37f7349779007ee589-Paper.pdf`.

Alexandre d'Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. January 2021.

Li Deng. The MNIST database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Nikita Doikov and Yurii Nesterov. Gradient regularization of Newton method with Bregman distances. *arXiv preprint arXiv:2112.02952*, 12/2021.

Nikita Doikov and Yurii Nesterov. Local convergence of tensor methods. *Mathematical Programming*, 193:315–336, 2022a.

Nikita Doikov and Yurii Nesterov. Affine-invariant contracting-point methods for convex optimization. *Mathematical Programming*, pages 1–23, 2022b.

Nikita Doikov and Peter Richtárik. Randomized block cubic Newton method. In Jennifer Dy and Andreas Krause, editors, *The 35th International Conference on Machine Learning (ICML)*, volume 80 of *Proceedings of Machine Learning Research*, pages 1290–1298, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL `http://proceedings.mlr.press/v80/doikov18a.html`.

Pavel Dvurechensky and Yurii Nesterov. Global performance guarantees of second-order methods for unconstrained convex minimization. Technical report, CORE, 2018.

Alexander Gasnikov, Pavel Dvurechensky, Eduard Gorbunov, Evgeniya Vorontsova, Daniil Selikhanovych, César A Uribe, Bo Jiang, Haoyue Wang, Shuzhong Zhang, Sébastien Bubeck, et al. Near optimal methods for minimizing convex functions with lipschitz $p$-th derivatives. In *Conference on Learning Theory*, pages 1392–1393. PMLR, 2019.

Saeed Ghadimi, Han Liu, and Tong Zhang. Second-order methods with cubic regularization under inexact information. *arXiv preprint arXiv:1710.05782*, 2017.

Robert M Gower, Mark Schmidt, Francis Bach, and Peter Richtárik. Variance-reduced methods for Machine Learning. *Proceedings of the IEEE*, 108(11):1968–1983, 2020.

Robert Ma Gower, Dmitry Kovalev, Felix Lieder, and Peter Richtárik. RSN: Randomized Subspace Newton. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 616–625. Curran Associates, Inc., 2019. URL `http://papers.nips.cc/paper/8351-rsn-randomized-subspace-newton.pdf`.

Robert Mansel Gower, Donald Goldfarb, and Peter Richtárik. Stochastic block BFGS: squeezing more curvature out of data. In *33rd International Conference on Machine Learning*, pages 1869–1878, 2016.

Andreas Griewank. The modification of Newton's method for unconstrained optimization by bounding cubic terms. Technical report, Department of Applied Mathematics and Theoretical Physics, University of Cambridge, 1981a. Technical Report NA/12.

Andreas Griewank. The modification of Newton's method for unconstrained optimization by bounding cubic terms. Technical report, Technical report NA/12, 1981b.

Filip Hanzely, Nikita Doikov, Peter Richtárik, and Yurii Nesterov. Stochastic subspace cubic Newton method. In *37th International Conference on Machine Learning (ICML)*, 2020.

Roland Hildebrand. Optimal step length for the Newton method near the minimum of a self-concordant function. *arXiv preprint arXiv:2003.08650*, 2020.

Rustem Islamov, Xun Qian, and Peter Richtárik. Distributed second order methods with fast rates and compressed communication. In *International Conference on Machine Learning (ICML)*, 2021.

Rustem Islamov, Xun Qian, Slavomír Hanzely, Mher Safaryan, and Peter Richtárik. Distributed Newton-type methods with communication compression and bernoulli aggregation. *arXiv preprint arXiv:2206.03588*, 2022.

Majid Jahani, Sergey Rusakov, Zheng Shi, Peter Richtárik, Michael W Mahoney, and Martin Takáč. Doubly adaptive scaled algorithm for machine learning using second-order information. In *Tenth International Conference on Learning Representations (ICLR 2022)*, 2022.

Florian Jarre and Philippe L Toint. Simple examples for the failure of Newton's method with line search for strictly convex minimization. *Mathematical Programming*, 158(1):23–34, 2016.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in Neural Information Processing Systems*, 26, 2013.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

Dmitry Kamzolov and Alexander Gasnikov. Near-optimal hyperfast second-order method for convex optimization and its sliding. *arXiv preprint arXiv:2002.09050*, 2020.

Leonid Vital'evich Kantorovich. Functional analysis and applied mathematics. *Uspekhi Matematicheskikh Nauk*, 3(6):89–185, 1948.

Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized SGD with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

Jakub Konečný, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated Learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Dmitry Kovalev and Alexander Gasnikov. The first optimal acceleration of high-order methods in smooth convex optimization. *arXiv preprint arXiv:2205.09647*, 2022.

Dmitry Kovalev, Konstanting Mishchenko, and Peter Richtárik. Stochastic Newton and cubic Newton methods with simple local linear-quadratic rates. In *NeurIPS Beyond First Order Methods Workshop*, 2019.

Dmitry Kovalev, Robert M. Gower, Peter Richtárik, and Alexander Rogozin. Fast linear convergence of randomized BFGS. *arXiv preprint arXiv:2002.11337*, 2020.

Walter F Mascarenhas. On the divergence of line search methods. *Computational & Applied Mathematics*, 26(1):129–169, 2007.

Konstantin Mishchenko. Regularized Newton method with global $\wr(1/k^2)$ convergence. *arXiv preprint arXiv 2112.02089*, 4/2021.

Aryan Mokhtari and Alejandro Ribeiro. Global convergence of online limited memory BFGS. *Journal of Machine Learning Research*, 16:3151–3181, 2015. URL `http://arxiv.org/abs/1409.2045`.

Renato DC Monteiro and Benar Fux Svaiter. An accelerated hybrid proximal extragradient method for convex optimization and its implications to second-order methods. *SIAM Journal on Optimization*, 23(2):1092–1125, 2013.

Jorge J Moré. The Levenberg-Marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer, 1978.

Yurii Nesterov. Accelerating the cubic regularization of Newton's method on convex problems. *Mathematical Programming*, 112(1):159–181, 2008.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer, 2013.

Yurii Nesterov. *Lectures on Convex Optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, 2018.

Yurii Nesterov. Implementable tensor methods in unconstrained convex optimization. *Mathematical Programming*, 186(1):157–183, 2021a.

Yurii Nesterov. Inexact high-order proximal-point methods with auxiliary search procedure. *SIAM Journal on Optimization*, 31(4):2807–2828, 2021b.

Yurii Nesterov. Superfast second-order methods for unconstrained convex optimization. *Journal of Optimization Theory and Applications*, 191(1):1–30, 2021c.

Yurii Nesterov. Set-limited functions and polynomial-time interior-point methods. Technical report, 2022.

Yurii Nesterov and Arkadi Nemirovski. Self-concordant functions and polynomial time methods in convex programming, ussr acad. *Sci., Central Economic&Mathematical Institute, Moscow*, 1989.

Yurii Nesterov and Arkadi Nemirovski. *Interior-point polynomial algorithms in Convex programming*. SIAM, 1994.

Yurii Nesterov and Boris T Polyak. Cubic regularization of Newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

Isaac Newton. *Philosophiae naturalis principia mathematica*. Jussu Societatis Regiae ac Typis Josephi Streater, 1687.

Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pages 2613–2621. PMLR, 2017.

Lam M Nguyen, Katya Scheinberg, and Martin Takáč. Inexact SARAH algorithm for stochastic optimization. *Optimization Methods and Software*, 36(1):237–258, 2021.

Mert Pilanci and Martin Wainwright. Newton sketch: A linear-time optimization algorithm with linear-quadratic convergence. *SIAM Journal on Optimization*, 27(1):205–245, 2017. URL `https://arxiv.org/pdf/1505.02250.pdf`.

Roman A Polyak. Regularized Newton method for unconstrained Convex optimization. *Mathematical Programming*, 120(1):125–145, 2009.

Xun Qian, Rustem Islamov, Mher Safaryan, and Peter Richtárik. Basis matters: better communication-efficient second order methods for federated learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

Zheng Qu, Peter Richtárik, Martin Takáč, and Olivier Fercoq. SDNA: Stochastic dual Newton ascent for empirical risk minimization. In *The 33rd International Conference on Machine Learning (ICML)*, pages 1823–1832, 2016.

Joseph Raphson. *Analysis Aequationum Universalis Seu Ad Aequationes Algebraicas Resolvendas Methodus Generalis & Expedita, Ex Nova Infinitarum Serierum Methodo, Deducta Ac Demonstrata*. Th. Braddyll, 1697.

Sashank J. Reddi, Jakub Konečný, Peter Richtárik, Barnabás Póczos, and Alex Smola. AIDE: Fast and communication efficient distributed optimization. *arXiv:1608.06879*, 2016.

Peter Richtárik and Martin Takáč. Distributed coordinate descent method for learning with big data. *Journal of Machine Learning Research*, 17:1–25, 2016.

Anton Rodomanov and Yurii Nesterov. Greedy quasi-Newton methods with explicit superlinear convergence. *SIAM Journal on Optimization*, 31(1):785–811, 2021.

Nicolas Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. *Advances in Neural Information Processing Systems*, 25, 2012.

Abdurakhmon Sadiev, Ekaterina Borodich, Aleksandr Beznosikov, Darina Dvinskikh, Saveliy Chezhegov, Rachael Tappenden, Martin Takáč, and Alexander V Gasnikov. Decentralized personalized federated learning: Lower bounds and optimal algorithm for all personalization modes. *arXiv preprint arXiv:2107.07190*, 2021.

Mher Safaryan, Rustem Islamov, Xun Qian, and Peter Richtárik. FedNL: Making Newton-type methods applicable to federated learning. In *Internatioanl Conference on Machine Learning*, 2022.

Ohad Shamir, Nati Srebro, and Tong Zhang. Communication-efficient distributed optimization using an approximate Newton-type method. In *The 31st International Conference on Machine Learning (ICML)*, volume 32, pages 1000–1008, 2014.

Thomas Simpson. *Essays on several curious and useful subjects, in speculative and mix'd mathematicks. Illustrated by a variety of examples.* Printed by H. Woodfall, jun. for J. Nourse, at the Lamb without Temple-Bar, 1740.

Martin Takáč, Peter Richtárik, and Nathan Srebro. Distributed mini-batch SDCA. *arXiv preprint arXiv:1507.08322*, 2015.

Shusen Wang, Farbod Roosta-Khorasani, Peng Xu, and Michael W. Mahoney. GIANT: Globally improved approximate Newton method for distributed optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Peng Xu, Fred Roosta, and Michael W Mahoney. Newton-type methods for non-convex optimization under inexact Hessian information. *Mathematical Programming*, 187(1):35–70, 2020.

Tjalling J Ypma. Historical development of the Newton–Raphson method. *SIAM Review*, 37(4): 531–551, 1995.

## Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Paper is of theoretical nature.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] We highlighted all assumptions.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix B
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] For source code, see supplementary materials.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Sections, 5, A
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A] We compare deterministic algorithms.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A] We used 0 GPU hours.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] We use publicly available datasets.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

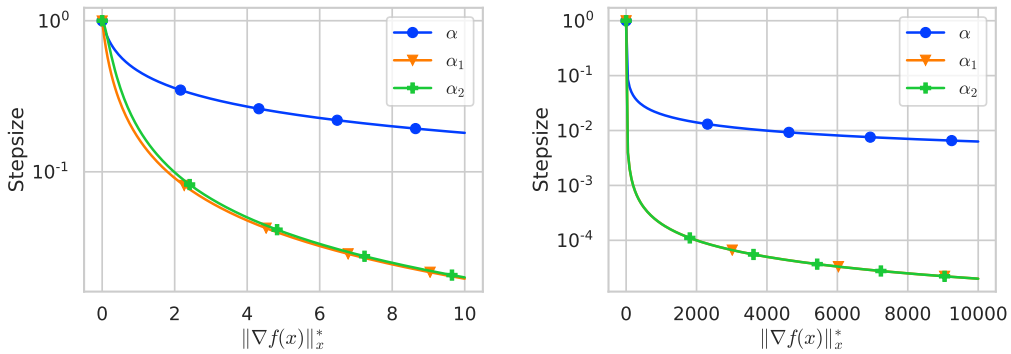


Figure 3: Comparison of stepsizes of affine-invariant damped Newton methods with quadratic local convergence. We compare AICN (blue), and stepsizes from Nesterov [2018] in orange and green. We set $L_{\text{semi}} = L_{\text{sc}} = 5$.

# Appendix

## A Extra comparisons to other methods

### A.1 Damped Newton method stepsize comparison

In Figure 3, we present comparison of stepizes of AICN with other damped Newton methods [Nesterov, 2018]. Our algorithm uses stepsize bigger by orders of magnitude. For reader's convenience, we repeat stepsize choices. For AICN stepsize is $\alpha = \frac{-1+\sqrt{1+2G}}{G}$, where $G \overset{\text{def}}{=} L_{\text{semi}}\|\nabla f(x_k)\|_{x_k}^*$. For damped Newton methods from Nesterov [2018], $\alpha_1 \overset{\text{def}}{=} \frac{1}{1+G_1}, \alpha_2 \overset{\text{def}}{=} \frac{1+G_1}{1+G_1+G_1^2}$, where $G_1 \overset{\text{def}}{=} L_{\text{sc}}\|\nabla f(x_k)\|_{x_k}^*$.

### A.2 Convergence rate comparison under various assumptions

In this subsection we present Table 2 – comparison of AICN to regularized Newton methods under different smoothness and convexity assumptions.

### A.3 Logistic regression experiments

We solve the following minimization problem:

$$\min_{x\in\mathbb{R}^d}\left\{f(x) \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m}\log\left(1-e^{-b_i a_i^\top x}\right)+\frac{\mu}{2}\|x\|_2^2\right\}.$$

To make problem and data balanced, we normalise each data point and get $\|a_i\|_2 = 1$ for every $i \in [1, \dots, m]$. Parameters of all methods are fine-tuned to get the fastest convergence. Note, that it is possible that for bigger $L$ method converges faster in practice.

In Figure 4, we consider classification task on dataset *w8a* [Chang and Lin, 2011]. Number of features for every data sample is $d = 300$, $m = 49749$. We take starting point $x_0 \overset{\text{def}}{=} 8[1, 1, \dots, 1]^\top$ and $\mu = 10^{-3}$. Fine-tuned values are $L_{\text{est}} = 0.6$, $L_2 = 0.0001$, $\alpha = 0.5$. We can see that all methods are very close. Damped Newton has rather big step $0.5$, so it is fast at the beginning but still struggle at the end because of the fixed step-size.

In Figure 5, we consider binary classification task on dataset *MNIST* [Deng, 2012] (one class contains images with 0, another one — all others). Number of features for every data sample is $d = 28^2 = 784$, $m = 60000$. We take starting point $x_0 \overset{\text{def}}{=} 3 \cdot [1, 1, \dots, 1]^\top$ (such that Newton method started from this point diverges) and $\mu = 10^{-3}$. Fine-tuned values are $L_{\text{est}} = 10$, $L_2 = 0.0003$ for Globally Reg.

Table 2: Convergence guarantees under different versions of convexity and smoothness assumptions. For simplicity, we disregard dependence on bounded level set assumptions. All compared assumptions are considered for $\forall x, h \in \mathbb{R}^d$. We highlight the best know rates in blue.

| Algorithm | Strong convexity constant | Smoothness assumption | Global convergence rate | Local [(1)] convergence exponent | Reference |
|---|---|---|---|---|---|
| Damped Newton B | 0[(2)] | self-concordance (Definition 1) | $\mathcal{O}\left(k^{-\frac{1}{2}}\right)$ | 2 | Nesterov [2018, (5.1.28)] |
| Damped Newton C | 0[(2)] | self-concordance (Definition 1) | ✗ | 2 | Nesterov [2018, $(5.2.1)_C$] |
| Cubic Newton | $\mu$ | Lipschitz-continuous Hessian (4) | $\mathcal{O}\left(k^{-2}\right)$ | 2 | Nesterov and Polyak [2006]<br>Doikov and Nesterov [2022a] |
| Cubic Newton | $\mu$-star-convex | Lipschitz-continuous Hessian (4) | $\mathcal{O}\left(k^{-2}\right)$ | $\frac{3}{2}$ | Nesterov and Polyak [2006] |
| Cubic Newton | non-convex, bounded below | Lipschitz-continuous Hessian (4) | $\mathcal{O}\left(k^{-\frac{2}{3}}\right)$ | ✗ | Nesterov and Polyak [2006] |
| Globally Reg. Newton | $\mu$ | Lipschitz-continuous Hessian (4) | $\mathcal{O}\left(k^{-2}\right)$ | $\frac{3}{2}$ | Mishchenko [4/2021]<br>Doikov and Nesterov [12/2021] |
| Globally Reg. Newton | 0 | Lipschitz-continuous Hessian (4) | $\mathcal{O}\left(k^{-2}\right)$ | ✗ | Mishchenko [4/2021]<br>Doikov and Nesterov [12/2021] |
| **AIC Newton** | 0[(2)] | semi-strong self-concordance (Definition 3) | $\mathcal{O}\left(k^{-2}\right)$ | 2 | Theorems 2, 3 |
| **AIC Newton** | $\mu$ | $f(x+h) - f(x) \leq \langle \nabla f(x), h \rangle + \frac{1}{2}\|h\|_x^2 + \frac{L_{\text{alt}}}{6}\|h\|_x^3$ | $\mathcal{O}\left(k^{-2}\right)$ | 2 | Theorems 4, 3 [(3)] |
| **AIC Newton** | 0 | $f(x+h) - f(x) \leq \langle \nabla f(x), h \rangle + \frac{1}{2}\|h\|_x^2 + \frac{L_{\text{alt}}}{6}\|h\|_x^3$ | $\mathcal{O}\left(k^{-2}\right)$ [(4)] | ✗ | Theorem 4 |

[(1)] For a Lyapunov function $\Phi$ and a constant $c > 0$, we report exponent $\beta > 1$ of $\Phi(x_{k+1}) \leq c\Phi(x_k)^{\beta}$. Mark ✗ means that such $\beta, c, \Phi$ are not known.
[(2)] Self-concordance implies strong convexity locally.
[(3)] Under strong convexity, we can prove local convergence analogically to Theorem 3.
[(4)] Convergence to a neighborhood of the solution

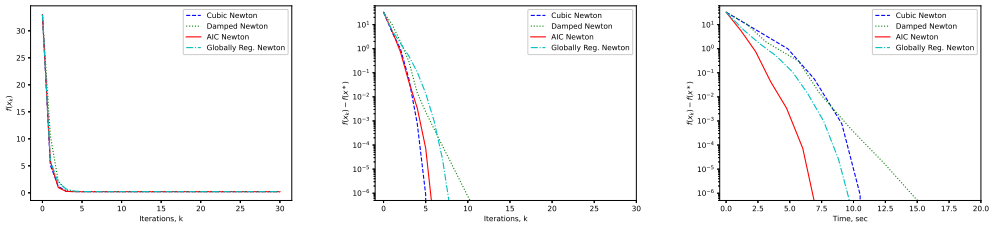


Figure 4: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on *w8a* dataset.

Newton and Cubic Newton, $\alpha = 0.1$. We see that AICN has the same iteration convergence as Cubic Newton but faster by time.

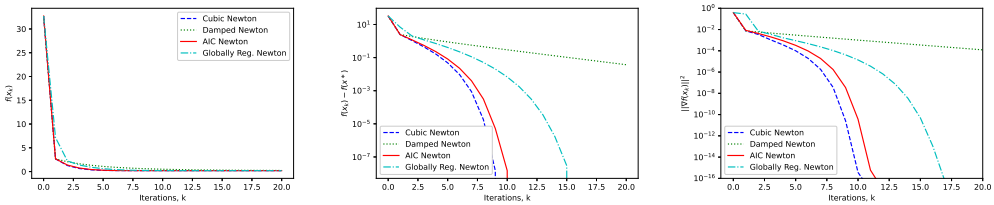


Figure 6: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on *MNIST* dataset (10 models for $i$ vs. other digits problems with argmax aggregation).

In Figure 6, we present the results for multi-class classification problem on dataset *MNIST*. We train 10 different models in parallel, each one for the problem of binary classification distinguishing $i$-th class out of others. Loss on current iteration for the plots is defined as average loss of 10 models.

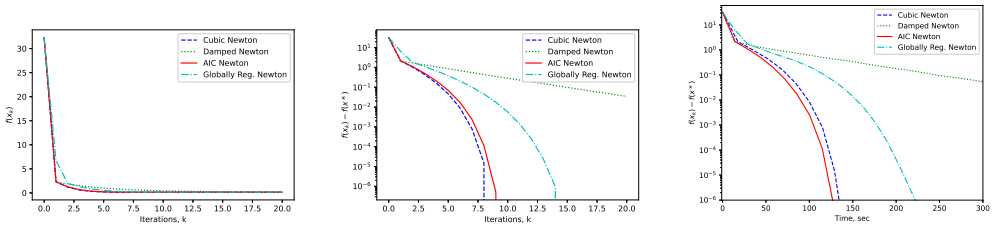


Figure 5: Comparison of regularized Newton methods and Damped Newton method for logistic regression task on *MNIST* dataset (0 vs. all other digits).

Prediction is determined by the maximum "probability" predicted by $i$-th model. The estimates for the parameters of methods are the same as in previous experiment.We see that AICN is the same speed as Cubic Newton and Globally Reg. Newton and much faster than Damped Newton in both value of function and gradient norm.

For normalized problem, we can analytically compute an upper bound for theoretical constant $L_2$

$$\|\nabla^3 f(x)\|_2 \leq L_2.$$

One can show that $L_2 = \frac{\sqrt{3}}{18} \simeq 0.1$. In our experiments, we show that Cubic Newton can work with much lower constants: *madelon* - 0.015, *w8a* - 0.00003, *a9a* - 0.000215. It means that theoretical approximation of the constants can be bad and we have to tune them for all methods.

# B Proofs of Results Appearing in the Paper

In this section, we present proofs of the lemmas and theorems from the main paper's body.

## B.1 Proofs regarding affine invariance (Section 3.1)

*Proof of Lemma 1 (Lemma 5.1.1, Nesterov [2018]).* **[Newton method is affine-invariant]**

Let $y_k = \mathbf{A}^{-1}x_k$ for some $k \geq 0$ and $\alpha_k$ be affine-invariant. Firstly,

$$\begin{aligned} y_{k+1} &= y_k - \alpha_k \left[\nabla^2 \phi(y_k)\right]^{-1} \nabla \phi(y_k) = y_k - \alpha_k \left[\mathbf{A}^\top \nabla^2 f(\mathbf{A}y_k)\mathbf{A}\right]^{-1} \mathbf{A}^\top \nabla f(\mathbf{A}y_k) \\ &= \mathbf{A}^{-1}x_k - \alpha_k \mathbf{A}^{-1} \left[\nabla^2 f(x_k)\mathbf{A}\right]^{-1} \nabla f(x_k) = \mathbf{A}^{-1}x_{k+1}. \end{aligned}$$

Secondly note that $\|\nabla f(x)\|_x^*$ is affine invariant, as

$$\|\nabla g(y_k)\|_{y_k}^* = \nabla g(y_k)^\top \nabla^2 g(y_k)^{-1} \nabla g(y_k) = \nabla f(x_k)^\top \nabla^2 f(x_k)^{-1} \nabla f(x_k) = \|\nabla f(x_k)\|_{x_k}^*.$$

Consequently, stepsizes $\alpha_k$ from AICN (11) and Nesterov [2018] are all affine-invariant. Hence these damped Newton algorithms are affine-invariant. □

*Proof of Lemma 2.* **[Upper bound from from semi-strong self-concordance]**

We rewrite function value approximation from the left hand side as

$$\begin{aligned} f(y) - f(x) - \nabla f(x)[y-x] &= \int_0^1 \left(\nabla f(x + \tau(y-x)) - \nabla f(x)\right)[y-x] d\tau \\ &= \int_0^1 \int_0^\tau \left(\nabla^2 f(x + \lambda(y-x))\right)[y-x]^2 d\lambda d\tau. \end{aligned}$$

Taking its norm, we can finish the proof as

$$\begin{aligned}
&\left| f(y) - f(x) - \nabla f(x)[y-x] - \frac{1}{2}\nabla^2 f(x)[y-x]^2 \right| \\
&= \left| \int_0^1 \int_0^\tau \left(\nabla^2 f(x+\lambda(y-x)) - \nabla^2 f(x)\right)[y-x]^2 d\lambda d\tau \right| \\
&\leq \int_0^1 \int_0^\tau \left| \left(\nabla^2 f(x+\lambda(y-x)) - \nabla^2 f(x)\right)[y-x]^2 \right| d\lambda d\tau \\
&\overset{(14)}{\leq} \int_0^1 \int_0^\tau L_{\text{semi}} \lambda \|y-x\|_x^3 d\lambda d\tau = \frac{L_{\text{semi}}}{6}\|y-x\|_x^3.
\end{aligned}$$

□

*Proof of Theorem 1.* **[Minimizer of the model** (18) **has form of damped Newton method** (11)**]**

Proof is straightforward. To show that AICN model update minimizes $S_{f,L_{\text{est}}}(x)$, we compute the gradient of $S_{f,L_{\text{est}}}(x)$ at next iterate of AICN. Showing that it is 0 concludes $x_{k+1} = S_{f,L_{\text{est}}}(x_k)$.

For simplicity, denote $h \overset{\text{def}}{=} y - x$. We can simplify the implicit update step $S_{f,L_{\text{est}}}(x)$

$$S_{f,L_{\text{est}}}(x) = \operatorname*{argmin}_{y\in\mathbb{E}} \left\{ f(x) + \langle \nabla f(x), y-x\rangle + \frac{1}{2}\left\langle \nabla^2 f(x)(y-x), y-x\right\rangle + \frac{L_{\text{est}}}{6}\|y-x\|_x^3 \right\} \tag{29}$$

$$= x + \operatorname*{argmin}_{h\in\mathbb{E}} \left\{ \langle \nabla f(x), h\rangle + \frac{1}{2}\|h\|_x^2. + \frac{L_{\text{est}}}{6}\|h\|_x^3 \right\}. \tag{30}$$

Taking gradient of the subproblem with respect to $h$,

$$\nabla_h \left( \langle \nabla f(x), h\rangle + \frac{1}{2}\|h\|_x^2 + \frac{L_{\text{est}}}{6}\|h\|_x^3 \right) = \nabla f(x) + \nabla^2 f(x)h + \frac{L_{\text{est}}}{2}\nabla^2 f(x)h\|h\|_x. \tag{31}$$

and setting $h$ according to AICN, $h = -\alpha\nabla^2 f(x)^{-1}\nabla f(x)$, leads to

$$\nabla f(x) - \alpha\nabla f(x) - \frac{L_{\text{est}}}{2}\alpha^2\nabla f(x)\left\|\nabla^2 f(x)^{-1}\nabla f(x)\right\|_x = -\nabla f(x)\left(-1+\alpha+\frac{L_{\text{est}}}{2}\alpha^2\|\nabla f(x)\|_x^*\right). \tag{32}$$

Finally, AICN stepsize $\alpha$ (11) is chosen as a root of quadratic function

$$\frac{L_{\text{est}}}{2}\|\nabla f(x)\|_x^*\alpha^2 + \alpha - 1 = 0, \tag{33}$$

hence the gradient of the (17) at next iterate of AICN is 0. This concludes the proof. □

### B.2 Proof of global convergence (Section 4.1)

*Proof of Lemma 3.* **[One step decrease globally under semi-strong self-concordance]**

This claim follows directly from Lemma 2.

$$\begin{aligned}
f(S_{f,L_{\text{est}}}(x)) &\overset{(16)}{\leq} f(x) + \langle \nabla f(x), S_{f,L_{\text{est}}}(x) - x\rangle + \frac{1}{2}\left\langle \nabla^2 f(x)(S_{f,L_{\text{est}}}(x)-x), S_{f,L_{\text{est}}}(x)-x\right\rangle \\
&\quad + \frac{L_{\text{est}}}{6}\|S_{f,L_{\text{est}}}(x)-x\|_x^3 \\
&\overset{(17)}{=} \min_{y\in\mathbb{E}} \left\{ f(x) + \langle \nabla f(x), y-x\rangle + \frac{1}{2}\left\langle \nabla^2 f(x)(y-x), y-x\right\rangle + \frac{L_{\text{est}}}{6}\|y-x\|_x^3 \right\} \\
&\overset{(15)}{\leq} \min_{y\in\mathbb{E}} \left\{ f(y) + \tfrac{L_{\text{est}}}{3}\|y-x\|_x^3 \right\}.
\end{aligned}$$

□

*Proof of Theorem 2.* **[Global convergence under semi-strong self-concordance]**

We start by taking Lemma 3 for any $t \geq 0$, we obtain

$$\begin{aligned} f(x_{t+1}) &\overset{(22)}{\leq} \min_{y \in \mathbb{E}} \left\{ f(y) + \tfrac{L_{\text{est}}}{3} \|y - x_t\|_{x_t}^3 \right\} \\ &\overset{(23)}{\leq} \min_{\eta_t \in [0,1]} \left\{ f(x_t + \eta_t(x_* - x_t)) + \tfrac{L_{\text{est}}}{3} \eta_t^3 D^3 \right\} \\ &\leq \min_{\eta_t \in [0,1]} \left\{ (1-\eta_t) f(x_t) + \eta_t f(x_*) + \tfrac{L_{\text{est}}}{3} \eta_t^3 D^3 \right\}, \end{aligned}$$

where for the second line we take $y = x_t + \eta_t(x_* - x_t)$ and use convexity of $f$ for the third line. Therefore, subtracting $f(x_*)$ from both sides, we obtain, for any $\eta_t \in [0,1]$

$$f(x_{t+1}) - f(x_*) \leq (1-\eta_t)(f(x_t) - f(x_*)) + \tfrac{L_{\text{est}}}{3} \eta_t^3 D^3 \tag{34}$$

Let us define the sequence $\{A_t\}_{t \geq 0}$ as follows:

$$A_t \overset{\text{def}}{=} \begin{cases} 1, & t = 0 \\ \prod_{i=1}^{t} (1 - \eta_i), & t \geq 1. \end{cases} \tag{35}$$

Then $A_t = (1-\eta_t) A_{t-1}$. Also, we define $\eta_0 \overset{\text{def}}{=} 1.$ Dividing both sides of (34) by $A_t$, we get

$$\begin{aligned} \frac{1}{A_t}\left(f(x_{t+1}) - f(x_*)\right) &\leq \frac{1}{A_t}(1-\eta_t)(f(x_t) - f(x_*)) + \frac{\eta_t^3}{A_t} \frac{L_{\text{est}} D^3}{3} \\ &= \frac{1}{A_{t-1}}(f(x_t) - f(x_*)) + \frac{\eta_t^3}{A_t} \frac{L_{\text{est}} D^3}{3}. \end{aligned} \tag{36}$$

Summing both sides of inequality (36) for $t = 0, \ldots, k$ , we obtain

$$\begin{aligned} \frac{1}{A_k}(f(x_{k+1}) - f(x_*)) &\leq \frac{(1-\eta_0)}{A_0}(f(x_0) - f(x_*)) + \frac{L_{\text{est}} D^3}{3} \sum_{t=0}^{k} \frac{\eta_t^3}{A_t} \\ &\overset{\eta_0 = 1}{=} \frac{L_{\text{est}} D^3}{3} \sum_{t=0}^{k} \frac{\eta_t^3}{A_t}. \end{aligned}$$

As a result,

$$f(x_{k+1}) - f(x_*) \leq \frac{L_{\text{est}} D^3}{3} \sum_{t=0}^{k} \frac{A_k \eta_t^3}{A_t} p \tag{37}$$

To finish the proof, we need to choose $\eta_t$ so that $\sum_{t=0}^{k} \frac{A_k \eta_t^3}{A_t} = O(k^{-2})$. This holds for[5]

$$\eta_t \overset{\text{def}}{=} \frac{3}{t+3}, \quad t \geq 0, \tag{38}$$

as stated in the next lemma.

**Lemma 6** (Properties of $\eta_t$ and $A_t$, [from (2.23), Ghadimi et al. [2017]). *] For choice $\eta_t$ as* (38) *and $A_t$ as* (35) *we have*

$$A_t = \frac{6}{(t+1)(t+2)(t+3)}, \tag{39}$$

$$\sum_{t=1}^{k} \frac{\eta_t^3}{A_t} = \sum_{t=1}^{k} \frac{9(t+1)(t+2)}{2(t+3)^2} \leq \frac{9k}{2}. \tag{40}$$

[5]Note that formula of $\eta_0$ coincides with its definition above.

Plugging Lemma 6 inequalities to (37) concludes the proof of the Theorem 2,

$$f(x_{k+1}) - f(x_*) \leq \frac{6}{(k+1)(k+2)(k+3)} \frac{L_{\text{est}} D^3}{3} \frac{9k}{2} \leq \frac{9 L_{\text{est}} D^3}{k^2} \leq \frac{9 L_{\text{est}} R^3}{k^2}.$$

□

For readers convenience, we include proof of Lemma 6 in Appendix B.4.

### B.3 Proofs of local convergence (Section 4.2)

*Proof of Lemma 4.* **[Strong convexity / Bound on inverse Hessian norm change]**

Claim follows from Theorem 5.1.7 of Nesterov [2018], which states that for $L_{\text{sc}}$-self-concordant function, hence also for $L_{\text{semi}}$-semi-strongly self-concordant function $f$ and $x_k, x_{k+1}$ such that $\frac{L_{\text{semi}}}{2} \|x_{k+1} - x_k\|_{x_k} < 1$ holds

$$\left(1 - \frac{L_{\text{semi}}}{2} \|x_{k+1} - x_k\|_{x_k}\right)^2 \nabla^2 f(x_{k+1}) \preceq \nabla^2 f(x_k) \preceq \left(1 - \frac{L_{\text{semi}}}{2} \|x_{k+1} - x_k\|_{x_k}\right)^{-2} \nabla^2 f(x_{k+1}).$$

Let $c < 1$ be some constant then for updates of AICN in the neighborhood

$$c \geq \frac{L_{\text{est}}}{2} \alpha_k \|\nabla f(x_k)\|^*_{x_k}$$

holds

$$\nabla^2 f(x_{k+1})^{-1} \preceq (1-c)^{-2} \nabla^2 f(x_k)^{-1}. \tag{41}$$

□

In order to prove Lemma 5, we first use semi-strong self-concordance to prove a key inequality – a version of Hessian smoothness, bounding gradient approximation by difference of points.

**Lemma 7.** *For semi-strongly self-concordant function $f$ holds*

$$\left\|\nabla f(y) - \nabla f(x) - \nabla^2 f(x)[y-x]\right\|^*_x \leq \frac{L_{semi}}{2} \|y - x\|^2_x. \tag{42}$$

*Proof of Lemma 7.* **[Local smoothness assumption follows from semi-strong self-concordance]**

We rewrite gradient approximation on the left hand side as

$$\nabla f(y) - \nabla f(x) - \nabla^2 f(x)[y-x] = \int_0^1 \left(\nabla^2 f(x + \tau(y-x)) - \nabla^2 f(x)\right)[y-x] d\tau.$$

Now, we can bound it in dual norm as

$$\begin{aligned}
\left\|\nabla f(y) - \nabla f(x) - \nabla^2 f(x)[y-x]\right\|^*_x &= \left\| \int_0^1 \left(\nabla^2 f(x + \tau(y-x)) - \nabla^2 f(x)\right)[y-x] d\tau \right\|^*_x \\
&\leq \int_0^1 \left\|\left(\nabla^2 f(x + \tau(y-x)) - \nabla^2 f(x)\right)[y-x]\right\|^*_x d\tau \\
&\leq \int_0^1 \left\|\nabla^2 f(x + \tau(y-x)) - \nabla^2 f(x)\right\|_{op} \|y-x\|_x d\tau \\
&\overset{(14)}{\leq} \int_0^1 L_{\text{semi}} \tau \|y-x\|^2_x d\tau = \frac{L_{\text{semi}}}{2} \|y-x\|^2_x,
\end{aligned}$$

where in second inequality we used property of operator norm (10). □

Finally, we are ready to prove one step decrease and the convergence theorem.

*Proof of Lemma 5.* **[One step decrease locally under semi-strong self-concordance]**
We bound norm of $\nabla f(x_{k+1})$ using basic norm manipulation and triangle inequality as

$$\begin{aligned}
\|\nabla f(x_{k+1})\|^*_{x_k} &\overset{(11)}{=} \left\|\nabla f(x_{k+1}) - \nabla^2 f(x_k)(x_{k+1} - x_k) - \alpha_k \nabla f(x_k)\right\|^*_{x_k} \\
&= \left\|\nabla f(x_{k+1}) - \nabla f(x_k) - \nabla^2 f(x_k)(x_{k+1} - x_k) + (1-\alpha_k)\nabla f(x_k)\right\|^*_{x_k} \\
&\leq \left\|\nabla f(x_{k+1}) - \nabla f(x_k) - \nabla^2 f(x_k)(x_{k+1} - x_k)\right\|^*_{x_k} + (1-\alpha_k)\|\nabla f(x_k)\|^*_{x_k}
\end{aligned}$$

Using Lemma 7, we can continue

$$\begin{aligned}
\|\nabla f(x_{k+1})\|^*_{x_k} &\leq \left\|\nabla f(x_{k+1}) - \nabla f(x_k) - \nabla^2 f(x_k)(x_{k+1} - x_k)\right\|^*_{x_k} + (1-\alpha_k)\|\nabla f(x_k)\|^*_{x_k} \\
&\overset{(42)}{\leq} \frac{L_{\text{semi}}}{2}\|x_{k+1} - x_k\|^2_{x_k} + (1-\alpha_k)\|\nabla f(x_k)\|^*_{x_k} \\
&\overset{(11)}{\leq} \frac{L_{\text{semi}}\alpha_k^2}{2}\|\nabla f(x_k)\|^{*2}_{x_k} + (1-\alpha_k)\|\nabla f(x_k)\|^*_{x_k} \\
&\leq \frac{L_{\text{est}}\alpha_k^2}{2}\|\nabla f(x_k)\|^{*2}_{x_k} + (1-\alpha_k)\|\nabla f(x_k)\|^*_{x_k} \\
&= \left(\frac{L_{\text{est}}\alpha_k^2}{2}\|\nabla f(x_k)\|^*_{x_k} - \alpha_k + 1\right)\|\nabla f(x_k)\|^*_{x_k} \\
&\overset{(33)}{=} L_{\text{est}}\alpha_k^2\|\nabla f(x_k)\|^{*2}_{x_k}
\end{aligned}$$

We use Lemma 4 to shift matrix norms.

$$\begin{aligned}
\|\nabla f(x_{k+1})\|^*_{x_{k+1}} &\overset{(41)}{\leq} \frac{1}{1-c}\|\nabla f(x_{k+1})\|^*_{x_k} \\
&\overset{(26)}{\leq} \frac{L_{\text{est}}\alpha_k^2}{1-c}\|\nabla f(x_k)\|^{*2}_{x_k} \\
&< \frac{L_{\text{est}}\alpha_k}{1-c}\|\nabla f(x_k)\|^{*2}_{x_k}.
\end{aligned} \tag{43}$$

We obtain neighborhood of decrease by solving $\frac{L_{\text{est}}\alpha_k}{1-c}\|\nabla f(x_k)\|^*_{x_k} \leq 1$ in $\|\nabla f(x_k)\|^*_{x_k}$.

□

*Proof of Theorem 3.* **[Local convergence under semi-strong self-concordance]**
Let $c = \frac{1}{3}$, then for $\|\nabla f(x_0)\|^*_{x_0} < \frac{8}{9L_{\text{est}}}$, we have $\frac{L_{\text{est}}\alpha_0}{1-c}\|\nabla f(x_0)\|^*_{x_0} \leq 1$ and $c \geq \frac{L_{\text{est}}}{2}\alpha_0\|\nabla f(x_0)\|^*_{x_0}$. Then from Lemma 5 we have guaranteed the decrease of gradients $\|g_{k+1}\|^*_{x_{k+1}} \leq \|g_k\|^*_{x_k} < \frac{8}{9L_{\text{est}}}$. We finish proof by chaining (43) and simplifying with $\alpha_i \leq 1$.

$$\|\nabla f(x_k)\|^*_{x_k} \leq \left(\tfrac{3}{2}L_{\text{est}}\right)^k \left(\prod_{i=0}^{k}\alpha_i^2\right)\left(\|\nabla f(x_0)\|^*_{x_0}\right)^{2^k}. \tag{44}$$

□

### B.4 Technical lemmas

**Lemma 8** (Arithmetic mean – Geometric mean inequality)**.** *For $c \geq 0$ we have*

$$1 + c = \frac{1 + (1+2c)}{2} \overset{AG}{\geq} \sqrt{1+2c}. \tag{45}$$

**Lemma 9** (Jensen for square root)**.** *Function $f(x) = \sqrt{x}$ is concave, hence for $c \geq 0$ we have*

$$\frac{1}{\sqrt{2}}(\sqrt{c}+1) \leq \sqrt{c+1} \qquad \leq \sqrt{c}+1. \tag{46}$$

**Lemma 6** [(2.23) from [Ghadimi et al., 2017]] For

$$\eta_t \stackrel{\text{def}}{=} \frac{3}{t+3}, \quad t \geq 0, \qquad \text{and} \qquad A_t \stackrel{\text{def}}{=} \begin{cases} 1, & t = 0 \\ \prod_{i=1}^{t}(1-\eta_i), & t \geq 1 \end{cases}$$

we have

$$A_t = \frac{6}{(t+1)(t+2)(t+3)} \qquad \text{and} \qquad \sum_{t=1}^{k} \frac{\eta_k^3}{A_t} \qquad = \sum_{t=1}^{k} \frac{9(t+1)(t+2)}{2(t+3)^2} \leq \frac{3k}{2}. \tag{47}$$

*Proof of Lemma 6.* We have

$$A_k = \prod_{t=1}^{k}(1-\eta_t) = \prod_{t=1}^{k} \frac{t}{t+3} = \frac{k!\,3!}{(k+3)!} = 3! \prod_{j=1}^{3} \frac{1}{k+j}, \tag{48}$$

which gives,

$$\sum_{t=0}^{k} \frac{A_k \eta_t^3}{A_t} = \sum_{t=0}^{T} \frac{3^3}{(t+3)^3} \prod_{j=1}^{3} \frac{t+j}{k+j} = 3^3 \prod_{j=1}^{3} \frac{1}{k+j} \sum_{t=0}^{k} \frac{\prod_{j=1}^{3}(t+j)}{(t+3)^3}. \tag{49}$$

The sum is non-decreasing. Indeed, we have

$$1 \leq 1 + \frac{1}{t+3} \leq 1 + \frac{1}{t+j}, \quad \forall j \in \{1,2,3\},$$

and, hence, for all $j \in \{1,2,3\}$,

$$\begin{aligned} \left(1 + \tfrac{1}{t+3}\right)^3 &\leq \prod_{j=1}^{3}\left(1 + \frac{1}{t+j}\right) \\ \Leftrightarrow \quad \left(\tfrac{t+4}{t+3}\right)^3 &\leq \prod_{j=1}^{3} \frac{t+j+1}{t+j} \\ \Leftrightarrow \quad \tfrac{\prod_{j=1}^{3}(t+j)}{(t+3)^3} &\leq \frac{\prod_{j=1}^{3}(t+1+j)}{(t+1+3)^3}. \end{aligned}$$

Thus, we have shown that the summands in the RHS of (49) are growing, whence we get the next upper bound for the sum

$$\begin{aligned} \sum_{t=0}^{k} \frac{A_k \eta_t^3}{A_t} &= 3^3 \prod_{j=1}^{3} \frac{1}{k+j} \sum_{t=0}^{k} \frac{\prod_{j=1}^{3}(t+j)}{(t+3)^3} \\ &\leq 3^3 \prod_{j=1}^{3} \frac{1}{k+j} \cdot (k+1) \cdot \frac{\prod_{j=1}^{3}(k+j)}{(k+3)^3} \leq \frac{(k+1)3^3}{(k+3)^3} \leq O\left(\frac{1}{k^2}\right). \end{aligned} \tag{50}$$

□

## C Global Convergence with weaker assumptions on Self-Concordance

We can prove global convergence to a neighborhood of the solution without using self-concordance directly, just by utilizing the following assumptions:

**Assumption 2** (Convexity)**.** *For function $f$ and any $x, h \in \mathbb{E}$ holds*

$$f(x+h) \geq f(x) + \langle \nabla f(x), h \rangle \tag{51}$$

**Assumption 3** (Hessian smoothness, in Hessian norms)**.** *Objective function $f$ satisfy*

$$f(x+h) - f(x) \leq \langle \nabla f(x), h \rangle + \tfrac{1}{2}\|h\|_x^2 + \tfrac{L_{alt}}{6}\|h\|_x^3, \qquad \forall x, h \in \mathbb{E}. \tag{52}$$

**Lemma 10** (One step decrease globally)**.** *Let Assumption 3 hold and let $L_{est} \geq L_{alt}$. Iterates of* AICN *eq.* (11) *yield function value decrease,*

$$f(x_{k+1}) - f(x_k) \leq \begin{cases} -\frac{1}{2\sqrt{L_{est}}}\|\nabla f(x_k)\|_{x_k}^{*\frac{3}{2}} & \text{if } \|\nabla f(x_k)\|_{x_k}^* \geq \frac{4}{L_{est}} \\ -\frac{1}{4}\|\nabla f(x_k)\|_{x_k}^{*2} & \text{if } \|\nabla f(x_k)\|_{x_k}^* \leq \frac{4}{L_{est}} \\ -\frac{\sqrt{c_1}}{2\sqrt{L_{est}}}\|\nabla f(x_k)\|_{x_k}^{*\frac{3}{2}} & \text{if } \|\nabla f(x_k)\|_{x_k}^* \geq \frac{4c_1}{L_{est}} \text{ and } 0 < c_1 \leq 1 \end{cases}. \tag{53}$$

Decrease of Lemma 10 is tight up to a constant factor. As far as $\|\nabla f(x_k)\|_{x_k}^* \leq \frac{4c_1}{L_{\text{est}}}$, we have functional value decrease as the first line of (53), which leads to $\mathcal{O}\left(k^{-2}\right)$ convergence rate.

We can obtain fast convergence to only neighborhood of solution, because close to the solution, gradient norm is sufficiently small $\|\nabla f(x_k)\|_{x_k}^* \leq \frac{4c_1}{L_{\text{est}}}$ and we get functional value decrease from second line of (53). However, this convergence is slower then $\mathcal{O}\left(k^{-2}\right)$ for $\|\nabla f(x_k)\|_{x_k}^* \approx 0$ and it is insufficient for $\mathcal{O}(k^{-2})$ rate.

Note that third line generalizes first line; we use it to remove a constant factor gap from the neighborhood of fast local convergence.

**Theorem 4** (Global convergence)**.** *Let Assumptions 2, 3, 1 hold, and constants $c_1, L_{est}$ satisfy $0 < c_1 \leq 1, L_{est} \geq L_{alt}$. For $k$ until $\|\nabla f(x_k)\|_{x_k}^* \geq \frac{4c_1}{L_{est}}$,* AICN *has global quadratic decrease, $f(x_k) - f^* \leq \mathcal{O}\left(\frac{L_{est}D^3}{k^2}\right)$.*

Note that the global quadratic decrease of AICN is only to a neighborhood of the solution. However, once AICN gets to this neighborhood, it converges using (faster) local convergence rate (Theorem 3).

### Proofs of global convergence without self-concordance

Throughout the rest of proofs, we simplify expressions by denoting terms

$$g_k \stackrel{\text{def}}{=} \nabla f(x_k) \qquad \text{and} \qquad h_k \stackrel{\text{def}}{=} x_{k+1} - x_k, \tag{54}$$

for which holds

$$h_k = -\alpha_k \nabla^2 f(x_k)^{-1} g_k, \quad g_k = -\tfrac{1}{\alpha_k}\nabla^2 f(x_k) h_k \quad \text{and} \quad \|h_k\|_{x_k} = \sqrt{\alpha}\|g_k\|_{x_k}^* m \tag{55}$$

and also $G_k \stackrel{\text{def}}{=} L_{\text{est}}\|g_k\|_{x_k}^*$.

*Proof of Lemma 10.* We can use Assumption 3 to obtain

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= f(x_k + h_k) - f(x_k) \\ &\stackrel{(52)}{\leq} \langle \nabla f(x_k), h_k \rangle + \frac{1}{2}\|h_k\|_{x_k}^2 + \frac{L_{\text{est}}}{6}\|h_k\|_{x_k}^3 \qquad (56) \\ &\stackrel{(55)}{=} -\alpha_k \|g_k\|_{x_k}^{*2} + \frac{1}{2}\alpha_k^2 \|g_k\|_{x_k}^{*2} + \frac{L_{\text{est}}}{6}\alpha_k^3 \|g_k\|_{x_k}^{*3} \\ &= -\alpha_k \|g_k\|_{x_k}^{*2}\left(1 - \frac{1}{2}\alpha_k - \frac{L_{\text{est}}}{6}\alpha_k^2 \|g_k\|_{x_k}^*\right). \qquad (57) \end{aligned}$$

We can simplify bracket in eq. (57) as

$$\begin{aligned} 1 - \frac{1}{2}\alpha_k - \frac{L_{\text{est}}}{6}\alpha_k^2\|g_k\|_{x_k}^* &= 1 - \frac{1}{2}\frac{\sqrt{1+2G_k} - 1}{G_k} - \frac{G_k}{6}\left(\frac{\sqrt{1+2G_k} - 1}{G_k}\right)^2 \\ &= \frac{4G_k + 1 - \sqrt{1+2G_k}}{6G_k} \stackrel{(45)}{\geq} \frac{1}{2}. \end{aligned}$$

Also, we can simplify the other term in eq. (57) as

$$\alpha_k \|g_k\|_{x_k}^{*2} = \frac{\|g_k\|_{x_k}^*}{L_{\text{est}}} \left(\sqrt{1+2G_k} - 1\right) \frac{\sqrt{1+2G_k}+1}{\sqrt{1+2G_k}+1} = \frac{2\|g_k\|_{x_k}^{*2}}{\sqrt{1+2G_k}+1}$$
$$\overset{(46)}{\geq} \frac{2\|g_k\|_{x_k}^{*2}}{\sqrt{G_k}+1+\frac{1}{\sqrt{2}}} \geq \frac{2\|g_k\|_{x_k}^{*2}}{\sqrt{G_k}+2} \geq \frac{\|g_k\|_{x_k}^{*2}}{\max\left(\sqrt{G_k}, 2\right)},$$

and plug these two result into eq. (57) to obtain first two lines of (53). Third line can be obtained from the first line of (53). For $c_1$ so that $0 < c_1 \leq 1$ and $x_k$ satisfying $\frac{4c_1}{L_{\text{est}}} \leq \|\nabla f(x_k)\|_{x_k}^* < \frac{4}{L_{\text{est}}}$ holds

$$f(x_{k+1}) - f(x_k) \leq -\tfrac{1}{4}\|\nabla f(x_k)\|_{x_k}^{*2} \leq -\tfrac{\sqrt{c_1}}{2\sqrt{L_{\text{est}}}}\|\nabla f(x_k)\|_{x_k}^{*\frac{3}{2}}.$$

□

*Proof of Theorem 4.* As a consequence of convexity (Assumption 2) and bounded level sets (Assumption 1), we have

$$f(x_k) - f^* \leq \langle g_k, x_k - x_* \rangle = \left\langle \nabla^2 f(x_k)^{-1/2} g_k, \nabla^2 f(x_k)^{1/2}(x_k - x_*) \right\rangle \leq \|g_k\|_{x_k}^* \|x_k - x_*\|_{x_k}$$
$$\leq D\|g_k\|_{x_k}^*. \tag{58}$$

Plugging it into eq. (53) (which holds under Assumption 3) and noting that it yields

$$f(x_{k+1}) - f(x_k) \leq -\frac{\sqrt{c_1}}{2\sqrt{L_{\text{est}}}D^{3/2}} \left(f(x_k) - f^*\right)^{3/2}. \tag{59}$$

Denote $\tau \overset{\text{def}}{=} \frac{\sqrt{c_1}}{2\sqrt{L_{\text{est}}}D^{3/2}}$ and $\beta_k \overset{\text{def}}{=} \tau^2(f(x_k) - f^*) \geq 0$. Terms $\beta_k$ satisfy recurrence

$$\beta_{k+1} = \tau^2(f(x_{k+1}) - f^*) \overset{(53)}{\leq} \tau^2(f(x_k) - f^*) - \tau^3(f(x_k) - f^*)^{3/2} = \beta_k - \beta_k^{3/2}.$$

Because $\beta_{k+1} \geq 0$, we have that $\beta_k \leq 1$.

Proposition 1 of Mishchenko [4/2021] shows that the sequence $\{\beta_k\}_{k=0}^{\infty}, 0 \leq \beta_k \leq 1$ decreases as $\mathcal{O}\left(\frac{1}{k^2}\right)$. Let $c_2$ be a constant satisfying $\beta_k \leq \frac{c_2}{k^2}$ for all $k$ ($c_2 \approx 3$ seems to be sufficient). Finally, fol $k \geq \frac{\sqrt{c_2}}{\tau\sqrt{\varepsilon}} = 2\sqrt{\frac{c_2 L_{\text{est}} D^3}{c_1 \varepsilon}} = \mathcal{O}\left(\sqrt{\frac{L_{\text{est}} D^3}{\varepsilon}}\right)$ we have

$$f(x_k) - f^* = \frac{\beta_k}{\tau^2} \leq \frac{c_2}{c_1 k^2 \tau^2} \leq \varepsilon.$$

□