Proximal Gradient Algorithm with Momentum and Flexible Parameter Restart for Nonconvex Optimization

Yi Zhou[1] , **Zhe Wang**[2] , **Kaiyi Ji**[2] , **Yingbin Liang**[2] and **Vahid Tarokh**[3]
[1]University of Utah
[2]The Ohio State University
[3]Duke University
yi.zhou@utah.edu, wang.10982@osu.edu, ji.367@osu.edu

Abstract

Various types of parameter restart schemes have been proposed for proximal gradient algorithm with momentum to facilitate their convergence in convex optimization. However, under parameter restart, the convergence of proximal gradient algorithm with momentum remains obscure in nonconvex optimization. In this paper, we propose a novel proximal gradient algorithm with momentum and parameter restart for solving nonconvex and nonsmooth problems. Our algorithm is designed to 1) allow for adopting flexible parameter restart schemes that cover many existing ones; 2) have a global sub-linear convergence rate in nonconvex and nonsmooth optimization; and 3) have guaranteed convergence to a critical point and have various types of asymptotic convergence rates depending on the parameterization of local geometry in nonconvex and nonsmooth optimization. Numerical experiments demonstrate the convergence and effectiveness of our proposed algorithm.

1 Introduction

Training modern machine learning models in real applications typically involves highly nonconvex optimization, and some effective interesting examples include deep learning [Wang *et al.*, 2019], nature language processing and computer vision, etc. To solve these nonconvex optimization problems, gradient-based algorithms [Nesterov, 2014] are popular choices due to their simplicity, effectiveness as well as well-understood convergence guarantees.

In practical training of machine learning models, momentum has been a successful and widely applied optimization trick that facilitates the convergence of gradient-based algorithms. Various types of momentum schemes have been developed, e.g., [Nesterov, 2014; Beck and Teboulle, 2009; Tseng, 2010; Ghadimi and Lan, 2016; Li and Lin, 2015], and have been shown to improve the order of convergence rates of gradient-based algorithms in solving convex and strongly convex optimization problems. In specific, gradient descent algorithms with momentum have been shown to achieve the complexity lower bound for convex optimization [Nesterov, 2014; Beck and Teboulle, 2009] and have guaranteed convergence in nonconvex optimization [Ghadimi and Lan, 2016; Li and Lin, 2015].

Despite the superior theoretical advantages of momentum acceleration schemes, they do not fully exploit the potential for acceleration. For example, the basic momentum scheme [Nesterov, 2014; Beck and Teboulle, 2009] adopts a diminishing momentum coefficient for accelerating smooth convex optimization, and it does not provide much momentum acceleration after a large number of iterations. Also, for accelerating strongly convex optimization, the choice of momentum coefficient requires the knowledge of condition number of the Hessian matrix, which is typically unknown a priori. To resolve these issues and further facilitate the practical convergence of gradient algorithms with momentum, various types of *parameter restart* techniques have been proposed, e.g., [O'Donoghue and Candès, 2015; Fercoq and Qu, 2016; 2017; Giselsson and Boyd, 2014; Kim and Fessler, 2018; Liang and Schonlieb, 2018; Lin and Xiao, 2015; Liu and Yang, 2017; Renegar and Grimmer, 2018; Roulet and dAspremont, 2017]. In these works, it has been demonstrated that restarting algorithm parameters (i.e., variables and momentum coefficient) periodically can suppress the oscillations of the training loss induced by the extrapolation step and improve the practical convergence in *convex* optimization. In specific, parameter restart is typically triggered by certain occurrences that may slow down the convergence, such as function value divergence [O'Donoghue and Candès, 2015; Renegar and Grimmer, 2018] and gradient mismatch [O'Donoghue and Candès, 2015], etc. Therefore, parameter restart can reduce the instability and oscillations caused by momentum. However, in *nonconvex* optimization, the applications of parameter restart to gradient algorithms with momentum require to deal with the following open issues.

(a) While the convergence of gradient algorithms with momentum and parameter restart have been well explored in *convex* optimization, they are of lack of theoretical understandings in *nonconvex* optimization, which are important for modern machine learning purpose. *(b)* Previous works on gradient algorithms with momentum and restart for convex optimization are based on very specific restart schemes in order to have convergence guarantee, but practically the best restart scheme can be problem dependent. Therefore, it is much desired to design a momentum scheme that al-

lows to adopt flexible parameter restart schemes with theoretical convergence guarantee. *(c)* The existing gradient algorithms with momentum for nonconvex optimization have convergence guarantees at the cost of either introducing extra computation steps [Li and Lin, 2015; Li *et al.*, 2017] or imposing restrictions on the objective function [Ghadimi and Lan, 2016]. It is important to explore whether parameter restart can help alleviate these costs or restrictions.

Considering all the issues above, we are motivated to design a gradient algorithm with momentum and parameter restart that *(a)* has convergence guarantee in nonconvex optimization, *(b)* allows to apply flexible restart schemes in practice and *(c)* avoids the existing weakness and restrictions in design of accelerated methods for nonconvex optimization. We summarize our contributions as follows.

1.1 Our Contributions

We consider the problem of minimizing a smooth nonconvex function plus a (non)smooth regularizer. To solve such a class of problems, we propose APG-restart: a momentum-accelerated proximal gradient algorithm with parameter restart (see Algorithm 1) and show that APG-restart satisfies the following properties.

- APG-restart allows for adopting any parameter restart scheme (hence covers many existing ones). In particular, it guarantees to make monotonic progress on function value between successive restart periods of iterations.
- The design of the proximal momentum component in APG-restart leverages the notion of generalized gradient mapping (see eq. (4)), which leads to convergence guarantee in nonconvex optimization. Also, APG-restart does not require extra computation steps compared to other accelerated algorithms for nonconvex optimization [Li *et al.*, 2017; Li and Lin, 2015], and removes the restriction of bounded domain on the regularizer function in existing works [Ghadimi and Lan, 2016].
- APG-restart achieves the stationary condition at a global sublinear convergence rate (see Lemma 1).
- Under the Kurdyka-Łojasiewicz (KŁ) property of nonconvex functions (see Definition 2), the variable sequence generated by APG-restart is guaranteed to converge to a critical point. Moreover, the asymptotic convergence rates of function value and variable sequences generated by APG-restart are fully characterized by the parameterization of the KŁ property of the objective function. This work is the first study of gradient methods with momentum and parameter restart under the KŁ property.

1.2 Related Works

Gradient algorithms with momentum and parameter restart: Various types of parameter restart schemes have been proposed for accelerated gradient-based algorithms for convex optimization. Specifically, [O'Donoghue and Candès, 2015] proposed to restart the accelerated gradient descent algorithm whenever certain function value-based criterion or gradient-based criterion is violated. These restart schemes were shown to achieve the optimal convergence rate without prior knowledge of the condition number of the function. [Giselsson and Boyd, 2014] further proposed an accelerated gradient algorithm with restart and established formal convergence rate analysis for smooth convex optimization. [Lin and Xiao, 2015] proposed a restart scheme that automatically estimates the strong convexity parameter and achieves a near-optimal iteration complexity. [Fercoq and Qu, 2016; 2017] proposed a restart scheme for accelerated algorithms that achieves a linear convergence in convex optimization under the quadratic growth condition. [Liu and Yang, 2017; Roulet and dAspremont, 2017] studied convergence rate of accelerated algorithms with restart in convex optimization under the error bound condition and the Łojasiewicz condition, respectively. [Renegar and Grimmer, 2018] proposed a restart scheme that is based on achieving a specified amount of decrease in function value. All these works studied accelerated gradient algorithms with restart in convex optimization, whereas this work focuses on nonconvex optimization.

Nonconvex optimization under KŁ property: The Kurdyka-Łojasiewicz property is a generalization of the Łojasiewicz gradient inequality for smooth analytic functions to nonsmooth sub-analytic functions. Such a local property was then widely applied to study the asymptotic convergence behavior of various gradient-based algorithms in nonconvex optimization [Attouch and Bolte, 2009; Bolte *et al.*, 2014; Zhou *et al.*, 2016; 2018b]. The KŁ property has also been applied to study convergence properties of accelerated gradient algorithms [Li *et al.*, 2017; Li and Lin, 2015] and heavy-ball algorithms [Ochs, 2018; Liang *et al.*, 2016] in nonconvex optimization. Some other works exploited the KŁ property to study the convergence of second-order algorithms in nonconvex optimization, e.g., [Zhou *et al.*, 2018a].

2 Preliminaries

In this section, we introduce some definitions that are useful in our analysis later. Consider a proper[1] and lower-semicontinuous function $h : \mathbb{R}^d \to \mathbb{R}$ which is *not* necessarily smooth nor convex. We introduce the following generalized notion of derivative for the function h.

Definition 1. *(Subdifferential and critical point, [Rockafellar and Wets, 1997]) The Frechét subdifferential $\widehat{\partial} h$ of function h at $x \in \operatorname{dom} h$ is the set of $u \in \mathbb{R}^d$ defined as*

$$\widehat{\partial} h(x) := \left\{ u : \liminf_{z \neq x, z \to x} \frac{h(z) - h(x) - u^\intercal (z - x)}{\|z - x\|} \geq 0 \right\},$$

and the limiting subdifferential ∂h at $x \in \operatorname{dom} h$ is the graphical closure of $\widehat{\partial} h$ defined as:

$$\partial h(x) := \{u : \exists x_k \to x, h(x_k) \to h(x), u_k \in \widehat{\partial} h(x_k) \to u\}.$$

The set of critical points of h is defined as $\operatorname{crit} h := \{x : \mathbf{0} \in \partial h(x)\}$.

Note that when the function h is continuously differentiable, the limiting sub-differential ∂h reduces to the usual

[1] An extended real-valued function h is proper if its domain $\operatorname{dom} h := \{x : h(x) < \infty\}$ is nonempty.

notion of gradient ∇h. Next, we introduce the Kurdyka-Łojasiewicz (KŁ) property of a function h. Throughout, we define the distance between a point $x \in \mathbb{R}^d$ and a set $\Omega \subseteq \mathbb{R}^d$ as $\operatorname{dist}_\Omega(x) := \inf_{w\in\Omega} \|x - w\|$.

Definition 2. *(KŁ property, [Bolte* et al.*, 2014]) A proper and lower-semicontinuous function h is said to satisfy the KŁ property if for every compact set $\Omega \subset \operatorname{dom} h$ on which h takes a constant value $h_\Omega \in \mathbb{R}$, there exist $\varepsilon, \lambda > 0$ such that for all $x \in \{z \in \mathbb{R}^d : \operatorname{dist}_\Omega(z) < \varepsilon, h_\Omega < h(z) < h_\Omega + \lambda\}$, the following inequality is satisfied*

$$\varphi'\left(h(x) - h_\Omega\right) \operatorname{dist}_{\partial h(x)}(\mathbf{0}) \geq 1, \tag{1}$$

where φ' is the derivative of function $\varphi : [0, \lambda) \to \mathbb{R}_+$, which takes the form $\varphi(t) = \frac{c}{\theta} t^\theta$ for some $c > 0, \theta \in (0, 1]$.

To elaborate, consider the case where h is differentiable. Then, the KŁ property in eq. (1) can be rewritten as

$$h(x) - h_\Omega \leq C\|\nabla h(x)\|^p \tag{2}$$

for some constant $C > 0$ and $p \in (1, +\infty)$. In fact, Equation (2) can be viewed as a generalization of the gradient dominance condition that corresponds to the special case of $p = 2$. A large class of functions have been shown to satisfy the KŁ property, e.g., sub-analytic functions, logarithm and exponential functions, etc [Bolte *et al.*, 2007]. These function classes cover most of nonconvex objective functions encountered in practical applications, e.g., logistic loss, vector and matrix norms, rank, and polynomial functions, etc. Please refer to [Bolte *et al.*, 2014, Section 5] and [Attouch *et al.*, 2010, Section 4] for more example functions.

To handle non-smooth objective functions, we introduce the following notion of proximal mapping.

Definition 3. *(Proximal mapping) For a proper and lower-semicontinuous function h, its proximal mapping at $x \in \mathbb{R}^d$ with parameter $\eta > 0$ is defined as:*

$$\operatorname{prox}_{\eta h}(x) := \operatorname*{argmin}_{z\in\mathbb{R}^d} \left\{ h(z) + \frac{1}{2\eta}\|z - x\|^2 \right\}. \tag{3}$$

3 APG-restart for Nonsmooth & Nonconvex Optimization

In this section, we propose a novel momentum-accelerated proximal gradient with parameter restart (referred to as APG-restart) for solving nonsmooth and nonconvex problems.

Consider the composite optimization problem of minimizing a smooth and nonconvex function $f : \mathbb{R}^d \to \mathbb{R}$ plus a possibly nonsmooth and convex function $g : \mathbb{R}^d \to \mathbb{R}$, which is written as

$$\min_{x\in\mathbb{R}^d} F(x) := f(x) + g(x). \tag{P}$$

We adopt the following standard assumptions on the objective function F in the problem (P).

Assumption 1. *The objective function F in the problem* (P) *satisfies:*

1. *Function F is bounded below, i.e., $F^* := \inf_{x\in\mathbb{R}^d} F(x) > -\infty$;*
2. *For any $\alpha \in \mathbb{R}$, the level set $\{x : F(x) \leq \alpha\}$ is compact;*
3. *The gradient of f is L-Lipschitz continuous and g is lower-semicontinuous and convex.*

Under Assumption 1, we further introduce the following mapping for any $\eta > 0$ and $x, u \in \mathbb{R}^d$:

$$G_\eta(x, u) := \frac{1}{\eta}\big(x - \operatorname{prox}_{\eta g}(x - \eta u)\big). \tag{4}$$

Such a mapping is well-defined and single-valued due to the convexity of g. Moreover, the critical points of function F (cf., Definition 1) in the problem (P) can be alternatively characterized as $\operatorname{crit} F := \{x : \mathbf{0} \in G_\eta(x, \nabla f(x))\}$. Therefore, $G_\eta(x, \nabla f(x))$ serves as a type of 'gradient' at point x, and we refer to such a mapping as *gradient mapping* in the rest of the paper. In particular, the gradient mapping reduces to the usual notion of gradient when the nonsmooth part $g \equiv 0$.

Algorithm 1 APG-restart for nonconvex optimization

Input: $K \in \mathbb{N}$, restart periods $q_0 = 0, \{q_t\}_{t\geq 1} \in \mathbb{N}$, stepsizes $\{\lambda_k\}_k, \{\beta_k\}_k > 0$.
Define: $Q_t := \sum_{\ell=0}^{t} q_\ell$.
Initialize: $x_{-1} \in \mathbb{R}^d$.
for $k = 0, 1, \ldots, K$ **do**
  Denote t the largest integer such that $Q_t \leq k$,
  Set: $\alpha_k = \frac{2}{k - Q_t + 2}$,
  if $k = Q_t$ *for some* $t \in \mathbb{N}$ **then**
    Reset: $x_k = y_k = x_{k-1}$,
  end
  $z_k = (1 - \alpha_{k+1})y_k + \alpha_{k+1}x_k$,
  $x_{k+1} = x_k - \lambda_k G_{\lambda_k}(x_k, \nabla f(z_k))$,
  $y_{k+1} = z_k - \beta_k G_{\lambda_k}(x_k, \nabla f(z_k))$.
end
Output: x_K.

To solve the nonsmooth and nonconvex problem (P), we propose the APG-restart algorithm that is presented in Algorithm 1. APG-restart consists of new design of momentum schemes for updating the variables x_k and y_k, the extrapolation step for updating the variable z_k where α_{k+1} denotes the associated momentum coefficient, and the restart periods $\{q_t\}_t$. We next elaborate the two major ingredients of APG-restart: new momentum design and flexible restart scheduling with convergence guarantee.

New momentum design: We adopt new momentum steps in APG-restart for updating the variables x_k and y_k, which are different from those of the AG method in [Ghadimi and Lan, 2016] and we compare our update rules with theirs as follows.

$$\text{(APG-restart):} \begin{cases} x_{k+1} = x_k - \lambda_k G_{\lambda_k}(x_k, \nabla f(z_k)), \\ y_{k+1} = z_k - \beta_k G_{\lambda_k}(x_k, \nabla f(z_k)). \end{cases} \tag{5}$$

$$\text{(AG):} \begin{cases} x_{k+1} = \operatorname{prox}_{\lambda_k g}(x_k - \lambda_k \nabla f(z_k)) \\ y_{k+1} = \operatorname{prox}_{\lambda_k g}(z_k - \beta_k \nabla f(z_k)) \end{cases} \tag{6}$$

It can be seen from the above comparison that our APG-restart uses the same gradient mapping term

$G_{\lambda_k}(x_k, \nabla f(z_k))$ to update both of the variables x_k and y_k, while the AG algorithm in [Ghadimi and Lan, 2016] updates them using different proximal gradient terms. Consequently, our APG-restart is more computationally efficient as it requires to compute one gradient mapping per iteration while the AG algorithm needs to perform two proximal updates. On the other hand, the update rules of the AG algorithm guarantee convergence in nonconvex optimization only for functions of g with bounded domain [Ghadimi and Lan, 2016]. Such a restriction rules out regularization functions with unbounded domain, which are commonly used in practical applications, e.g., ℓ_1, ℓ_2 regularization, elastic net, etc. In comparison, as we show in the analysis later, the update rules of APG-restart has guaranteed convergence in nonconvex optimization and does not require the regularizer g to be domain-bounded.

Guarantee for any restart scheduling: APG-restart retains the convergence guarantee with any restart scheduling. In specific, by specifying an arbitrary sequence of iteration periods $\{q_t\}_t \in \mathbb{N}$, APG-restart calls the restart operation at the end of each period (i.e., whenever $k = Q_t$ for some t). Upon restart, both x_k and y_k are reset to be the variable x_{k-1} generated at the previous iteration, and the momentum coefficient α_k is reset to be 1. In the subsequent iterations, the momentum coefficient is diminished inversely proportionally to the number of iterations within the restart period.

Since our APG-restart retains convergence guarantee for any restart periods $\{q_t\}_t$, it can implement any criterion that determines when to perform the parameter restart and have a convergence guarantee (see our analysis later). We list in Table 1 some popular restart criteria from existing literature and compare their practical performance under our APG-restart framework in the experiment section later. We note that the restart criterion of the gradient mapping scheme implicitly depends on the gradient mapping, as $y_{k+1} - z_k \propto G_{\lambda_k}(x_k, \nabla f(z_k))$ from the update rule in Algorithm 1.

Performing parameter restart has appealing benefits. First, synchronizing the variables x_k and y_k periodically can suppress the deviation between them caused by the extrapolation step. This further helps to reduce the oscillation of the generated function value sequence. Furthermore, restarting the momentum coefficient α_k periodically injects more momentum into the algorithm dynamic, and therefore facilitates the practical convergence of the algorithm.

4 Convergence Analysis of APG-restart

In this section, we study the convergence properties of APG-restart in solving nonconvex and nonsmooth optimization problems. We first characterize the algorithm dynamic of APG-restart.

Lemma 1. *[Algorithm dynamic] Let Assumption 1 hold and apply Algorithm 1 to solve the problem (P). Set $\beta_k \equiv \frac{1}{8L}$ and $\lambda_k \in [\beta_k, (1+\alpha_{k+1})\beta_k]$. Then, the sequence $\{x_k\}_k$ generated by APG-restart satisfies: for all $t = 1, 2, ...$*

$$F(x_{Q_t}) \leq F(x_{Q_{t-1}}) - \frac{L}{4} \sum_{k=Q_{t-1}}^{Q_t-1} \|x_{k+1} - x_k\|^2, \quad (7)$$

$$\mathrm{dist}^2_{\partial F(x_{Q_t})}(\mathbf{0}) \leq 162L^2 \sum_{k=Q_{t-1}}^{Q_t-1} \|x_{k+1} - x_k\|^2. \quad (8)$$

Lemma 1 characterizes the period-wise algorithm dynamic of APG-restart. In specific, eq. (7) shows that the function value sequence generated by APG-restart is guaranteed to decrease between two adjacent restart checkpoint (i.e., Q_{t-1} and Q_t), and the corresponding progress $F(x_{Q_{t-1}}) - F(x_{Q_t})$ is bounded below by the square length of the iteration path between the restart checkpoints, i.e., $\sum_{k=Q_{t-1}}^{Q_t-1} \|x_{k+1} - x_k\|^2$. On the other hand, eq. (8) shows that the norm of the subdifferential at the t-th restart checkpoint is bounded by the square length of the same iteration path. In summary, the algorithm dynamic of APG-restart is different from that of traditional gradient-based algorithms in several aspects: First, the dynamic of APG-restart is characterized at the restart checkpoints, while the dynamic of gradient descent is characterized iteration-wise [Attouch and Bolte, 2009; Attouch *et al.*, 2013]. As we elaborate later, such a property makes the convergence analysis of APG-restart more involved; Second, APG-restart makes monotonic progress on the function value between two adjacent restart checkpoints. In other accelerated gradient algorithms, such a monotonicity property is achieved by introducing a function value check step [Li and Lin, 2015] or an additional proximal gradient step [Li *et al.*, 2017].

Based on the algorithm dynamic in Lemma 1, we obtain the following global convergence rate of APG-restart for nonconvex and nonsmooth optimization. Throughout the paper, we denote $f(n) = \Theta(g(n))$ if and only if for some $0 < c_1 < c_2$, $c_1 g(n) \leq f(n) \leq c_2 g(n)$ for all $n \geq n_0$.

Theorem 1. *[Global convergence rate] Under the same conditions as those of Lemma 1, the sequence $\{z_k\}_k$ generated by APG-restart satisfies: for all $K = 1, 2, ...$*

$$\min_{0 \leq k \leq K-1} \|G_{\lambda_k}(z_k, \nabla f(z_k))\|^2 \leq \Theta\Big(\frac{L\big(F(x_0) - F^*\big)}{K}\Big).$$

Theorem 1 establishes the global convergence rate of APG-restart in terms of the gradient mapping, which we recall characterizes the critical point of the nonconvex objective function F. In particular, the order of the above global convergence rate matches that of other accelerated gradient algorithms [Ghadimi and Lan, 2016] for nonconvex optimization, and APG-restart further benefits from the flexible parameter restart scheme that provides extra acceleration in practice (as we demonstrate via experiments later).

Theorem 1 does not fully capture the entire convergence property of APG-restart. To elaborate, convergence of the gradient mapping in Theorem 1 does not necessarily guarantee the convergence of the *variable sequence* generated by APG-restart. On the other hand, the convergence rate estimate is based on the global Lipschitz condition of the objective function, which may not capture the local geometry of the

Restart scheme	Fixed restart [Nesterov, 2007]	Function value [O'Donoghue and Candès, 2015] [Giselsson and Boyd, 2014] [Kim and Fessler, 2018]	Gradient mapping [O'Donoghue and Candès, 2015] [Giselsson and Boyd, 2014] [Kim and Fessler, 2018]	Non-monotonic [Giselsson and Boyd, 2014]
Check condition	$q_t \equiv q \in \mathbb{N}$ for all t	restart whenever $F(x_k) > F(x_{k-1})$	restart whenever $\langle z_k - y_k, y_{k+1} - z_k \rangle \geq 0$	restart whenever $\langle z_k - y_k, y_{k+1} - \frac{z_k + x_k}{2} \rangle \geq 0$

Table 1: Restart conditions for different parameter restart schemes.

function around critical points and therefore leads to a coarse convergence rate estimate in the asymptotic regime. To further explore stronger convergence results of APG-restart, we next exploit the ubiquitous Kurdyka-Łojasiewicz (KŁ) property (cf., Definition 2) of nonconvex functions. We make the following assumption.

Assumption 2. *The objective function F in the problem* (P) *satisfies the KŁ property.*

Based on the algorithm dynamic in Lemma 1 and further leveraging the KŁ property of the objective function, we obtain the following convergence result of APG-restart in nonconvex optimization.

Theorem 2. *[Variable convergence] Let Assumptions 1 and 2 hold and apply Algorithm 1 to solve the problem (P). Set $\beta_k \equiv \frac{1}{8L}$ and $\lambda_k \in [\beta_k, (1+\alpha_{k+1})\beta_k]$. Define the length of iteration path of the t-th restart period as $L_t := \sqrt{\sum_{k=Q_t}^{Q_{t+1}-1} \|x_{k+1} - x_k\|^2}$. Then, the sequence $\{L_t\}_t$ generated by APG-restart satisfies: for all periods of iterations $t = 1, 2, ...$*

$$\sum_{t=0}^{\infty} L_t < +\infty. \tag{9}$$

Consequently, the variable sequences $\{x_k\}_k, \{y_k\}_k, \{z_k\}_k$ generated by APG-restart converge to the same critical point of the problem (P), i.e.,

$$x_k, y_k, z_k \xrightarrow{k} x^* \in \operatorname{crit} F. \tag{10}$$

Theorem 2 establishes the formal convergence of APG-restart in nonconvex optimization. We note that such a convergence guarantee holds for any parameter restart schemes, therefore demonstrating the flexibility and generality of our algorithm. Also, unlike other accelerated gradient-type of algorithms that guarantee only convergence of function value [Li and Lin, 2015; Li *et al.*, 2017], our APG-restart is guaranteed to generate convergent variable sequences to a critical point in nonconvex optimization.

To highlight the proof technique, we first exploit the dynamic of APG-restart in Lemma 1 to characterize the limit points of the sequences $\{x_{Q_t}\}_t, \{F(x_{Q_t})\}_t$ that are indexed by the restart checkpoints. Then, we further show that the entire sequences $\{x_k\}_k, \{F(x_k)\}_k$ share the same limiting properties, which in turn guarantee the sequences to enter a local parameter region of the objective function where the KŁ property can be exploited. Taking advantage of the KŁ property, we are able to show that the length of the optimization path is finite as iteration $k \to \infty$. Consequently, the generated variable sequences can be shown to converge to a certain critical point of the Problem (P).

Besides the variable convergence guarantee under the KŁ property, we also obtain various types of convergence rate estimates of APG-restart depending on the specific parameterization of the local KŁ property of the objective function. We obtain the following results.

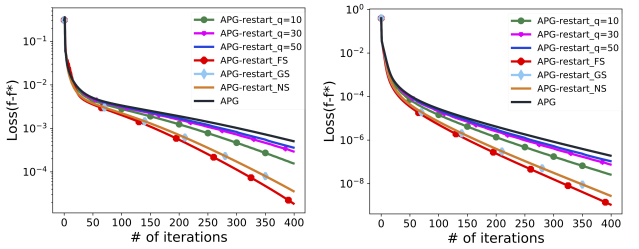
Theorem 3. *[Convergence rate of function value] Let Assumptions 1 and 2 hold and apply Algorithm 1 to solve the problem (P). Set $\beta_k \equiv \frac{1}{8L}$ and $\lambda_k \in [\beta_k, (1+\alpha_{k+1})\beta_k]$. Suppose the algorithm generates a sequence $\{x_k\}_k$ that converges to a certain critical point x^* where the KŁ property holds with parameter $\theta \in (0, 1]$. Then, there exists a sufficiently large $t_0 \in \mathbb{N}$ such that for all $t \geq t_0$,*

1. *If $\theta = 1$, then $F(x_{Q_t}) \downarrow F(x^*)$ within finite number of periods of iterations;*
2. *If $\theta \in [\frac{1}{2}, 1)$, then $F(x_{Q_t}) \downarrow F(x^*)$ linearly as $F(x_{Q_t}) - F(x^*) \leq \exp\big(-\Theta(t - t_0)\big)$;*
3. *If $\theta \in (0, \frac{1}{2})$, then $F(x_{Q_t}) \downarrow F(x^*)$ sub-linearly as $F(x_{Q_t}) - F(x^*) \leq \Theta\Big((t - t_0)^{-\frac{1}{1-2\theta}}\Big)$.*

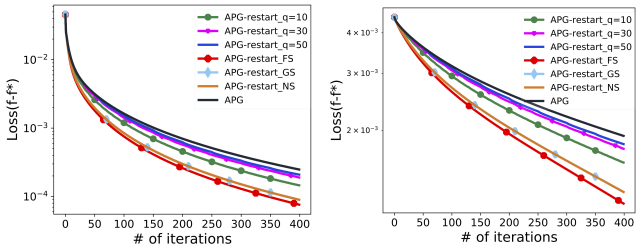
Theorem 4. *[Convergence rate of variable] Under the same conditions as those of Theorem 3, suppose APG-restart generates a sequence $\{x_k\}_k$ that converges to a certain critical point x^* where the KŁ property holds with parameter $\theta \in (0, 1]$. Then, there exists a sufficiently large $t_0 \in \mathbb{N}$ such that for all $t \geq t_0$,*

1. *If $\theta = 1$, then $x_{Q_t} \xrightarrow{t} x^*$ within finite number of periods of iterations;*
2. *If $\theta \in [\frac{1}{2}, 1)$, then $x_{Q_t} \xrightarrow{t} x^*$ linearly as $\|x_{Q_t} - x^*\| \leq \exp\big(-\Theta(t - t_0)\big)$;*
3. *If $\theta \in (0, \frac{1}{2})$, then $x_{Q_t} \xrightarrow{t} x^*$ sub-linearly as $\|x_{Q_t} - x^*\| \leq \Theta\Big((t - t_0)^{-\frac{\theta}{1-2\theta}}\Big)$.*

Theorem 3 and Theorem 4 establish the asymptotic convergence rate results for the function value sequence and variable sequence generated by APG-restart, respectively. Intuitively, after a sufficiently large number of training iterations, APG-restart enters a local neighborhood of a certain critical point. In such a case, the global convergence rate characterized in Theorem 1 can be a coarse estimate because it exploits only the global Lipschitz property of the function. On the contrary, the local KŁ property characterizes the function geometry in a more accurate way and leads to the above tighter convergence

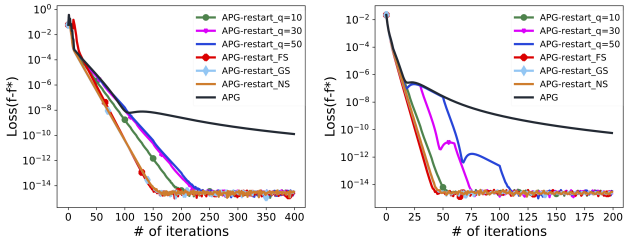


(a) logistic regression a9a (b) logistic regression w8a

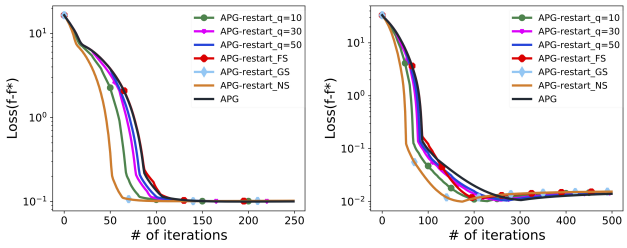


(c) robust regression a9a (d) robust regression w8a

Figure 1: Comparison of different restart schemes in smooth nonconvex optimization.



(a) logistic regression a9a (b) logistic regression w8a



(c) robust regression a9a (d) robust regression w8a

Figure 2: Comparison of different restart schemes in *nonsmooth* nonconvex optimization.

rate estimates. In particular, the KŁ parameter θ captures the 'sharpness' of the local geometry of the function, i.e., a larger θ induces a faster convergence rate.

5 Experiments

In this section, we implement the APG-restart algorithm with different restart schemes listed in Table 1 to corroborate our theory that APG-restart has guaranteed convergence with any restart scheme. In specific, for the fixed restart scheme we set the restart period to be $q = 10, 30, 50$, respectively.

We first solve two smooth nonconvex problems, i.e., the logistic regression problem with a nonconvex regularizer (i.e., $g(x) := \alpha \sum_{i=1}^{d} \frac{x_i^2}{1+x_i^2}$) and the robust linear regression problem. For the logistic regression problem, we adopt the cross-entropy loss and set $\alpha = 0.01$, and for the robust linear regression problem, we adopt the robust nonconvex loss $\ell(s) := \log(\frac{s^2}{2} + 1)$. We test both problems on two LIBSVM datasets: a9a and w8a [Chang and Lin, 2011]. We use stepsizes $\beta_k = 1, \lambda_k = (1 + \alpha_{k+1})\beta_k$ for the APG-restart as suggested by our theorems. We note that in these experiments, we plot the loss gap versus number of iterations for all algorithms. The comparison of running time is similar as all the algorithms require the same computation per iteration.

Figure 1 shows the experiment results of APG-restart with fixed scheme (constant q), function value scheme (FS), gradient mapping scheme (GS) and non-monotone scheme (NS). It can be seen that APG-restart under the function scheme performs the best among all restart schemes. In fact, the function scheme restarts the APG algorithm the most often in these experiments. The gradient mapping scheme and the non-monotone scheme have very similar performance, and both of them perform slightly worse than the function scheme. Moreover, the fixed restart schemes have the worst performance. In particular, the performance of fixed scheme gets better as the restart period q decreases (i.e., more restarts take place).

Next, we further add a nonsmooth ℓ_1 norm regularizer to the objective functions of all the problems mentioned above, and apply APG-restart with different restart schemes to solve them. The results are shown in Figure 2. One can see that for the nonsmooth logistic regression, all the non-fixed restart schemes have comparable performances and they perform better than the fixed restart schemes. For the nonsmooth robust linear regression, both the gradient mapping scheme and the non-monotone scheme outperform the other schemes. In this experiment, the function scheme has a degraded performance that is comparable to the fixed restart schemes. This is possibly due to the highly nonconvexity of the loss landscape.

6 Conclusion

In this paper, we propose a novel accelerated proximal gradient algorithm with parameter restart for nonconvex optimization. Our proposed APG-restart allows for adopting any parameter restart schemes and have guaranteed convergence. We establish both the global convergence rate and various types of asymptotic convergence rates of the algorithm, and we demonstrate the effectiveness of the proposed algorithm via numerical experiments. We expect that such a parameter restart algorithm framework can inspire new design of optimization algorithms with faster convergence for solving nonconvex machine learning problems.

Acknowledgment

The work of Z. Wang, K. Ji and Y. Liang was supported in part by the U.S. National Science Foundation under the grants CCF-1761506, CCF-1909291 and CCF-1900145.

References

[Attouch and Bolte, 2009] H. Attouch and J. Bolte. On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. *Mathematical Programming*, 116(1-2):5–16, 2009.

[Attouch *et al.*, 2010] H. Attouch, J. Bolte, P. Redont, and A. Soubeyran. Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality. *Mathematics of Operations Research*, 35(2):438–457, 2010.

[Attouch *et al.*, 2013] H. Attouch, J. Bolte, and B. Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized Gauss–Seidel methods. *Mathematical Programming*, 137(1):91–129, Feb 2013.

[Beck and Teboulle, 2009] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, March 2009.

[Bolte *et al.*, 2007] J. Bolte, A. Daniilidis, and A. Lewis. The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems. *SIAM Journal on Optimization*, 17:1205–1223, 2007.

[Bolte *et al.*, 2014] J. Bolte, S. Sabach, and M. Teboulle. Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494, 2014.

[Chang and Lin, 2011] C. Chang and C. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):1–27, 2011.

[Fercoq and Qu, 2016] O. Fercoq and Z. Qu. Restarting accelerated gradient methods with a rough strong convexity estimate. *ArXiv:1609.07358v1*, Sep 2016.

[Fercoq and Qu, 2017] O. Fercoq and Z. Qu. Adaptive restart of accelerated gradient methods under local quadratic growth condition. *arXiv:1709.02300v1*, Sep 2017.

[Ghadimi and Lan, 2016] S. Ghadimi and G. Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, March 2016.

[Ghadimi *et al.*, 2016] S. Ghadimi, G. Lan, and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1):267–305, Jan 2016.

[Giselsson and Boyd, 2014] P. Giselsson and S. Boyd. Monotonicity and restart in fast gradient methods. In *Proc. IEEE Conference on Decision and Control (CDC)*, pages 5058–5063, Dec 2014.

[Kim and Fessler, 2018] D. Kim and J.A. Fessler. Adaptive restart of the optimized gradient method for convex optimization. *Journal of Optimization Theory and Applications*, 178(1):240–263, Jul 2018.

[Li and Lin, 2015] H. Li and Z. Lin. Accelerated proximal gradient methods for nonconvex programming. In *Proc. International Conference on Neural Information Processing Systems (NIPS)*, pages 379–387, 2015.

[Li *et al.*, 2017] Q. Li, Y. Zhou, Y. Liang, and P. K. Varshney. Convergence analysis of proximal gradient with momentum for nonconvex optimization. In *Proc. International Conference on Machine Learning (ICML*, pages 2111–2119, Aug 2017.

[Liang and Schonlieb, 2018] J. Liang and C. Schonlieb. Improving FISTA: Faster, Smarter and Greedier. *arXiv:1811.01430v2*, Nov 2018.

[Liang *et al.*, 2016] Jingwei Liang, Jalal M. Fadili, and Gabriel Peyré. A multi-step inertial forward-backward splitting method for non-convex optimization. In *Proc. Neural Information Processing Systems (NIPS)*, pages 4042–4050, 2016.

[Lin and Xiao, 2015] Q. Lin and L. Xiao. An adaptive accelerated proximal gradient method and its homotopy continuation for sparse optimization. *Computational Optimization and Applications*, 60(3):633–674, Apr 2015.

[Liu and Yang, 2017] M. Liu and T. Yang. Adaptive accelerated gradient converging method under holderian error bound condition. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3114. 2017.

[Nesterov, 2007] Y. Nesterov. Gradient methods for minimizing composite objective function. Core discussion papers, 2007.

[Nesterov, 2014] Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2014.

[Ochs, 2018] P. Ochs. Local convergence of the heavy-ball method and iPiano for non-convex optimization. *Journal of Optimization Theory and Applications*, 177(1):153–180, Apr 2018.

[O'Donoghue and Candès, 2015] B. O'Donoghue and E. Candès. Adaptive restart for accelerated gradient schemes. *Foundations of Computational Mathematics*, 15(3):715–732, Jun 2015.

[Renegar and Grimmer, 2018] J. Renegar and B. Grimmer. A simple nearly-optimal restart scheme for speeding-up first order methods. *arXiv:1803.00151v1*, Mar 2018.

[Rockafellar and Wets, 1997] R.T. Rockafellar and R.J.B. Wets. *Variational Analysis*. Springer, 1997.

[Roulet and dAspremont, 2017] V. Roulet and A. dAspremont. Sharpness, restart and acceleration. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 1119–1129. 2017.

[Tseng, 2010] P. Tseng. Approximation accuracy, gradient methods, and error bound for structured convex optimization. *Mathematical Programming*, 125(2):263–295, Oct 2010.

[Wang *et al.*, 2019] G. Wang, G. B. Giannakis, and J. Chen. Learning ReLU networks on linearly separable data: Algorithm, optimality, and generalization. *IEEE Transactions on Signal Processing*, 67(9):2357–2370, 2019.

[Zhou *et al.*, 2016] Y. Zhou, Y. Yu, W. Dai, Y. Liang, and P. Xing. On convergence of model parallel proximal gradient algorithm for stale synchronous parallel system. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS*, pages 713–722, May 2016.

[Zhou *et al.*, 2018a] Y. Zhou, Z. Wang, and Y. Liang. Convergence of cubic regularization for nonconvex optimization under KŁ property. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[Zhou *et al.*, 2018b] Y. Zhou, Y. Yu, W. Dai, Y. Liang, and E. P. Xing. Distributed proximal gradient algorithm for partially asynchronous computer clusters. *Journal of Machine Learning Research (JMLR)*, 19(19):1–32, 2018.