A Review on Efficient Approaches for Escaping Higher Order Saddle Points in Non-convex Optimization

Jier Qiu[1*]
[1]University of Michigan

Abstract

Local search can solve NP problems by maximizing an objective function among several potential solutions. It is widely used in non-convex optimization and is important to many machine learning algorithms, particularly in the training of deep neural networks. However, traditional algorithms often fail to converge to local minimum due to the presence of saddle points in high-dimensional spaces. Often degenerate saddle points exist and the first and second order derivatives cannot distinguish them with local optima. This review analyzes the approaches proposed by Anima Anandkumar and Rong Ge, which used higher order derivatives to escape saddle points effectively. They first proposed a method to guarantee convergence to third order optimum and proposed that finding a fourth order local optima is NP-hard. This paper examines the methodology, compares it with existing approaches, and discusses its extensions and potential future directions in non-convex optimization.

Key words: Non-convex optimization, machine learning, Saddle points

1 Introduction and Problem statement

The rapid increasing application of non-convex optimization in large-scale machine learning applications, such as deep learning, presents significant challenges due to the complexity of the optimization landscape. This complexity is primarily due to the presence of numerous local minima and saddle points, which are especially difficult to distinguish in high-dimensional spaces. Saddle points, in particular, are critical points where the objective function does not improve in all directions, causing algorithms like gradient descent to stop growing.

Recent theoretical research has mostly highlighted the difficulties in achieving global optima in these scenarios. Few attention has been paid to more achievable goals such as attaining local optima, which still remains a challenging task due to the diversity of saddle points in high-dimensional spaces. The set of critical points also consists of saddle points, which possess directions along which the objective value improves. The objective function can be arbitrarily bad at these points, it is important to develop strategies to escape them. This issue is complex by the curse of dimensionality, with the number of saddle points expanding exponentially in many problems of interest, complicating the path to even local optima.

There have been research of optimization methods modifications addressing these challenges. For instance, second-order Hessian information(Nesterov and Polyak, 2006[8]) or noisy stochastic gradient descent(Ge et al., 2015[6]) have been developed to help escape saddle points under certain conditions, known as the strict saddle condition. These approaches rely on the Hessian matrix at the saddle point having at least one strictly negative eigenvalue, which allows the algorithm to find a descent direction away from the saddle point. This is always satisfied in the complete dictionary learning, phase retrieval and orthogonal tensor decomposition problems.(Sun et al., 2015[9])

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

However, these methods fall short in cases without the strict saddle property, where saddle points can pretend to be local minima if only first and second-order information is used. To address this, Anima Anandkumar and Rong Ge proposed an advanced algorithm that extends the notion of second-order optimality to higher order conditions[1]. The method uses third-order derivatives to effectively escape higher order saddle points, ensuring convergence to a third-order local minimum. Moreover, the paper of interest also demonstrate that finding a fourth-order local minimum is NP-hard, underscoring the complexity of the problem.

The findings can be applied for systems with degenerate critical points, where the Hessian matrix is singular and this is often caused due to symmetries in the optimization problems. Singularities also arise in models where the model capacity (e.g. number of neurons in neural networks) exceeds the complexity of the target function. In these models, certain neurons can be eliminated or two neurons can have the same weight(Wei et al., 2008[11]). Intuitively, these models, including multi-layer neural networks and Gaussian mixtures, often exhibit singularities that can slow down suboptimal learning due to the presence of flat regions around singular saddle points, where the gradient and Hessian can not provide a direction to the local optima. By employing higher-order derivatives, the algorithm of interest can distinguish between local optima and deceptive saddle points, offering a robust solution in machine learning optimization.

1.1 Brief Introduction to the Results

A point x is a p^{th} order local minimum if for any nearby point y, $f(x) - f(y) \leq o(\|x - y\|^p)$.

1.1.1 Necessary and Sufficient Condition for Third Order Local Minimum

Theorem 1. *(Informal) There is an algorithm that always converges to a third order local minimum . Also, in polynomial time the algorithm can find a point that is "similar" to a third order local minimum.*

"similar" necessary and sufficient condition for third order local minimum: for a point x, the gradient $\nabla f(x)$ is small, Hessian $\nabla^2 f(x)$ is almost positive semidefinite (p.s.d), and in every subspace where the Hessian is small, the norm of the third order derivatives is also small.

1.1.2 Finding a Fourth Order Local Minimum is NP-hard

Theorem 2. *(Informal) Even for a well-behaved function, it is NP-hard to find a fourth order local minimum*

2 Related Work

The review of related work reveals several approaches to escaping saddle points in non-convex optimization, including incorporating second-order information and trust region method. Newton's method generally converge to any critical points and therefore can't distinguish between local minima and saddle points. However, we can still use Hessian information. We can use directions along negative values of the Hessian matrix whenever gradient descent improvements are small ,which signals the approach towards a critical point.(Frieze et al., 1996[5])

Another approach is to use the trust region method which involves optimizing the second order Taylor's approximation of the objective function in a local neighborhood of the current point. (Dauphin et al., 2014[4], Sun et al., 2015[9]) Intuitively, the objective "switches" smoothly between first order and second order updates.

Nesterov and Polyak([8]) propose adding a cubic regularization term to this Taylor's approximation and show that in each step, this cubic regularized objective can be solved optimally and the algorithm converges to a local optimum in bounded time. There are some related modified algorithms. Baes(2009[2]) generalizes this idea with higher order Taylor expansion, however the optimization problem is intractable even for third order Taylor expansion. Ge et al.(2015[6]) showed that it is possible to escape saddle points in polynomial time in high dimensions,using only first order information with noisy stochastic gradient descent (SGD).

Here we give a brief introduction of some methods mentioned above.

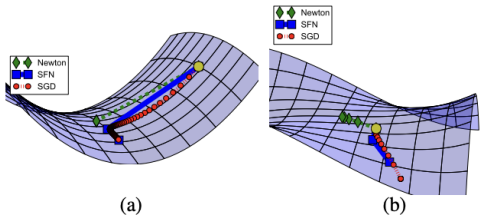

Figure 1: Behaviors of different optimization methods near a saddle point for (a) classical saddle structure $5x^2 - y^2$; (b) monkey saddle structure $x^3 - 3xy^2$. The yellow dot indicates the starting point. SFN stands for the saddle free Newton method proposed by Dauphin et al., 2014[4]

2.1 Hessian Matrix Method

Frieze et al.(1996[5]) introduced a polynomial time algorithm that can learn an arbitrarily oriented cube in n-dimensional space, using uniformly distributed sample points from the cube. The work extended to the broader problem of learning a linear transformation of a product distribution. Suppose x is an n-vector whose coordinates are mutually independent random variables with unknown probability distributions and A is an unknown nonsingular n×n matrix. Then given polynomially many samples of y = Ax, the method is able to learn the columns of A approximately.

The method hinges on analyzing second moments and determining rotations through fourth moments of observed variables, allowing approximation of matrix columns. By using some standard Linear Algebra, we can learn parallelepipeds upto rotations, which only involves analyzing the matrix of second moments of the "observed" variables v. The central problem is determining the rotation. The paper first proved that the maxima (and minima) of the fourth moment function gives the columns of A. Then it showed that the maxima and minima can be found approximately by a nonlinear fourth degree optimization algorithm.

The general problem Statement: There are n real valued random variables x = (x_1, x_2, ...x_n). We are given observations of

$$y = Ax + b \tag{1}$$

where A is an unknown nonsingular matrix and b is an unknown vector. Our aim is to find A, b approximately from polynomial number of samples of y.

Assumption 1. *for all i, E(x_i) = 0 and E(x_i^2) = 1.*

Assumption 2. *We will assume that in fact the variance-covariance matrix of x is the identity.*

Assumption 3. *We will assume a weak form of 4-way independence. I.e., we will assume that the expectation of each monomial of degree 4 in the x_i's is the product of the expectations of each variable to the suitable power. More precisely, we assume that*

$$E(x_i x_j x_k x_l) = Ax + b \tag{2}$$

whenever for any s, x_s occurs an odd number of times in the product $x_i x_j x_k x_l$;
we also assume that $E(x_i^2 x_j^2) = 1$.

Lemma 1. *Suppose we have random variables $x = (x_1, x_2, ...x_n)$ satisfying assumptions 1,2 and 3 above. Suppose R is an orthonormal matrix. Consider the function $F(2)$ (where u is a column vector in R^n) defined by*

$$F(u) = E((u^T Rx)^4) \tag{3}$$

The local maxima (respectively, the local minima) of $F()$ over the unit sphere ($\{u : \|u\| = 1\}$) are precisely the rows of A^{-1} corresponding to i such that $E(x_i^4) > 3$ (respectively, $E(x_i^4) < 3$)

2.2 Trust Region Methods

Dauphin et al., 2014[4] mainly proposed that while doing the widely used Gradient descent or quasi-Newton methods in non-convex optimization problems, difficulty to find the global minimum originates from the proliferation of saddle points, not local minima, especially in high dimensional

Algorithm 1 Approximate saddle-free Newton

Require: Function $f(\theta)$ to minimize
  for $i = 1 \to M$ **do**
    $\mathbf{V} \leftarrow k$ Lanczos vectors of $\frac{\partial^2 f}{\partial \theta^2}$
    $\hat{f}(\alpha) \leftarrow g(\theta + \mathbf{V}\alpha)$
    $|\hat{\mathbf{H}}| \leftarrow \left|\frac{\partial^2 \hat{f}}{\partial \alpha^2}\right|$ by using an eigen decomposition of $\hat{\mathbf{H}}$
    for $j = 1 \to m$ **do**
      $\mathbf{g} \leftarrow -\frac{\partial \hat{f}}{\partial \alpha}$
      $\lambda \leftarrow \arg\min_\lambda \hat{f}(\mathbf{g}(|\hat{\mathbf{H}}| + \lambda \mathbf{I})^{-1})$
      $\theta \leftarrow \theta + \mathbf{g}(|\hat{\mathbf{H}}| + \lambda \mathbf{I})^{-1}\mathbf{V}$
    end for
  end for

Figure 2: Dauphin et al., 2014[4]

problems. High error plateaus around those saddle points can slow down learning and make the saddle points look like local minima. The paper raised awareness of this issue, and proposed the saddle-free Newton method which can rapidly escape high dimensional saddle points and provided numerical evidence that it outperforms quasi-Newton methods in some high dimensional problems involving deep or recurrent networks.

2.3 Nestorov's Cubic Regularization

Nesterov and Polyak([8] proposed a cubic regularization of Newton method in unconstrained minimization problem. The general local convergence results and global worst-case complexity bounds for non-convex cases is significant for the paper of interest[1] to prove analogous results for third order local minimum, which we will see in the following section(**Section 3**).

The algorithm requires the first two order derivatives exist and the following smoothness constraint:

Assumption 4 (Lipschitz-Hessian)**.**

$$\forall x, y, \|\nabla^2 f(x) - \nabla^2 f(y)\| \leq R\|x - y\|.$$

At a point x, the algorithm tries to find a nearby point z that optimizes the degree two Taylor's expansion: $f(x) + \langle f(x), z - x\rangle + \frac{1}{2}(z-x)^\top(\nabla^2 f(x))(z-x)$, with the cubic distance $\frac{R}{6}\|z-x\|^3$ as a regularizer.

The algorithm generates a sequence of points $x^{(0)}, x^{(1)}, x^{(2)}, \ldots$ where $x^{(i+1)} = \text{CubicReg}(x^{(i)})$.

Algorithm 1 CubicReg

Require: function f, current point x, Hessian smoothness R
Ensure: Next point z that satisfies Theorem 3.
  Let $z = \arg\min f(x) + \langle f(x), z - x\rangle + \frac{1}{2}(z-x)^\top(\nabla^2 f(x))(z-x) + \frac{R}{6}\|z-x\|^3$.
  return z

The paper[8] showed that it is possible to solve this optimization problem in polynomial time.

Define $\mu(z)$ to measure how close each point z is to satisfying the second order optimality condition:

Definition 1. $\mu(z) = \max\left\{\sqrt{\frac{1}{R}\|\nabla f(z)\|}, -\frac{2}{3R}\lambda_n \nabla^2 f(z)\right\}$

When $\mu(z) = 0$, $\nabla f(z) = 0$ and $\nabla^2 f(z) \succeq 0$, which satisfies the second order necessary conditions and z is a second order local minimum. When $\mu(z)$ is small, the point z approximately satisfies the second order optimality condition.

Theorem 3. *Suppose $z = CubicRegularize(x)$, then $\|z - x\| \geq \mu(z)$ and $f(z) \leq f(x) - R\|z - x\|^3/12$.*

Theorem 4. *(Convergence)If $f(x)$ is bounded below by $f(x^*)$, then $\lim_{i\to\infty}\mu(x^{(i)})=0$, and for any $t\geq 1$ we have*

$$\min_{1\leq i\leq t}\mu(x^{(i)})\leq\frac{8}{3}\cdot\left(\frac{3(f(x^{(0)})-f(x^*))}{2tR}\right)^{1/3}.$$

Theorem 5. *(A Stronger guarantee) If the level set $\mathcal{L}(x^{(0)}):=\{x|f(x)\leq f(x^{(0)})\}$ is bounded, then the following limit exists.*

$$\lim_{i\to\infty}f(x^{(i)})=f^*,$$

The set X^* of the limit points of this sequence is non-empty and a connected set. For any $x\in X^*$ we have

$$f(x)=f^*,\nabla f(x)=\vec{0},\nabla^2 f(x)\succeq 0.$$

2.4 Comparison to the Paper of Interest

The main algorithm of the paper of interest[1] alternates between a second order step using cubic regularization(Nesterov and Polyak, 2006[8]) and a third order step. The third order step first identifies a "competitive subspace" where the third order derivative has a much larger norm than the second order. It then tries to find a good direction in this subspace to make improvement. We will see detailed description of the algorithm in the following section(Section 3).

However, for the reviewed paper, when the Hessian is singular and positive semi-definite, these second-order methods are insufficient to guarantee escaping from saddle points. The paper by Anandkumar and Ge introduced a novel approach using higher order derivatives to ensure escape from higher order saddle points. This is develped on the basis of a higher order optimality conditions(Bernstein,1984[3], Warga, 1986[10]). But such conditions are NP-hard to determine.(Murty and Kabadi, 1987[7])

The main improvement of the paper of interest is that it took a step forward to gave an algorithm that guarantees convergence to third order local minima and proposed a statement of NP-hard fourth order optimality. This is a very concise conclusion that summarized the previous research and made breakthrough for optimization in higher dimensions without strict saddle property.

3 Methodology

3.1 Reflection on the Method

The method basically adapted and extended results from the Nesterov's Cubic Regularization[8] to alter between the second and third order. We will see the detailed adaptation and modification process in from section 3.2 to section 3.5. In this way, the paper specifically built the algorithm for finding third order local minima. The algorithm is shown in section 4.

3.2 Local Minimum and Tensor

Functions $f:R^n\to R$ with first three order derivatives exist. The derivatives are $\nabla f(x)\in R^n$, $\nabla^2 f(x)\in R^{n\times n}$ and $\nabla^3 f(x)\in R^{n^3}$, where

$$[\nabla f(x)]_i=\frac{\partial}{\partial x_i}f(x),[\nabla^2 f(x)]_{i,j}=\frac{\partial^2}{\partial x_i\partial x_j}f(x),[\nabla^3 f(x)]_{i,j,k}=\frac{\partial^3}{\partial x_i\partial x_j\partial x_k}f(x).$$

For such smooth function $f(x)$, we say x is a *critical point* if $\nabla f(x)=\vec{0}$. Traditionally, critical points are classified into four cases according to the Hessian matrix:

1. (Local Minimum) All eigenvalues of $\nabla^2 f(x)$ are positive.
2. (Local Maximum) All eigenvalues of $\nabla^2 f(x)$ are negative.
3. (Strict saddle) $\nabla^2 f(x)$ has at least one positive and one negative eigenvalues.
4. (Degenerate) $\nabla^2 f(x)$ has either nonnegative or nonpositive eigenvalues, with some eigenvalues equal to 0.

Definition 2 (p-th order local minimum)**.** *A critical point x is a p-th order local minimum, if there exists constants $C, \epsilon > 0$ such that for every y with $\|y - x\| \leq \epsilon$,*

$$f(y) \geq f(x) - C\|x - y\|^{p+1}.$$

For a vector $v \in R^n$, we use $\|v\|$ to denote its ℓ_2 norm. For a matrix $M \in R^{n \times n}$, we use $\|M\|$ to denote its spectral (operator) norm.

Consider symmetric matrices with eigen-decomposition: $M = \sum_{i=1}^n \lambda_i v_i v_i^\top$. $\lambda_1 \geq \ldots \geq \lambda_n$. Spectral norm $\|M\| = \max\{|\lambda_1(M)|, |\lambda_n(M)|\}$. Frobenius norm $\|M\|_F = \sqrt{\sum_{i,j\in[n]} M_{i,j}^2}$.

The third order derivative is represented by a $n \times n \times n$ tensor T.

Definition 3 (Multilinear notations)**.** *Let $T \in R^{n \times n \times n}$ be a third order tensor. Let $U \in R^{n \times n_1}$, $V \in R^{n \times n_2}$ and $W \in R^{n \times n_3}$ be three matrices, then the multilinear form $T(U, V, W)$ is a tensor in $R^{n_1 \otimes n_2 \otimes n_3}$ that is equal to*

$$[T(U, V, W)]_{p,q,r} = \sum_{i,j,k \in [n]} T_{i,j,k} U_{i,p} V_{j,q} W_{k,r}.$$

For vectors $u, v, w \in R^n$, $T(u, v, w)$ is a number that relates linearly in u, v and w (similar to $u^\top M v$ for a matrix); $T(u, v, I)$ is a vector in R^n (similar to Mu for a matrix); $T(u, I, I)$ is a matrix in $R^{n \times n}$.

Frobenius norm $\|T\|_F$=$\sqrt{\sum_{i,j,k\in[n]} T_{i,j,k}^2}$. Spectral norm $\|T\|$= $\max_{\|u\|=1, \|v\|=1, \|w\|=1} T(u, v, w)$.

Symmetric tensor if $T_{i,j,k} = T_{\pi(i,j,k)}$ for any permutation of the indices. For symmetric tensors, $\|T\| = \max_{\|u\|=1} T(u, u, u)$.

Let P be the projection matrix to the subspace P, we use the notation $\text{Proj}_P T$ which denotes $T(P, P, P)$. $[T(P, P, P)]_{u,v,w} = T(Pu, Pv, Pw)$

3.3 Necessary Condition for Third Order Optima

Consider functions satisfying the following smoothness conditions.

Assumption 5 (Lipschitz third Order)**.** *We assume the first three derivatives of $f(x)$ exist, and for any $x, y \in \mathcal{R}^n$,*

$$\|\nabla^3 f(x) - \nabla^3 f(y)\|_F \leq L\|x - y\|.$$

Definition 4 (Third-order necessary condition)**.** *A point x satisfy third-order necessary condition, if*

1. $\nabla f(x) = 0$.
2. $\nabla^2 f(x) \succeq 0$.
3. *For any u that satisfy $u^\top(\nabla^2 f(x))u = 0$, $[\nabla^3 f(x)](u, u, u) = 0$.*

This can be verified in polynomial time. We can check whether $\nabla f(x) = 0$ and $\nabla^2 f(x) \succeq 0$, compute the projection of $\nabla^3 f(x)$ in the subspace $\mathcal{P}$, and claim the third condition is violated if and only if the projection is nonzero. If projection Z is nonzero, let u be a uniform Gaussian vector that has unit variance in all directions, there must exists an $u \in \mathcal{P}$ such that $[\nabla^3 f(x)](u, u, u) \neq 0$.

Theorem 6. *Given a function f that satisfies Assumption 5, a point x is third order optimal if and only if it satisfies Condition 4[1].*

Lemma 2. *For any x, y, we have*

$$|f(y) - f(x) - \langle \nabla f(x), y - x, + \rangle \frac{1}{2}(y - x)^\top \nabla^2 f(x)(y - x) - \frac{1}{6}\nabla^3 f(x)(y - x, y - x, y - x)| \leq \frac{L}{24}\|y - x\|^4. [I]$$

3.4 Finding Third Order Optima

Definition 5 (eigensubspace)**.** *For any symmetric matrix M, let its eigendecomposition be $M = \sum_{i=1}^n \lambda_i v_i v_i^\top$ (where λ_i's are eigenvalues and $\|v_i\| = 1$), we use $\mathcal{S}_\tau(M)$ to denote the span of eigenvectors with eigenvalue at most τ. That is $\mathcal{S}_\tau(M) = span\{v_i | \lambda_i \leq \tau\}$.*

Definition 6 (competitive subspace)**.** *For any $Q > 0$, and any point z, let the competitive subspace $\mathcal{S}(z)$ be the largest eigensubspace $\mathcal{S}_\tau(\nabla^2 f(z))$, such that if we let $C_Q(z)$ be the norm of the third order derivatives in this subspace $C_Q(z) = \|Proj_{\mathcal{S}(z)}\nabla^3 f(z)\|_F$, then $\tau \le C_Q^2/12LQ^2$. If no such subspace exists then let $\mathcal{S}(z)$ be empty and $C_Q(z) = 0$.*

Assumptions 4 and 5 are satiesfied. Checking the first two conditions is easy, and the third can be verified in a competitive subspace. If both $\mu(z)$ and $C_Q(z)$ are 0 then the point z satisfies third order necessary conditions. Competitive subspace is a subspace where the eigenvalues of the Hessian are small, but the Frobenius norm of the third order derivative is large.

Theorem 7. *There is a universal constant B such that the expected number of iterations of Algorithm 3 is at most 2, and the output of Approx is a unit vector u that satisfies $T(u,u,u) \ge \|Proj_{\mathcal{S}}T\|_F/Q$ for $Q = Bn^{1.5}$.*

We can find a new point by choosing parameters in the algorithm, where the sum of second, third and fourth order term can be bounded.

Lemma 3. *If $C_Q(z) \ge Q(24\epsilon_1 L)^{1/3}$, u is a unit vector in $\mathcal{S}(z)$ and $[\nabla^3 f(z)](u,u,u) \ge \|Proj_{\mathcal{S}(z)}\nabla^3 f(z)\|_F/Q$. Let $x' = z - C_Q(z)/LQ \cdot u$. then we have*

$$f(x') \le f(z) - \frac{C_Q(z)^4}{24L^3Q^4}.$$

Theorem 8 (Convergence)**.** *Suppose the algorithm starts at $f(x_0)$, and f has global min at $f(x^*)$. Then in one of the t iterations we have*

1. $\mu(z) \le \left(\frac{12(f(x_0)-f(x^*))}{Rt}\right)^{1/3}$.
2. $C_Q(z) \le \max\left\{Q(24\|\nabla f(z)\|L)^{1/3}, Q\left(\frac{24L^3(f(x_0)-f(x^*))}{t}\right)^{1/4}\right\}$.

Notice $\mu(z) = \max\left\{\sqrt{\frac{1}{R}\|\nabla f(z)\|}, -\frac{2}{3R}\lambda_n\nabla^2 f(z)\right\}$ measures the first and second order progress. $C_Q(z)$ 6 is the third order progress. Both values goes to 0 as t increases.

Theorem 9. *When t goes to infinity, the values $f(x^{(t)})$ converge. If the level set $\mathcal{L}(f(x_0)) = \{x|f(x) \le f(x_0)\}$ is compact, then the sequence of points $x^{(t)}, z^{(t)}$ has nonempty limit points, and every limit point x satisfies the third order necessary conditions.*

Sometimes a function can be well-behaved and even if it has degenerate saddle points, it may still satisfy the following condition.

Definition 7 (strict third order saddle[6].)**.** *We say a function is strict third order saddle, if there exists constants $\alpha, c_1, c_2, c_3, c_4 > 0$ such that for any point x one of the following is true:*

1. $\|\nabla f(x)\| \ge c_1$.
2. $\lambda_n(f(x)) \le -c_2$.
3. $C_Q(f(x)) \ge c_3$.
4. *There is a local minimum x^* such that $\|x - x^*\| \le c_4$ and the function is α-strongly convex restricted to the region $\{x|\|x - x^*\| \le 2c_4\}$.*

Corollary 1. *When $t \ge poly(n, L, R, Q, f(x_0) - f(x^*)) \max\{(1/c_1)^{1.5}, (1/c_2)^3, (1/c_3)^{4.5}\}$, there must be a point $z^{(i)}$ with $i \le t$ that is in case 4 in Definition 7.*

3.5 Fourth Order Complexity

First, the paper defined well-behaved functions, for which all local minimizers of a well-behaved function lies within the unit ℓ_2 ball and and $f(x)$ is smooth with bounded derivatives within the ball.

Definition 8 (Well-behaved function)**.** *We say a function f is well-behaved if it is infinite-order differentiable, and satisfies:*

1. *$f(x)$ has a global minimizer at some point $\|x\| \leq 1$.*
2. *$f(x)$ has bounded first 5 derivatives for $\|x\| \leq 1$.*
3. *For any direction $\|x\| = 1$, $f(tx)$ is increasing for $t \geq 1$.*

Theorem 10. *It is NP-hard to find a fourth order local minimum of a function $f(x)$, even if f is guaranteed to be well-behaved.*

Theorem 11. *It is NP-hard to tell whether a degree 4 homogeneous polynomial $f(x)$ is nonnegative.*

The main trick of proof here is to reduce the verification of Theorem 11 nonnegativeness to finding a fourth order local minimum in Theorem 10. The degree 4 polynomial can be a well-behaved function if added with $\|x\|^6$. If $f(x)$ is nonnegative, then $\vec{0}$ is the unique fourth order local minimizer of $g(x)$. If $f(x)$ is negative for some x, then if x is a fourth order local minimum of $g(x)$ then $f(x) < 0$.

4 Results and Analysis

4.1 Third order optima

Algorithm 2 Third Order Optimization

for $i = 0$ **to** $t - 1$ **do**
  $z^{(i)} = \text{CubicReg}(x^{(i)})$.
  Let $\epsilon_1 = \|\nabla f(z^{(i)})\|$,
  Let $\mathcal{S}(z), C_Q(z)$ be the competitive subpace of $f(z)$ (Definition 6).
  if $C_Q(z) \geq Q(24\epsilon_1 L)^{1/3}$ **then**
    $u = \text{Approx}(\nabla^3 f(z^{(i)}), \mathcal{S})$.
    $x^{(i+1)} = z^{(i)} - \frac{C_Q(z)}{LQ} u$.
  else
    $x^{(i+1)} = z^{(i)}$.
  end if
end for

Algorithm 3 Approximate Tensor Norms

Require: Tensor T, subspace $\mathcal{S}$.
Ensure: unit vector $u \in \mathcal{S}$ such that $T(u, u, u) \geq \|\text{Proj}_{\mathcal{S}} T\|_F / Q$.
repeat
  Let $\hat{u}$ be a random standard Gaussian in subspace $\mathcal{S}$.
  Let $u = \hat{u}$
until $|T(u, u, u)| \geq \|\text{Proj}_{\mathcal{S}} T\|_F / Bn^{1.5}$ for a fixed constant B
return u if $T(u, u, u) > 0$ and $-u$ otherwise.

Algorithm 4 Algorithm for computing the competitive subspace

Require: Function f, point z, Hessian $M = \nabla^2 f(z)$, third order derivative $T = \nabla^3 f(z)$, approximation ratio Q, Lipschitz Bound L,
Ensure: Competitive subpace $\mathcal{S}(z)$ and $C_Q(z)$.
Compute the eigendecomposition $M = \sum_{i=1}^{n} \lambda_i v_i v_i^\top$.
for $i = 1$ **to** n **do**
  Let $\mathcal{S} = \text{span}\{v_i, v_{i+1}, ..., v_n\}$.
  Let $C_Q = \|\text{Proj}_{\mathcal{S}} T\|_F$.
  if $\frac{C_Q^2}{12LQ^2} \geq \lambda_i$ **then**
    return $\mathcal{S}, C_Q$.
  end if
end for
return $\mathcal{S} = \emptyset, C_Q = 0$.

4.2 Fourth order optima

Theorem 10 It is NP-hard to find a fourth order local minimum of a function $f(x)$, even if f is guaranteed to be well-behaved. 8

4.3 Algorithm Analysis and Interpretation

The third order optimization algorithm method extended results from the Nesterov's Cubic Regularization[8] and provided a way to guarantee convergence to third order optima. The third order optima algorithm is a combination of the cubic regularization algorithm and a third order step that tries to use the third order derivative in order to improve the function value in the competitive subspace.

For Algorithm 2. Not all third order local minimum can be the limit point. If $f(x)$ has very large third order derivatives but relatively smaller Hessian, even though the Hessian might be positive definite, Algorithm 2 may still find a non-empty competitive subspace.Therefore, a local minimum might also make the algorithm escape from it.

For Algorithm 4, the competitive subspace can be computed in polynomial time. The algorithm is to compute the eigendecomposition of the Hessian $\nabla^2 f(z) = \sum_{i=1}^{n} \lambda_i v_i v_i^\top$, and Enumerate over all of n different subspaces (span$\{v_n\}$, span$\{v_{n-1}, v_n\}$, $\ldots$, span$\{v_1, v_2, \ldots v_n\}$). Check for which subspaces the norm of the third order derivative is large.

The NP hardness for fourth order optimization is proved in a beautiful way by reducing another NP-hard problem into it. It can be proven in different ways, as was mentioned in the paper of interest, which could rely on SUBSET SUM problem or copositive matrices.

5 Conclusion and Future work

The main contribution of the paper by Anima Anandkumar and Rong Ge[1] is that they introduced an algorithm based on many previous non-convex optimization paper and proposed one algorithm for and one upper-bound theorem for efficiently escaping higher order saddle points with third order and fourth order method. Before them, traditional methods basically focused on first or second order conditions for the local optimum and strict saddle point. The paper proposed an algorithm for third order optimization method based mainly on the Nestorov's Cubic Regularization.[8]

While the algorithm proposed breakthrough theory, the optimization of the algorithm is still an open problem. The method relies on higher order derivatives, which introduces computational complexities and may limit the performance. Furthermore, the method's performance in different and typical high-dimensional space structure also remains not explored.

Future research could focus on optimizing the computational efficiency of the algorithm. For Algorithm 3, this paper did not try to optimize over dependencies over the dimension n. There are other methods that might be able to potentially give better approximation of Q, which can also be a unique way to improve the performance of the third order optimization. This should be a valuable area of exploration. Additionally, exploring the algorithm's applicability to problems with different landscapes could also improve its utility. Extending the theoretical framework for fourth order hardness in different problem settings could also be a research direction.

A Github copy of this review: Project Project(private copy)

References

[1] Anima Anandkumar and Rong Ge. Efficient approaches for escaping higher order saddle points in non-convex optimization. *CoRR*, abs/1602.05908, 2016.

[2] Michel Baes. Estimate sequence methods: extensions and approximations. 09 2009.

[3] Dennis S. Bernstein. A systematic approach to higher-order necessary conditions in optimization theory. *Siam Journal on Control and Optimization*, 22:211–238, 1984.

[4] Yann N. Dauphin, Razvan Pascanu, Çaglar Gülçehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *CoRR*, abs/1406.2572, 2014.

[5] A. Frieze, M. Jerrum, and R. Kannan. Learning linear transformations. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 359–368, 1996.

[6] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points — online stochastic gradient for tensor decomposition. In Peter Grünwald, Elad Hazan, and Satyen Kale, editors, *Proceedings of The 28th Conference on Learning Theory*, volume 40 of *Proceedings of Machine Learning Research*, pages 797–842, Paris, France, 03–06 Jul 2015. PMLR.

[7] Katta G. Murty and Santosh N. Kabadi. Some np-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39:117–129, 1987.

[8] Yurii Nesterov and Boris Polyak. Cubic regularization of newton method and its global performance. *Math. Program.*, 108:177–205, 08 2006.

[9] Ju Sun, Qing Qu, and John Wright. When are nonconvex problems not scary? *ArXiv*, abs/1510.06096, 2015.

[10] J. Warga. Higher order conditions with and without lagrange multipliers. *SIAM Journal on Control and Optimization*, 24(4):715–730, 1986.

[11] Haikun Wei, Jun Zhang, Florent Cousseau, Tomoko Ozeki, and Shun-ichi Amari. Dynamics of learning near singularities in layered networks. *Neural Computation*, 20(3):813–843, 2008.