FAST AND GLOBAL OPTIMAL NONCONVEX MATRIX FACTORIZATION VIA PERTURBED ALTERNATING PROXIMAL POINT

Songtao Lu[†]*, Mingyi Hong*[†]*, and Zhengdao Wang*[‡]

[†]Department of Electrical and Computer Engineering, University of Minnesota Twin Cities, Minneapolis, MN, 55455, USA

[‡]Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, 50010, USA

ABSTRACT

In this paper, we use the perturbed gradient based alternating minimization for solving a class of low-rank matrix factorization problems. Alternating minimization is a simple but popular approach which has been applied to problems in optimization, machine learning, data mining, and signal processing, etc. By leveraging the block structure of the problem, the algorithm updates two blocks of variables in an alternating manner. For the nonconvex optimization problem, it is well-known the alternating minimization algorithm converges to the first-order stationary solution with a global sublinear rate. In this paper, a perturbed alternating proximal point (PA-PP) algorithm is proposed, which 1) minimizes the smooth nonconvex problem by updating two blocks of variables alternatively and 2) adds some random noise occasionally under some conditions to extract the negative curvature of the second-order information of the objective function. We show that the proposed PA-PP is able to converge (with high probability) to the set of second-order stationary solutions (SS2) with a global sublinear rate, and as a consequence quickly finds global optimal solutions for the problems considered.

***Index Terms*—** Matrix factorization, perturbed alternating proximal point (PA-PP), spline function, convergence rate, first-order stationary (SS1), second-order stationary (SS2)

1. INTRODUCTION

Algorithms that can escape from strict saddle points, which are stationary points whose Hessian matrices have negative eigenvalues, have wide applications. Many recent works have analyzed the saddle points in machine learning problems [1]. Such as learning in shallow networks, the stationary points are either global minimum points or strict saddle points. Previous work in [2] has shown that the saddle points in tensor decomposition are indeed strict saddle points. Also, it has been shown that any saddle points are strict in dictionary learning and phase retrieval problems in [3]. A recent work [4] proposed a unified analysis of saddle points for a board class of low-rank matrix factorization problems, which indicates that these saddle points are strict. The landscape of the saddle points of the asymmetric matrix factorization has been studied comprehensively in [5]. However, in these unconstrained problems, the objective function is quartic with respect to the optimization variables so that the function has no global Lipschitz continuous. In this paper, we consider a new loss function that has the same saddle points as the original one [6].

By leveraging the block structure of the optimization problems, such as matrix factorization [7, 8], tensor decomposition, matrix completion [9, 10], block coordinate descent (BCD)-type algorithms have shown superiority than other methods. Under relatively mild conditions, it is well-known that the BCD-type of algorithms converges to the first-order stationary (SS1) solution in a global sublinear rate [11]. However, despite its popularity and significant recent progress in understanding its behavior, it remains unclear whether BCD-type algorithms can converge to the set of second-order stationary solutions (SS2) with a provable global convergence rate, even for the simplest problem with two blocks of variables.

1.1. Related work

Many recent works have been focused on the performance analysis and/or design of algorithms with convergence guarantees to local minimum points/SS2 for nonconvex optimization problems. These include the trust region method [12], cubic regularized Newton's method [13, 14], a mixed approach of the first-order and second-order methods [15], and gradient descent with one-step escaping (GOSE) [16] by the calculation of eigenvectors, etc. However, these algorithms typically require second-order information, therefore they incur high computational complexity when the problem dimension becomes large.

There has been a line of work on stochastic gradient descent (SGD) algorithms, where properly scaled Gaussian noise is added to the iterates of the gradient at each time [also known as stochastic gradient Langevin dynamics, (SGLD) [17]]. However, these algorithms require a large number of iterations with $\mathcal{O}(1/\epsilon^4)$ steps to achieve the optimal point. There are fruitful results that show some carefully designed stochastic algorithms can escape from strict saddle points efficiently, such as NEgative-curvature-Originated-from-Noise (NEON) [18] and NEON2 [19].

On the other hand, there is also a line of work analyzing the deterministic gradient descent (GD) type method. With random initialization, it has been shown that GD only converges to SS2 for unconstrained smooth problems [20]. More recently, BCD, block mirror descent and proximal BCD have been proven to almost always converge to SS2 with random initialization [21, 22], but there is no convergence rate reported. Unfortunately, a follow-up study indicated that GD requires exponential time to escape from saddle points for certain pathological problems [23]. Adding some noise occasionally to the iterates of the algorithm is another way of finding the negative curvature. A perturbed version of GD has been proposed with convergence guarantees to SS2 [24], which shows a faster provable convergence rate than the ordinary gradient descent algorithm with random initialization.

1.2. Contribution of this Work

In this paper, we design and analyze a perturbed alternating minimization algorithm for solving a class of block structured unconstrained nonconvex problem, namely perturbed alternating proximal point (PA-PP). Through the perturbation of alternating minimization, the algorithm is guaranteed to converge to a set of SS2 of a nonconvex problem with high probability. By utilizing

the matrix perturbation theory, the convergence rate of the proposed algorithm is also established, which shows that the algorithm takes $\widetilde{\mathcal{O}}(1/\epsilon^{7/3})$ iterations to achieve an $(\epsilon, \epsilon^{1/3})$-SS2 with high probability, where $\widetilde{\mathcal{O}}(\cdot)$ hides factor polylog(d) that is polynomial of the logarithm of problem dimension d.

The main contributions of this work are listed as follows:

1. The landscape of a class of nonconvex problems is studied. A new structure of loss functions is considered so that the global Lipschitz continuity of the objective functions is satisfied.
2. To the best of our knowledge, it is the first time that the convergence analysis shows that some variants of alternating minimization (using the first-order information) can converge to SS2 for nonconvex optimization problems in a sublinear convergence rate.
3. Numerical results verify the effectiveness of the perturbed first-order algorithms of escaping strict points.

2. MOTIVATION OF THIS WORK

From a geometric view of the loss functions in machine learning problems, there are two types of undesired critical points: (1) local minima that are not global minima; (2) saddle points. If all critical points of a function $f(\mathbf{X})$ are either global minima or strict saddle points, we say that $f(\mathbf{X})$ has *benign* landscape [25], which is the main property interested in this paper.

2.1. Problem formulation

We consider a general asymmetric low-rank matrix factorization problem as follows,

$$\underset{\mathbf{U}\in\mathbb{R}^{n\times r},\mathbf{V}\in\mathbb{R}^{m\times r}}{\text{minimize}} f(\mathbf{U},\mathbf{V}) \triangleq \frac{1}{2}\|\mathbf{U}\mathbf{V}^T-\mathbf{M}^*\|_F^2, \tag{1}$$

where $\mathbf{M}^*$ denotes the data matrix. It is not hard to see that there is a scaling ambiguity between $\mathbf{U}$ and $\mathbf{V}$. Recent works [5] have shown that after adding a proper regularizer, the reformulated problem will not change the global optimal solution of the original one. In order to make the notation of the function concise, let $\mathbf{W} \triangleq [\mathbf{U};\mathbf{V}]$. Then, the reformulated problem is given by

$$\underset{\mathbf{U}\in\mathbb{R}^{n\times r},\mathbf{V}\in\mathbb{R}^{m\times r}}{\text{minimize}} g(\mathbf{W}) \triangleq f(\mathbf{U},\mathbf{V})+\rho(\mathbf{U},\mathbf{V}), \tag{2}$$

where

$$\rho(\mathbf{U},\mathbf{V}) \triangleq \frac{\mu}{4}\|\mathbf{U}^T\mathbf{U}-\mathbf{V}^T\mathbf{V}\|_F^2.$$

This regularizer is able to enforce the size difference between $\mathbf{U}$ and $\mathbf{V}$ as small as possible.

The problem has a wide application in areas of machine learning and signal processing. To be specific, we give the following two interesting examples. Note that both these objective functions have *benign* landscape.

2.2. Motivated examples

Matrix sensing: the matrix sensing problem with the low-rank matrix factorization approach can be formulated as

$$\underset{\mathbf{U}\in\mathbb{R}^{n\times r},\mathbf{V}\in\mathbb{R}^{m\times r}}{\text{minimize}} \frac{1}{2}\|\mathcal{A}(\mathbf{U}\mathbf{V}^T-\mathbf{M}^*)\|^2+\rho(\mathbf{U},\mathbf{V}), \tag{3}$$

where mapping $\mathcal{A}(\cdot)$ satisfies the restricted isometry property [26].

Two-layer linear neural network: given a set of data points $\{\boldsymbol{x}_i,\boldsymbol{y}_i\}_{i=1}^m$ of size m, we wish to fit a two-layer linear network using the quadratic loss as follows,

$$\underset{\mathbf{U}\in\mathbb{R}^{n\times r},\mathbf{V}\in\mathbb{R}^{m\times r}}{\text{minimize}} \sum_{i=1}^{k}\|\boldsymbol{y}_i-\mathbf{U}\mathbf{V}^T\boldsymbol{x}_i\|_2^2=\|\boldsymbol{Y}-\mathbf{U}\mathbf{V}^T\boldsymbol{X}\|_F^2, \tag{4}$$

where $\boldsymbol{X} \triangleq [\boldsymbol{x}_1,\ldots,\boldsymbol{x}_k] \in \mathbb{R}^{n\times k}$ and $\boldsymbol{Y} \triangleq [\boldsymbol{y}_1,\ldots,\boldsymbol{y}_k] \in \mathbb{R}^{m\times k}$.

2.3. Challenges of matrix factorization

Since the objective function is quartic with respect to $\mathbf{W}$, it lacks the global Lipschitz continuity. However, in most algorithms' convergence analysis, Lipschitz continuity is the key assumption, which quantifies how fast of the objective function can change.

3. LANDSCAPE OF THE LOSS FUNCTION

The landscape of problem (1) has been studied in [5]. From [5, Theorem 1], we know that when $\|\mathbf{W}\mathbf{W}^T\|_F > \frac{20}{9}\|\mathbf{M}^*\|_F$, we have $\|\nabla_{\mathbf{W}} g(\mathbf{W})\| \geq \|\mathbf{W}\mathbf{W}^T\|_F^{3/2}$, meaning that there are no saddle points whose the size of the gradient is zero. Also, it is shown in [5] that all saddle points of the problems are strict and within a ball with some certain radius, and every local optimal point of this problem is a global optimal. Unfortunately, the objective function has no global Lipschitz continuity. Instead, we will consider the following new formulation.

In this work, we consider a loss function $l(\mathbf{W})$ which has form $\sqrt{g(\mathbf{W})}$ when $\sqrt{g(\mathbf{W})} \geq 2\tau, \tau > 0$ and $g(\mathbf{W})$ when $\sqrt{g(\mathbf{W})} \leq \tau$ and use some spline function [27, Chapter 5.1] that smoothly connects functions $\sqrt{g(\mathbf{W})}$ and $g(\mathbf{W})$ within region $[\tau, 2\tau]$, where τ denotes the radius of the ball that contains all stationary point of the problem. When $\sqrt{g(\mathbf{W})} \leq 2\tau$, function has the Lipschitz continuity since the it is a bounded set. When $\|\mathbf{W}\|$ is large, we have the following lemma.

Lemma 1. *Function $\sqrt{g(\mathbf{W})}$ is Lipschitz gradient continues, when $\sqrt{g(\mathbf{W})} \geq 2\tau$ where $\tau = 3\|\mathbf{M}^*\|_F$.*

The proof of Lemma 1 is elementary (i.e., checking the boundedness the second derivative of the objective function) but cumbersome. Due to the page limit, details are omitted.

Also, we need to verify that $l(\mathbf{W})$ has no critical point within $[\tau, 2\tau]$. Therefore, as long as the algorithm can escape from the strict saddle points, the algorithm will converge to the global optimal solution.

3.1. Loss function

Consider loss function $l(x)$ that is defined for argument $x \geq 0$ as

$$l(x) \triangleq \begin{cases} x^2+\alpha, & \text{if} \quad 0 \leq x \leq \tau, \\ p(x), & \text{if} \quad \tau \leq x \leq 2\tau, \\ \beta x, & \text{if} \quad x \geq 2\tau, \end{cases} \tag{5}$$

where $p(\cdot)$ denotes the spline function and α, β are parameters such that the spline function can connect the two functions $g(\mathbf{W})$ and $\sqrt{g(\mathbf{W})}$ smoothly.

3.2. The choice of τ

First, we need to determine the size of τ. By the definition of $g(\mathbf{W})$ and $\mathbf{W}$, we have

$$\begin{aligned} \sqrt{g(\mathbf{W})} &= \sqrt{\frac{1}{2}\|\mathbf{U}\mathbf{V}^T-\mathbf{M}^*\|^2+\frac{1}{8}\|\mathbf{U}^T\mathbf{U}-\mathbf{V}^T\mathbf{V}\|^2} \\ &\leq \sqrt{\|\mathbf{U}\mathbf{V}^T\|^2+\|\mathbf{M}^*\|^2+\frac{1}{4}\|\mathbf{U}^T\mathbf{U}\|^2+\frac{1}{4}\|\mathbf{V}^T\mathbf{V}\|^2} \\ &\leq \sqrt{\frac{3}{2}\|\mathbf{W}\mathbf{W}^T\|^2+\|\mathbf{M}^*\|^2}. \end{aligned} \tag{6}$$

From [5, Theorem 1], we know that when $\|\mathbf{W}\mathbf{W}^T\|_F \geq \frac{20}{9}\|\mathbf{M}^*\|_F$ there is no SS1 point, which implies that all critical points of problem (2) are within

$$\begin{aligned} \sqrt{g(\mathbf{W})} &\leq \sqrt{\frac{3}{2}\|\mathbf{W}\mathbf{W}^T\|^2+\|\mathbf{M}^*\|^2} \\ &< \sqrt{7.5\|\mathbf{M}^*\|^2+\|\mathbf{M}^*\|^2} \leq 3\|\mathbf{M}^*\|_F. \end{aligned} \tag{7}$$

Therefore, we choose $\tau \triangleq 3\|\mathbf{M}^*\|_F$.

3.3. Objective function

For the convenience of the expression, let $h(\mathbf{W}) \triangleq \sqrt{g(\mathbf{W})}$. Then, we have

$$\nabla_{\mathbf{W}} h(\mathbf{W}) = \nabla_{\mathbf{W}} \sqrt{g(\mathbf{W})} = \frac{1}{2} \frac{\nabla_{\mathbf{W}} g(\mathbf{W})}{h(\mathbf{W})}. \tag{8}$$

By leveraging the form of function $l(x)$, we can construct the new objective function as follows:

$$f_h(\mathbf{W}) \triangleq \begin{cases} h^2(\mathbf{W}) + \alpha, & \text{if} \quad h(\mathbf{W}) \leq \tau, \\ p(h(\mathbf{W})), & \text{if} \quad \tau \leq h(\mathbf{W}) \leq 2\tau, \\ \beta h(\mathbf{W}), & \text{if} \quad h(\mathbf{W}) \geq 2\tau. \end{cases} \tag{9}$$

Taking the gradient and second-order derivative of the objective function, we have

$$\begin{aligned} \nabla_{\mathbf{W}} f_h(\mathbf{W}) =& l'(h(\mathbf{W})) \nabla_{\mathbf{W}} h(\mathbf{W}) \\ \nabla^2_{\mathbf{W}} f_h(\mathbf{W}) =& l''(h(\mathbf{W})) \nabla_{\mathbf{W}} h(\mathbf{W}) \nabla_{\mathbf{W}} h(\mathbf{W})^T \\ &+ l'(h(\mathbf{W})) \nabla^2_{\mathbf{W}} h(\mathbf{W}), \end{aligned} \tag{10}$$

which implies that the first- and second- gradients of function $f_h(\mathbf{W})$ and $h(\mathbf{W})$ with respect to variable $\mathbf{W}$ are only scaled by the new loss function. It is the key idea of designing such loss function so that the gradient properties of $h(\mathbf{W})$ is preserved by $f_h(\mathbf{W})$.

Finally, we only need to construct the spline function $p(\cdot)$ to satisfy the following boundary properties:

P1 : $p(\tau) = \tau^2 + \alpha$ and $p(2\tau) = 2\beta\tau$,

P2 : $p'(\tau) = 2\tau$ and $p'(2\tau) = \beta$,

P3 : $p''(\tau) = 2$ and $p''(2\tau) = 0$,

P4 : $p'([\tau, 2\tau]) > 0$.

Theorem 1. *The loss function $f_h(\mathbf{W})$ shown in (9) has the global Lipschitz continuity with constant L and all critical points of the function $f_h(\mathbf{W})$ have a one-to-one correspondence to the original loss function $h^2(\mathbf{W})$, where the spline function is*

$$p(x) = -\frac{1}{3\tau}(x-\tau)^3 + (x-\tau)^2 + 2(x-\tau) + \frac{10}{3}\tau^2, \tag{11}$$

and parameters $\tau = 3\|\mathbf{M}^*\|_F$, $\beta = 3\tau$, and $\alpha = \frac{7}{3}\tau^2$.

Proof. : Consider a third-order polynomial

$$p(x) = a(x-\tau)^3 + b(x-\tau)^2 + c(x-\tau) + d. \tag{12}$$

From P3, we can get $a = -\frac{1}{3\tau}$ and $b = 1$. Similarly, we can obtain $c = 2\tau$ and $\beta = 3\tau$ from P2. Finally, we have $d = \frac{10}{3}\tau^2$ and $\alpha = \frac{7}{3}\tau^2$. Therefore, we have

$$p(x) = -\frac{1}{3\tau}(x-\tau)^3 + (x-\tau)^2 + 2\tau(x-\tau) + \frac{10}{3}\tau^2. \tag{13}$$

Also, we know

$$p'(x) = -\frac{1}{\tau}(x-\tau)^2 + 2x, \tag{14}$$

where the two roots of $p'(x) = 0$ are $(2 \pm \sqrt{3})\tau$, implying that $p'([\tau, 2\tau]) > 0$. Hence, we have

$$\|\nabla_{\mathbf{W}} f_h(\mathbf{W})\| = p'(h(\mathbf{W}))\|\nabla_{\mathbf{W}} h(\mathbf{W})\| \overset{(a)}{\geq} 0 \tag{15}$$

where (a) is true because from (8) we can know that the stationary points of problems $g(\mathbf{W})$ and $h(\mathbf{W})$ are the same within region $[\tau, 2\tau]$ and from [5, Theorem 1] we know that there is no critical points of $h(\mathbf{W})$ within region $[\tau, 2\tau]$. Therefore, we can conclude that no critical points are involved in this region for the constructed function $p(\mathbf{W})$. Applying the geometric structure results of the objective function shown before, we obtain the claim of Theorem 1. □

Algorithm 1 Perturbed Alternating Proximal Point (PA-PP)

Input: $\mathbf{W}^{(0)}$, ν, r, g_{th}, f_{th}, t_{th}
for $t = 0, 1, \ldots$ **do**
  $\mathbf{U}^{(t+1)} = \arg\min_{\mathbf{U}} f_h(\mathbf{U}, \mathbf{V}^{(t)}) + \frac{\nu}{2}\|\mathbf{U} - \mathbf{U}^{(t)}\|^2$
  $\mathbf{V}^{(t+1)} = \arg\min_{\mathbf{V}} f_h(\mathbf{U}^{(t+1)}, \mathbf{V}) + \frac{\nu}{2}\|\mathbf{V} - \mathbf{V}^{(t)}\|^2$
  if $\|\mathbf{W}^{(t+1)} - \mathbf{W}^{(t)}\| \leq g_{\mathsf{th}}^2$ and $t - t_{\mathsf{p}} > t_{\mathsf{th}}$ **then**
    $\widetilde{\mathbf{W}}^{(t)} \leftarrow \mathbf{W}^{(t)}$ and $t_{\mathsf{p}} \leftarrow t$
    $\mathbf{W}^{(t)} = \widetilde{\mathbf{W}}^{(t)} + \xi^{(t)}$, $\xi^{(t)}$ follows $\mathbb{B}_0(r)$
    $\mathbf{U}^{(t+1)} = \arg\min_{\mathbf{U}} f_h(\mathbf{U}, \mathbf{V}^{(t)}) + \frac{\nu}{2}\|\mathbf{U} - \mathbf{U}^{(t)}\|^2$
    $\mathbf{V}^{(t+1)} = \arg\min_{\mathbf{V}} f_h(\mathbf{U}^{(t+1)}, \mathbf{V}) + \frac{\nu}{2}\|\mathbf{V} - \mathbf{V}^{(t)}\|^2$
  end if
  if $t - t_{\mathsf{p}} = t_{\mathsf{th}}$ and $f_h(\mathbf{W}^{(t)}) - f_h(\widetilde{\mathbf{W}}^{(t_{\mathsf{p}})}) > -f_{\mathsf{th}}$ **then**
    return $\widetilde{\mathbf{W}}^{t_{\mathsf{p}}}$
  end if
end for

Remark 1. It is worth mentioning that the proposed loss function structure is not only applied in the asymmetric matrix factorization problems but also an option for symmetric matrix factorization related problems, phase retrieval problems, over-parameterization problems, etc.

4. ALGORITHM DESIGN

The loss function has been constructed well enough. Now, we need to develop an efficient algorithm by exploiting the block structure of the problem such that it is able to escape from the strict saddle points quickly. Before showing the details of the algorithm, we first need some assumptions of the objective function.

Assumption 1. *Function $f(\cdot)$ is L-smooth, ρ-Hessian Lipschitz, and block-wise smooth with gradient Lipschitz constants $\{L_U, L_V\}$.*

The function is called block-wise smooth with gradient Lipschitz constant L_V, if

$$\|\nabla_{\mathbf{V}} f(\mathbf{W}, \mathbf{V}) - \nabla_{\mathbf{V}} f(\mathbf{W}, \mathbf{V}')\| \leq L_V \|\mathbf{V} - \mathbf{V}'\|, \ \forall \mathbf{V}, \mathbf{V}'$$

or with gradient Lipschitz constant L_U, if

$$\|\nabla_{\mathbf{U}} f(\mathbf{U}, \mathbf{V}) - \nabla_{\mathbf{U}} f(\mathbf{U}', \mathbf{V})\| \leq L_U \|\mathbf{U} - \mathbf{U}'\|, \ \forall \mathbf{U}, \mathbf{U}'.$$

Further, let $L_{\max} \triangleq \max\{L_U, L_V\}$. Note that $L_{\max} \ll L$ in many block structured nonconvex optimization problems.

4.1. Algorithm Description

PA-PP is proposed in this section. It is a simple algorithm, which basically implements the ordinary alternating minimization at each iteration and add some random noise occasionally to extract the negative curvature when some conditions (iterates are very close to the first-order stationary points) are satisfied.

The details of the implementation of the PA-PP is given in Algorithm 1, where the regularizer is $\nu = L_{\max}/c_{\max}$, the radius of the ball added is $r = (c^3/\chi^3)(\rho\epsilon/L_{\max}\mathcal{P})$; the threshold of gradient size is $g_{\mathsf{th}} = c^2\epsilon/(\chi^3\mathcal{P})$; the threshold of the decrease of the objective value is $f_{\mathsf{th}} = c^5\epsilon^2/(L_{\max}\chi^6\mathcal{P}^2)$; the threshold of the number of iterations that the algorithm will not add any perturbation is $t_{\mathsf{th}} = L_{\max}\chi/(c^2(L_{\max}\rho\epsilon)^{\frac{1}{3}})$; and parameters $\mathcal{P} \triangleq (1 + L\log(2d)/L_{\max})$ and $\chi \triangleq 6\max\{\log(\frac{\mathcal{P}^2 d L_{\max}^{5/3}\Delta_f}{c^5\rho^{1/3}\epsilon^{7/3}\delta}), 4\}$.

Remark 2. Note the subproblems shown in the update of $\mathbf{U}^{(t+1)}$ and $\mathbf{V}^{(t+1)}$ are the least squares problems, which have the closed-form solutions. Solving these problems may not need to calculate the gradient of the objective function, resulting in a low computational complexity.

4.2. Convergence Rate

Due to the page limit, the details of the convergence analysis will be given in the journal version (see preprint [28] or [29]). Here, we only give the result as the following.

Theorem 2. *Under Assumption 1, there exists a constant* $c_{\max}$ *such that: for any* $\delta \in (0,1]$, $\epsilon \leq L_{\max}^2/\rho$, $\Delta_f \triangleq f_h(\mathbf{W}^{(0)}) - f^\star$, *and constant* $c \leq c_{\max}$, *with probability* $1-\delta$, *the iterates generated by PA-PP converge to an* ϵ*-SS2* $\mathbf{W}$ *satisfying*

$$\|\nabla f_h(\mathbf{W})\| \leq \epsilon, \quad \textit{and} \quad \lambda_{\min}(\nabla^2 f_h(\mathbf{W})) \geq -(L_{\max}\rho\epsilon)^{1/3}$$

in the following number of iterations:

$$\mathcal{O}\left(\frac{L_{\max}^{5/3}\mathcal{P}^2\Delta f}{\rho^{1/3}\epsilon^{7/3}}\log^7\left(\frac{\mathcal{P}^2 d L_{\max}^{5/3}\Delta_f}{c^5\rho^{1/3}\epsilon^{7/3}\delta}\right)\right) \tag{16}$$

where $f^\star$ *denotes the global minimum value of the objective function.*

Remark 3. Combining Theorem 2 and Theorem 1, we can conclude that PA-PP converges to the global optimal solution of the matrix factorization problem considered with high probability.

4.3. Connection with Existing Works

In Theorem 2 we characterized the convergence rate to an $(\epsilon, \epsilon^{1/3})$-SS2. We can also translate this bound to the one for achieving an $(\epsilon, \sqrt{\epsilon})$-SS2, and in this case PA-PP needs $\widetilde{\mathcal{O}}(1/\epsilon^{3.5})$ iterations. Compared with the existing recent works [24], the convergence rate of PA-PP is slower than GD. The main reason is the fact that different from GD-type algorithms, PA-PP cannot fully utilize the Hessian information because they never see a full iteration. A similar situation happens for SGD-type of algorithms which also cannot get the exact negative curvature around strict saddle points.

However, the convergence rate of PA-PP is still faster than SGD [2], SGLD [17], NEON+SGD [18], and NEON2+SGD [19] to achieve an $(\epsilon, \sqrt{\epsilon})$-SS2. We emphasize that PA-PP represents the first BCD-type algorithms with the convergence rate guarantee to escape from the strict saddle points efficiently. At this point, it is unclear whether our obtained rate is the best that is achievable, and the question of whether the resulting rate can be improved will be left as the future work.

5. NUMERICAL RESULTS

In this section, we will use several numerical results to illustrate the effectiveness of the proposed algorithm on escaping strict saddle points.

5.1. A toy example

First, we present a simple example that shows there are two equal local optimal solutions, i.e., the global optimal points. Consider a nonconvex objective function, i.e.,

$$f(\mathbf{w}) = \mathbf{w}^T\mathbf{A}\mathbf{w} + \frac{1}{4}\|\mathbf{w}\|^4, \tag{17}$$

where $\mathbf{w} = [u; v]$. Here, we can easily show the shape of objective function (17) in the two dimensional (2D) case in Figure 1 (left), where $\mathbf{A} = [1\ \ 2; 2\ \ 1] \in \mathbb{R}^{2\times 2}$. It can be observed clearly that there exists a strict saddle point at $[0, 0]$ and two other local optimal points. We randomly initialize the algorithms around strict saddle point $[0, 0]$. The convergence comparison between GD and PA-PP is shown in Figure 1 (right). It can be observed that PA-PP converges faster than GD to the global optimal point. Note that if we initialize the iterates exactly at the origin, GD will not move but PA-PP can still converge to the global optimal solution.

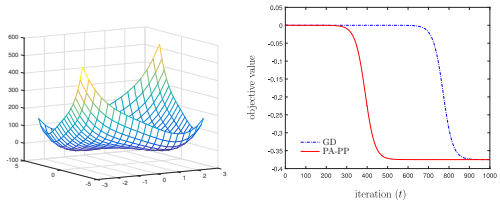

Fig. 1. Convergence comparison between GD and PA-PP, where $\epsilon = 10^{-4}$, $g_{\mathsf{th}} = \epsilon/10$, $\nu = 50$, $t_{\mathsf{th}} = 10/\epsilon^{1/3}$, $r = \epsilon/10$.

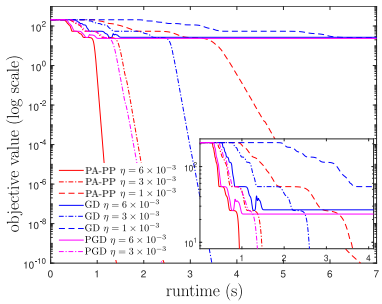

Fig. 2. Convergence comparison among GD, PGD and PA-PP for asymmetric matrix factorization, where $\epsilon = 10^{-10}$, $g_{\mathsf{th}} = \epsilon/10$, $t_{\mathsf{th}} = 10/\epsilon^{1/3}$, $r = \epsilon/10$.

5.2. Asymmetric matrix factorization

We also test the algorithm for the problem of asymmetric matrix factorization. We randomly generate matrix $\mathbf{M}^* = \mathbf{U}^*(\mathbf{V}^*)^T$ with dimension $n = 200, m = 20, r = 10$ and initialize GD, perturbed gradient descent (PGD) and PA-PP around saddle point 0. The step-sizes of the GD and PGD algorithms are denoted as η, which is equivalent to $1/\nu$ of PA-PP [1]. All perturbation related parameters of PA-PP and PGD, e.g., $g_{\mathsf{th}}, t_{\mathsf{th}}, r$, are the same. Figure 2 shows the superiority of PA-PP in the asymmetric matrix factorization problem. When the step-size is large, GD and PGD cannot decrease the objective value monotonically or sufficiently but PA-PP can, since the regularizer (or step-size) of PA-PP depends on $L_{\max}$ rather than L. PA-PP and PGD converge faster since the negative curvature can be captured by adding the random noise. Also, it can be observed that PA-PP converges to the global optimal solution of this problem.

6. CONCLUSION

In this paper, we studied the geometric structure of the asymmetric matrix factorization problems, where a modified new loss function is proposed such that the global Lipschitz continuity of the objective function can be satisfied without loss of optimality. PA-PP is applied to solve this nonconvex problem, where the convergence rate of the algorithm is also established. Numerical results show that the proposed algorithm is able to escape from saddle much faster than the existing methods.

7. ACKNOWLEDGMENT

The authors would like to thank Jason Lee and Meisam Razaviyayn for discussion on the landscape of the loss function.

[1] In Figure 1, the regularizer ν of PA-PP is shown by $\eta = 1/\nu$ for fairness comparison with GD and PGD.

8. REFERENCES

[1] K. Kawaguchi, "Deep learning without poor local minima," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2016, pp. 586–594.

[2] R. Ge, F. Huang, C. Jin, and Y. Yuan, "Escaping from saddle points — online stochastic gradient for tensor decomposition," in *Proceedings of Annual Conference on Learning Theory (COLT)*, 2015, pp. 797–842.

[3] J. Sun, Q. Qu, and J. Wright, "A geometric analysis of phase retrieval," *Foundations of Computational Mathematics*, vol. 18, no. 5, pp. 1131–1198, Oct. 2018.

[4] R. Ge, C. Jin, and Y. Zheng, "No spurious local minima in nonconvex low rank problems: A unified geometric analysis," in *Proceedings of International Conference on Machine Learning (ICML)*, 2017, pp. 1233–1242.

[5] Z. Zhu, Q. Li, G. Tang, and M. B. Wakin, "The global optimization geometry of low-rank matrix optimization," *arXiv:1703.01256*, 2018.

[6] S. Lu, J. D. Lee, M. Razaviyayn, and M. Hong, "Gradient primal-dual methods of solving linear constrained non-convex problems: Convergence, optimality, and applications," 2019, working paper.

[7] T. Zhao, Z. Wang, and H. Liu, "A nonconvex optimization framework for low rank matrix estimation," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2015, pp. 559–567.

[8] S. Lu, M. Hong, and Z. Wang, "A nonconvex splitting method for symmetric nonnegative matrix factorization: Convergence analysis and optimality," *IEEE Transactions on Signal Processing*, vol. 65, no. 12, pp. 3120–3135, June 2017.

[9] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1758–1789, 2013.

[10] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Proceedings of Annual ACM Symposium on Theory of Computing*, 2013, pp. 665–674.

[11] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of Optimization Theory and Applications*, vol. 109, no. 3, pp. 475–494, 2001.

[12] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust Region Methods*, SIAM, 2000.

[13] Y. Nesterov and B. T. Polyak, "Cubic regularization of Newton method and its global performance," *Mathematical Programming*, vol. 108, no. 1, pp. 177–205, 2006.

[14] Y. Carmon and J. C. Duchi, "Gradient descent efficiently finds the cubic-regularized non-convex Newton step," *arXiv preprint arXiv:1612.00547*, 2016.

[15] S. J. Reddi, M. Zaheer, S. Sra, B. Póczos, F. Bach, R. Salakhutdinov, and A. J. Smola, "A generic approach for escaping saddle points," in *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018, pp. 12330–1242.

[16] Y. Yu, D. Zou, and Q. Gu, "Saving gradient and negative curvature computations: Finding local minima more efficiently," *arXiv preprint arXiv:1712.03950*, 2017.

[17] Y. Zhang, P. Liang, and M. Charikar, "A hitting time analysis of stochastic gradient langevin dynamics," in *Proceedings of Annual Conference on Learning Theory (COLT)*, 2017, pp. 1980–2022.

[18] Y. Xu and T. Yang, "First-order stochastic algorithms for escaping from saddle points in almost linear time," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2018.

[19] Z. Allen-Zhu and Y. Li, "NEON2: Finding local minima via first-order oracles," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2018.

[20] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht, "Gradient descent only converges to minimizers," in *Proceedings of Annual Conference on Learning Theory (COLT)*, 2016, pp. 1246–1257.

[21] J. D. Lee, I. Panageas, G. Piliouras, M. Simchowitz, M. I. Jordan, and B. Recht, "First-order methods almost always avoid saddle points," *arXiv:1710.07406v1 [stat.ML]*, 2017.

[22] E. Song, Z. Shen, and Q. Shi, "Block coordinate descent almost surely converges to a stationary point satisfying the second-order necessary condition," *optimization-online*, 2017.

[23] S. S. Du, C. Jin, J. D. Lee, M. I. Jordan, B. Póczos, and A. Singh, "Gradient descent can take exponential time to escape saddle points," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2017.

[24] C. Jin, R. Ge, P. Netrapalli, S. M. Kakade, and M. I. Jordan, "How to escape saddle points efficiently," in *Proceedings of International Conference on Machine Learning (ICML)*, 2017, pp. 1724–1732.

[25] Y. Chi, Y. M. Lu, and Y. Chen, "Nonconvex optimization meets low-rank matrix factorization: An overview," *arXiv:1809.09573*, 2018.

[26] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Review*, vol. 52, no. 3, pp. 471–501, 2010.

[27] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning*, vol. 1, 2001.

[28] S. Lu, M. Hong, and Z. Wang, "On the sublinear convergence of randomly perturbed alternating gradient descent to second order stationary solutions," *arXiv:1802.10418*, 2018.

[29] S. Lu, "First-order methods of solving nonconvex optimization problems: Algorithms, convergence, and optimality," *Graduate Theses and Dissertations*, 2018.