Paper Review: Use Third Order Optimal Condition to Escape Higher Order Saddle Points in Non-convex Optimization[1]

Fengyin Cen

April 30, 2024

Abstract

Local search heuristics for non-convex optimizations are popular in applied machine learning. However, the algorithms may converge to saddle points which have complicated structures and may not be local minimum. So in general it is hard to guarantee that such algorithms even converge to a local minimum. This work shows a method which uses higher order derivatives to escape these saddle points.

***Keywords**: saddle points, local minimum, third order derivative*

1 Introduction

When applying machine learning, it's popular to use local search heuristics for non-convex optimizations. However, less attention is paid to the issue of reaching a locally optimal solution. In fact, even this is computationally hard in the worst case[2]. With the dimension increases, the number of saddle points grows exponentially for many problems of interest, e.g.[3], [4]. Ordinary gradient descent can be stuck in a saddle point for an arbitrarily long time before making progress. To solve this problem, a few works have been addressed. However, these works require the Hessian matrix at the saddle point to have a strictly negative eigenvalue, which is also known as the strict saddle condition. But what if we come across regions where neither the gradient or Hessian information can lead to a direction that improves the function value? Neither the first or second order derivative would provide enough information. So, we need higher order derivatives to classify the point as either a local optimum or a saddle point.

1.1 Critical Points

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 non-negative or non-positive eigenvalues, with some eigenvalues equal to 0.

For the first three cases second order algorithms can either find a direction to reduce the function value (in case of local maximum or strict saddle), or correct asserting that the current point is a local minimum. However, second order algorithms cannot handle degenerate saddle points. Degeneracy of Hessian indicates the presence of a gutter structure, where a set of connected points all have the same value, and all are local minima, maxima or saddle points[5].

The paper considers higher order local minimum:

Definition 1 *(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 $||y - x|| \leq \epsilon$,*

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

1.2 Tensor Notations

The third order derivative is represented by a $n \times n \times n$ tensor T. The following multilinear notation is used to simplify the notations of tensors:

Definition 2 *(Multilinear notations):Let $T \in \mathbb{R}^{n \times n \times n}$ be a third order tensor. Let $U \in \mathbb{R}^{n \times n_1}$, $V \in \mathbb{R}^{n \times n_2}$ and $W \in \mathbb{R}^{n \times n_3}$ be three matrices, then the multilinear form $T(U, V, W)$ is a tensor in $\mathbb{R}^{n_1 \otimes n_2 \otimes n_3}$ that is equal to*

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

In particular, for vectors $u, v, w \in \mathbb{R}^n$, $T(u, v, w)$ is a number that relates linearly in u, v and w; $T(u, v, I)$ is a vector in $\mathbb{R}^n$; $T(u, I, I)$ is a matrix in $\mathbb{R}^{n \times n}$. Let P be the projection matrix to the subspace $\mathcal{P}$. To project a tensor T to a subspace $\mathcal{P}$, we use the notation $Proj_{\mathcal{P}} T$ which denotes $T(P, P, P)$. Intuitively, $[T(P, P, P)]_{u,v,w} = T(Pu, Pv, Pw)$, that is, the projected tensor applied to vector u, v, w is equivalent to the original tensor applied to the projection of u, v, w.

2 Related Work

One way to overcoming saddle points is to incorporate second order information. Directions along negative values of the Hessian matrix help in escaping the saddle point. A simple solution is then to use these directions, whenever gradient descent improvements are small[6].

Another more elegant way is trust region method[5] which involves optimizing the second order Taylor's approximation of the objective function in a local neighborhood of the current point. In [7], it's proposed that a cubic regularization term is added to this Taylor's approximation. As a result, in each step, this cubic regularized objective can be solved optimally due to hidden convexity and overall, the algorithm converges to a local optimum in bounded time. [8]generalizes this idea to use higher order Taylor expansion, however the optimization problem is intractable even for third order Taylor expansion with quartic regularizer.

All the above works deal with local optimality based on second order conditions, which is easy to compute. But when the Hessian matrix is singular and p.s.d., higher order derivatives are required to determine whether it is a local optimum or a saddle point. Higher order optimality conditions, both necessary and sufficient, have been characterized before, e.g. [9], [10]. These conditions are not efficiently computable, and it is NP-hard to determine local optimality, given such information about higher order derivatives[11].

3 Nestorov's Cubic Regularization[7]

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

Assumption 1 *(Lipschitz-Hessian):*

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

At a point x, the algorithm tries to find a nearby point z that optimizes the degree two Taylor's expansion: $f(x) + \langle \nabla f(x), z - x \rangle + \frac{1}{2}(z-x)^T (\nabla^2 f(x))(z-x)$, with cubic distance $\frac{R}{6}||z-x||^3$ as a regularizer. Algorithm 1 denotes one iteration of the algorithm. The final algorithm generates a sequence of points $x^{(0)}, x^{(1)}, x^{(2)}$...where $x^{(i+1)}$ =CubicReg($x^{(i)}$). The optimization problem could be solved in polynomial time.

Algorithm 1 CubicReg

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

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

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

When $\mu(z)$ is small, we know $\nabla f(z) \approx 0$ and $\nabla^2 f(z) \succeq 0$, which means the point z approximately satisfies the second order optimality condition.

Theorem 1 *Suppose $z = CubicRegularize(x)$, then $||z - x|| \geq \mu(z)$ and $f(z) \leq f(x) - R||z-x||^3/12$*

Use Theorem 3 could get strong convergence results for the sequence $x^{(0)}, x^{(1)}, x^{(2)}$...

Theorem 2 *If $f(x)$ is bounded below by $f(x^*)$, then $\lim_{i\to\infty} \mu\left(x^{(i)}\right) = 0$, and for any $t \geq 1$ we have*

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

This theorem shows that within first t iterations, a point that "looks similar" to a second order local minimum could be found in the sense that gradient is small and Hessian does not have a negative eigenvalue with large absolute value. It is also possible to prove stronger guarantees for the limit points of the sequence:

Theorem 3 *If the levelset $\mathcal{L}\left(x^{(0)}\right) := \left\{x | f(x) \leqslant f\left(x^{(0)}\right)\right\}$ is bounded, then the following limit exists*

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

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

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

Therefore the algorithm always converges to a set of points that are all second order local minima.

4 Third Order Optimal Condition

4.1 Third Order Necessary Condition

Assumption 2 *(Lipschitz third Order). We assume the first three derivatives of $f(x)$ exist, and for any $x, y \in \mathbb{R}^n$,*

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

Under this assumption, we could state conditions for a point to be a third order local minimum.

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 $u^T(\nabla^2 f(x))u = 0$,$[\nabla^3 f(x)](u, u, u) = 0$.*

Claim 1 *Conditions in Definition 4 can be verified in polynomial time given the gradients $\nabla f(x)$, $\nabla^2 f(x)$ and $\nabla^3 f(x)$.*

Proof. It is easy to check the first 2 conditions. We could use SVD to construct a subspace $\mathcal{P}$ such that $u^T(\nabla^2 f(x))u = 0$ if and only if $u \in \mathcal{P}$. Then we can compute the projection of $\nabla^3 f(x)$ in the subspace $\mathcal{P}$, and we claim the third condition is violated if and only if the projection is nonzero.

If the projection is zero, it's clearly that $[\nabla^3 f(x)](u, u, u) = 0$ for any $u \in \mathcal{P}$. If the projection Z is nonzero, let u be a uniform Gaussian vector that has unit variance in all directions of u, then we know $\mathbb{E}[[[\nabla^3 f(x)](u, u, u)]^2] \geq ||Z||_F^2 > 0$, so there must exists an $u \in \mathcal{P}$ such that $[\nabla^3 f(x)](u, u, u) \neq 0$.

Lemma 1 *For any x, y, we have*

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

We could prove the lemma by integrating over the third order derivatives three times and bounding the differences.

Theorem 4 *Given a function f that satisfies Assumption2, a point x is third order optimal if and only if it satisfies Condition 4.*

Proof. For the "only if" direction, if any condition in Definition 4 is violated, we can use that particular derivative to find a direction that improves the function value. For the "if" direction, let α be the smallest nonzero eigenvalue of $\nabla^2 f(x)$, U be nullspace of $\nabla^2 f(x)$ and V be the orthogonal subspace. Break $\nabla^3 f(x)$ into two tensors G_1 and G_2, where G_1 is the projection to $V \otimes V \otimes V$, $V \otimes V \otimes U$ (and its symmetries), and G_2 is the projection to $U \otimes U \otimes U$. Let β be the max injective norm of G_1 and G_2. For any $u \in U$ and $v \in V$, we have

$$f(x + u + v) - f(x) \geq -(\frac{\beta^2}{\alpha} + \frac{L}{24})||u + v||^4$$

4.2 Algorithm for Finding Third Order Optimal Points

The main intuition of the algorithm is similar to the proof of Theorem 4. Consider a potential local minimum point x. It is easy to check whether $\nabla f(x) \neq 0$ or $\lambda_{min}(\nabla^2 f(x)) < 0$. To verify Condition 3 in Definition 4, the naive guess is that take the eigensubspace of $\nabla^2 f(x)$ with eigenvalue at most 0. However, even if x is a second order local minimum that does not satisfy the third order condition, it is still possible to have a sequence of $x^{(i)}$'s that converge to x with $\nabla^2 f(x^{(i)})$ all be strictly positive definite. We need to identify a subspace that may have some positive eigenvalues. In order to make sure we can find a vector the contribution from third order term is larger than the second order term, we define competitive subspace below:

Definition 5 *(eigensubspace).For any symmetric matrix M, let its eigendecomposition be $M = \sum_{i=1}^{n} \lambda_i v_i {v_i}^T$ (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 $\tau \leq C_Q^2/12LQ^2$. If no such subspace exists then let $\mathcal{S}(z)$ be empty and $C_Q(z) = 0$.

$C_Q(z)$ can be viewed as how Condition 3 in Definition 4 is satisfied approximately. 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. The parameters in Definition 6 are set so that if there is a unit vector $u \in \mathcal{S}(z)$ such that $[\nabla^3 f(z)](u,u,u) \geq ||Proj_{\mathcal{S}(z)} \nabla^3 f(z)||_F / Q$, then we can find a new point where the sum of second, third and fourth order term can be bounded.

The competitive subspace can be computed in polynomial time. Because We can compute the eigendecomposition of the Hessian $\nabla^2 f(z) = \sum_{i=1}^{n} \lambda_i v_i {v_i}^T$. There are n different subspaces(span$\{v_n\}$,span$\{v_{n-1}, v_n\}$,...,span$\{v_1, v_2, ..., v_n\}$). We just need to check in which of them the norm of the third order derivative is the largest.

Theorem 5 *There is a universal constant B such that the expected number of iterations of Algorithm 4.2 is at most 2, and the output of Approx is a unit vector u that satisfies $T(u,u,u) \geq ||Proj_{\mathcal{S}} T||_F / Q$ for $Q = Bn^{1.5}$.*

The proof of this theorem follows directly from anti-concentration(Theorem 6).

Theorem 6 *(anti-concentration[15]). Let $x \in \mathbb{R}^n$ be a Gaussian variable $x \sim N(0, I)$, for any polynomial $p(x)$ of degree d, there exists a constant κ such that*

$$\Pr[|p(x)| \leq \epsilon \sqrt{\mathrm{Var}[p(x)]}] \leq \kappa \epsilon^{1/d}.$$

In the case $d = 3$, we can choose some universal constant ϵ such that the probability of $p(x)$ being small is bounded by $1/3$. It is easy to check that the variance is lowerbounded by the Frobenius norm squared, so

$$\Pr[|T(\hat{u}, \hat{u}, \hat{u})| \geq \epsilon ||\text{Proj}_S T||_F] \geq 2/3.$$

On the other hand with high probability we know the norm of the Gaussian $\hat{u}$ is at most $2\sqrt{n}$. Therefore with probability at least $1/2$, $|T(\hat{u}, \hat{u}, \hat{u})| \geq \epsilon ||\text{Proj}_S T||_F$ and $||\hat{u}|| \leq 2\sqrt{n}$, therefore $|T(u, u, u)| \geq \frac{\epsilon}{8n^{1.5}} ||\text{Proj}_S T||_F$. Choosing $B = 8/\epsilon$ implies the theorem.

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 subspace 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 $T(u, u, u) \geq \|Proj_{\mathcal{S}} T\|_F / Q$.
repeat
  Let $\hat{u}$ be a random standard Gaussian in subspace $\mathcal{S}$.
  Let $u = \hat{u}$
until $|T(u, u, u)| \geq \|Proj_{\mathcal{S}} T\|_F / Bn^{1.5}$ for a fixed constant B
return u if $T(u, u, u) > 0$ and $-u$ otherwise.

Lemma 2 *If $C_Q(z) \geq Q(24\epsilon_1 L)^{1/3}$, u is a unit vector in $\mathcal{S}(z)$ and $[\nabla^3 f(z)](u, u, u) \geq ||Proj_{\mathcal{S}(z)} \nabla^3 f(z)||_F / Q$. Let $x' = z - C_Q(z)/LQ \cdot u$. Then we have*

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

Proof. Let $\epsilon = C_Q(z)/LQ$, then by Lemma 1 we know

$$f(x') \leq f(z) - \frac{\epsilon^3 C}{6Q} + \epsilon_1 \epsilon + \epsilon_2 \epsilon^2/2 + L\epsilon^4/24.$$

Here $\epsilon_1 = ||\nabla f(z)||$, and $\epsilon_2 \leq \frac{C_Q(z)^2}{12LQ^2}$. So the terms $\epsilon_1\epsilon, \epsilon_2\epsilon^2/2, L\epsilon^4/24$ are all bounded by $\frac{\epsilon^3 C_Q(z)}{24Q}$, therefore

$$f(x') \leq f(z) - \frac{\epsilon^3 C_Q(z)}{24Q} = f(z) - \frac{C_Q(z)^4}{24 L^3 Q^4}$$

Theorem 7 *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. $\mu(z) \leq \left(\frac{12(f(x_0)-f(x^*)}{Rt}\right)^{1/3}$.
2. $C_Q(z) \leq \max\left\{Q(24\|\nabla f(z)\|L)^{1/3}, Q\left(\frac{24L^3(f(x_0)-f(x^*))}{t}\right)^{1/4}\right\}$.

Just like Definition 3 measures how much first and second order progress the algorithm can make, $C_Q(z)$ measures how much third order progress the algorithm can make. Both values goes to 0 as t increases.

Proof. By the guarantees of Theorem 1 and Lemma 2, we know the sequence of points $x^{(0)}, z^{(0)}, ..., x^{(i)}, z^{(i)}, ...$ has non-increasing function values. Also,

$$\sum_{i=1}^{t} f(x^{(i)}) - f(x^{(i-1)}) \leq f(x_0) - f(x^*).$$

So there must be an iteration where $f\left(x^{(i)}\right) - f\left(x^{(i-1)}\right) \leq \frac{f(x_0)-f(x^*)}{t}$. If the condition 1 of Theorem 7 is violated, Theorem 1 implies $f\left(x^{(i-1)}\right) - f\left(z^{(i-1)}\right) > \frac{f(x_0)-f(x^*)}{t}$, which is impossible. If the condition 2 is violated, then the third order step makes progress, and we know $f\left(x^{(i-1)}\right) - f\left(z^{(i-1)}\right) > \frac{f(x_0)-f(x^*)}{t}$, which is either impossible.

Theorem 8 *When t goes to infinity, the values $f(x^{(t)})$ converge. If the level set $\mathcal{L}(f(x_0)) = \{x \,|f\,(x) \leqslant f\,(x_0)\}$ 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.*

Proof. Since the fucntion value is non-increasing, and it has a lowerbound $f(x^*)$, so the value must converge. We only need to prove that every limit point x must satisfy the third order necessary conditions.

Notice that $f(x^{(0)}) - \lim_{t\to\infty} f(x^{(t)}) \geq \sum_{i=0}^{\infty} \frac{R\mu(z^{(i)})^3}{12} + \frac{C_Q(z^{(i)})^4}{24L^3Q^4}$, so $\lim_{i\to\infty}\mu(z^{(i)}) = 0$ and $\lim_{i\to\infty} C_Q(z^{(i)}) = 0$. Also we know further $\lim_{i\to\infty}||z^{(i)} - x^{(i)}|| = 0$. Therefore a limit point x is also a limit point of sequence z, and $\lim_{i\to\infty}||\nabla f(z)|| = 0$. Also we know $H = \nabla^2 f(x)$ is PSD, otherwise the points near x will have nonzero $\mu(z^{(i)})$ and x cannot be a limit point.

To prove the third order condition, we can assume it's contradiction is true. So, the Hessian has a subspace $\mathcal{P}$ with 0 eigenvalues, and the third order derivative has norm at least ϵ in this subspace. By matrix perturbation theory, $\mathcal{P}$ is very close to $\mathcal{S}_\epsilon(z)$ for $\epsilon \to 0$ when z is very close to x. And the third order tensor also converges to $\nabla^3 f(x)$, so $\mathcal{S}_\epsilon(z)$ will eventually be a competitive subspace and $C_Q(z)$ is at least $\epsilon/2$ for all z. However this is impossible as $\lim_{i\to\infty} C_Q(z^{(i)}) = 0$.

In the most general case it is hard to get a convergence rate for the algorithm because the function may have higher order local minima. However, if the function has nice properties then it is possible to prove polynomial rates of convergence.

Definition 7 *(strict third order saddle). 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. $||\nabla f(x)|| \geq c_1$.
2. $\lambda_n(f(x)) \leq -c_2$.
3. $C_Q(f(x)) \geq c_3$.
4. *There is a local minimum x^* such that $||x - x^*|| \leq c_4$ and the function is $\alpha-$strongly convex restricted to the region $\{x \,|||x - x^*|| \leqslant 2c_4\}$.*

This is a generalization of the strict saddle functions defined in [12]. Even if a function has degenerate saddle points, it may still satisfy this condition.

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

Proof. We use $\tilde{O}$ to only focus on the polynomial dependency on t and ignore polynomial dependency on all other parameters.By Theorem 7, we know there must be a $z^{(i)}$ which satisfies $\mu(z^{(i)} \leq \tilde{O}((1/t)^{1/3}))$ and $C_Q(z) \leq \tilde{O}(\max\{(1/t)^{1/4}, ||\nabla f(z)||^{1/3}\})$. By the Definition of μ(Definition 3), we know $\|\nabla f(z)\| \leq \tilde{O}(\mu(z))^2 = \tilde{O}(t^{-2/3})$, $\lambda_n(\nabla^2 f(z)) \geq -\tilde{O}(t^{-1/3})$. Using the fact that $||\nabla f(z)|| \leq \tilde{O}(\mu(z))^2 = \tilde{O}(t^{-2/3})$, we know

$$C_Q(z) \leq \tilde{O}(\max\{(1/t)^{1/4}, \|\nabla f(z)\|^{1/3}\}) = \tilde{O}(t^{-2/9}).$$

Therefore, when $t \geq poly(n, L, R, Q, f(x_0) - f(x^*))\max\{(1/c_1)^{1.5}, (1/c_2)^3, (1/c_3)^{4.5}\}$, the point z will satisfy none of the first three conditions in Definition 7, which means z must be near a local minimum.

5 Hardness for Finding a fourth order Local Minimum

In the paper, the author also proves that it is hard to find a fourth order local minimum even if the function very well-behaved.

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 $t \geq 1$*

Obviously, if a function is well-behaved, all local minimizers lies within the unit ℓ_2 ball, and $f(x)$ is smooth with bounded derivatives within the unit ℓ_2 ball. These functions also satisfy Assumptions 1 and 2. All the algorithms mentioned in previous sections can work in this case and find a local minimum up to order 3. However, this is not possible for fourth order.

Theorem 9 *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.*

The main idea of the proof comes from the fact that we cannot even verify the non-negativeness of a degree 4 polynomial (hence there are cases where we cannot verify whether a point is a fourth order local minimum or not).

Theorem 10 *It is NP-hard to tell whether a degree 4 homogeneous polynomial $f(x)$ is non-negative*

The NP hardness for non-negativeness of degree 4 polynomial has been proved has been proved in several ways. For example, the reduction in [13] relies on the hardness of copositive matrices, which in turn depends on the hardness of INDEPENDENT SET[14]. This reduction gives a polynomial whose coefficients can be bounded by poly(n), and a polynomial gap that rules out FPTAS.

To prove Theorem 9 we only need to reduce the non-negativeness problem in Theorem 10 to the problem of finding a fourth order local minimum. We can convert a degree 4 polynomial to a well behaved function by adding a degree 6 regularizer $||x||^6$. When the degree 4 polynomial is non-negative the $\vec{0}$ point is the only fourth order local minimum; when the degree 4 polynomial has negative directions then every fourth order local minimum must have negative function value.

6 Conclusion

Complicated structures of saddle points are a major problem for optimization algorithms. In the paper, the author proposes an idea of using higher order derivatives in order to avoid degenerate saddle points and gives the first algorithm that is guaranteed to find a 3rd order local minimum, which can solve some problems caused by degenerate saddle points. However, it's proved the same ideas cannot be generalized to higher orders.

As for future work, there are many open problems to investigate. For example, Are there interesting class of functions that satisfies the strict 3rd order saddle property (Definition 7)? Can we design a 3rd order optimization algorithm for constrained optimization? Hopefully, there problems could be solved and efficient optimization algorithms whose performance do not suffer from degenerate saddle points could be found.

References

[1] "Efficient Approaches for Escaping Higher Order Saddle Points in Non-convex Optimization," Anima Anandkumar, Rong Ge. (2016)

[2] Jiawang Nie. The hierarchy of local minimums in polynomial optimization. Mathematical Programming, 151(2):555-583, 2015.

[3] Dustin Cartwright and Bernd Sturmfels. The number of eigenvalues of a tensor. Linear algebra and its applications, 438(2):942-952, 2013.

[4] Antonio Auffinger, Gerard Ben Arous, et al. Complexity of random smooth functions on the high-dimensional sphere. The Annals of Probability, 41(6):4214-4247, 2013.

[5] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Advances in neural information processing systems, pages 2933-2941, 2014.

[6] Santosh S Vempala and Ying Xiao. Structure from local optima: Learning subspace juntas via higher order pca. arXiv preprint arXiv:1108.3329, 2011.

[7] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. Mathematical Programming, 108(1):177-205, 2006.

[8] Michel Baes. Estimate sequence methods: extensions and approximations. Institute for Operations Research, ETH, Z¨urich, Switzerland, 2009.

[9] Dennis S Bernstein. A systematic approach to higher-order necessary conditions in optimization theory. SIAM journal on control and optimization, 22(2):211-238, 1984.

[10] Jack Warga. Higher order conditions with and without lagrange multipliers. SIAM journal on control and optimization, 24(4):715-730, 1986.

[11] Katta G Murty and Santosh N Kabadi. Some np-complete problems in quadratic and nonlinear programming. Mathematical programming, 39(2):117-129, 1987.

[12] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points|online stochastic gradient for tensor decomposition. In Proceedings of The 28th Conference on Learning Theory, pages 797-842, 2015

[13] Christopher J Hillar and Lek-Heng Lim. Most tensor problems are np-hard. Journal of the ACM (JACM), 60(6):45, 2013.

[14] Peter JC Dickinson and Luuk Gijben. On the computational complexity of membership problems for the completely positive cone and its dual. Computational optimization andapplications, 57(2):403-415, 2014.

[15] Anthony Carbery and James Wright. Distributional and l^q norm inequalities for polynomials over convex bodies in r^n. Mathematical Research Letters, 8(3):233-248, 2001.