Nesterov's First Order Method Accelerations: Unifications, Applications and Numerial Experiments

Hongda Li

Department of Mathematics
University of British Columbia,
Okanagan Campus.

February 5, 2025

Contents

1 Introduction

Let $\mathbb{R}^n$ be the ambient space. We consider

$$\min_{x\in\mathbb{R}^n} \{F(x) := f(x) + g(x)\}. \tag{1.1}$$

Unless specified, assume $f : \mathbb{R}^n \to \mathbb{R}$ is L-Lipscthiz smooth $\mu \geq 0$ strongly convex and $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ is closed convex and proper. This type of problem is referred to as additive composite problems in the literature.

Our ongoing research concerns accelerated proximal gradient type method for solving (1.1). In the expository writing by Walkington [33], a variant for of accelerated gradient method for strongly convex function f is discussed. We had some lingering questions after reading it.

(i) Do there exist a unified description for the convergence for both variants of the algorithms?

(ii) Is it possible to attain fast convergence rate without knowledge about the strong convexity of function f?

(iii) Is it possible to describe the convergence of function value for momentum sequences that are much weaker than the Nesterov's rule?

The good news is we have definitive answers for all questions in our draft paper.

In this proposal we explore the Goldilocks zones between theories and practices of optimization algorithms. Our topics are Nesterov's acceleration and the method of Catalyst Acceleration which is realization of the theories of Nesterov's acceleration and inexact proximal point method in the settings of variance reduced method in Machine Learning.

Organizations now follows. In section 3 we propose the method of "Relaxed Weak Accelerated Proximal Gradient (R-WAPG)" unifies the convergence results of several Euclidean variants of Accelerated Proximal Gradient (APG) method in the literatures and in addition to claiming the convergence when the momentum sequence doesn't follow the Nesterov's update rule.

Section 4 gives an algorithm which is a reformulation of R-WAPG and, it achieves competitive convergence results (1.1) without restarting and knowing parameter L, μ in prior. Numerical experiments are presented and potential future direction of research is given by the end of the section for the R-WAPG framework.

Section 5 reviews the series of papers [24, 25, 34] on Catalyst Meta Acceleration for First Order Variance Reduced Methods. The content complements report completed for MATH 590 Fall Winter 2024. The end of Section 5 points out potential future direction of research of Catalyst Acceleration.

2 Preliminaries

This section contains the basics of contents from convex optimization, and variational analysis. Throughout, we adopt the notation $\overline{\mathbb{R}} := \mathbb{R} \cup \{\infty, -\infty\}$.

2.1 Fundamentals in non-convex analysis

Let the ambient space be $\mathbb{R}^n$ equipped with inner product $\langle \cdot, \cdot \rangle$ and 2-norm $\|\cdot\|$. Let O be an open subset of $\mathbb{R}^n$, the weakest assumption we make for objective function $F : O \subseteq \mathbb{R}^n \to \mathbb{R}$ of an optimization problem is local Lipschitz continuity. The assumption of local Lipschitz continuity is weak enough to describe most problems in applications, and strong enough to avoid most pathologies in analysis.

Definition 2.1 (Local Lipschitz continuity) *Let $F : O \subseteq \mathbb{R}^n \to \mathbb{R}$ be Locally Lipschitz and O is an open set. Then for all $\bar{x} \in O$, there exists a Neighborhood: $\mathcal{N}(\bar{x})$ and $K \in \mathbb{R}$ such that for all $x, y \in \mathcal{N}(\bar{x})$: $|F(x) - F(y)| \leq K\|x - y\|$.*

Definition 2.2 (Regular subgradient) *Let $F : O \subseteq \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz and $\bar{x} \in O$. The regular subdifferential at $\bar{x}$ is defined as*

$$\widehat{\partial} F(\bar{x}) := \left\{ v \in \mathbb{R}^n \;\middle|\; \liminf_{\bar{x} \neq x \to \bar{x}} \frac{F(x) - F(\bar{x}) - \langle v, x - \bar{x} \rangle}{\|x - \bar{x}\|} \geq 0 \right\}.$$

Remark 2.3 Definition taken from Definition 4.3.1 from Pang, Cui's book [44]

Definition 2.4 (Limiting subgradient) *Let $F : O \subseteq \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz and $\bar{x} \in O$. The limiting subdifferential at $\bar{x}$ is defined as*

$$\partial F(\bar{x}) := \left\{ v \in \mathbb{R}^n \;\middle|\; \exists x_k \to \bar{x}, v_k \to v : v_k \in \widehat{\partial} F(x_k) \; \forall k \in \mathbb{N} \right\}.$$

Remark 2.5 Definition taken from Definition 4.3.1 from Pang, Cui's book [44]

Definition 2.6 (Weakly convex function) *$F : \mathbb{R}^n \to \overline{\mathbb{R}}$ is μ weakly convex if and only if $F + \frac{\mu}{2}\|\cdot\|^2$ is convex.*

Definition 2.7 (Bregman divergence) *Let $F : O \subseteq \mathbb{R}^n \to \mathbb{R}$ be a differentiable function. Then the Bregman divergence of F is defined as:*

$$D_F(x, y) : O \times \mathrm{dom}(\partial F) \to \mathbb{R} := F(x) - F(y) - \langle \nabla F(y), x - y \rangle.$$

2.2 Fundamentals in convex analysis

This section introduces the classics and basics of convex analysis. Define F to be closed, proper and convex in this section. When F is convex, the limiting subgradient and the regular subgradient reduced to the following:

$$\partial F(x) = \{v \in \mathbb{R}^n \mid \forall y \in \mathbb{R}^n \ : F(y) - F(x) \geq \langle v, y - x \rangle \}.$$

A convex function is locally Lipschitz in the relative interior of its domain, denoted as $\mathrm{ri}(\mathrm{dom}(F))$. So it has $\mathrm{ri}(\mathrm{dom}\, F) \subseteq \mathrm{dom}(\partial F) \subseteq \mathrm{dom}\, F$. See Rockafellar's book [37, pg 82].

When we say $F : \mathbb{R}^n \to \mathbb{R}$ is L Lipschitz smooth function, it means that there exists L such that for all $x \in \mathbb{R}^n, y \in \mathbb{R}^n$, it has:

$$\|\nabla F(x) - \nabla F(y)\| \leq L\|x - y\|.$$

When F convex, then it has descent lemma:

$$(\forall x \in \mathbb{R}^n)(\forall y \in \mathbb{R}^n) : 0 \leq F(x) - F(y) - \langle \nabla f(y), x - y \rangle \leq \frac{L}{2}\|x - y\|^2.$$

The converse holds under convexity as well. The definitions that follow narrow things further.

Definition 2.8 (Strong convexity) *A function $F : \mathbb{R}^n \to \overline{\mathbb{R}}$ is $\mu \geq 0$ strongly convex if and only if for all $y \in \mathrm{dom}(\partial F)$, $x \in \mathbb{R}^n$:*

$$(\forall v \in \partial F(x)) \quad F(x) - F(y) \geq \langle v, x - y \rangle + \frac{\mu}{2}\|x - y\|^2.$$

Lemma 2.9 (Quadratic growth from strong convexity) *If F is $\mu \geq 0$ strongly convex, $\bar{x}$ is a minimizer of F. Then for all $x \in \mathbb{R}^n$*

$$F(x) - F(\bar{x}) \geq \frac{\mu}{2}\|x - \bar{x}\|^2.$$

Remark 2.10 The minimizer is unique whenever $\mu > 0$. For contradiction, assume $x \neq \bar{x}$ is another minimizer, then $F(x) \neq F(\bar{x})$, which is a direct contradiction. This condition is called quadratic growth over the set of minimizer, it is much weaker than strong convexity.

2.2.1 Smooth, nonsmooth additive composite

This section zooms in further into the case of additive composite objective $F := f + g$. Assume f is L Lipschitz smooth and $\mu \geq 0$ strongly convex, $g : \mathbb{R} \to \overline{\mathbb{R}}$ is closed convex. For all $\beta \geq 0$, define the proximal gradient, proximal point model functions as a mapping of $\mathbb{R}^n \times \mathbb{R}^n \to \overline{\mathbb{R}}$:

$$\widetilde{\mathcal{M}}_F^{\beta^{-1}}(x; y) := g(x) + f(y) + \langle \nabla f(y), x - y \rangle + \frac{\beta}{2}\|x - y\|^2,$$
$$\mathcal{M}_F^{\beta^{-1}}(x; y) := F(x) + \frac{\beta}{2}\|x - y\|^2.$$

Under the assumptions of this section, $\widetilde{\mathcal{M}}_F^{\beta^{-1}}(\cdot; y), \mathcal{M}_F^{\beta^{-1}}(\cdot; y)$ are both $\beta+\mu$ strongly convex.

Definition 2.11 (Proximal gradient operator) *Define the proximal gradient operator T_L on all $y \in \mathbb{R}^n$:*

$$T_L y := \operatorname*{argmin}_{x \in \mathbb{R}^n} \left\{ g(x) + f(y) + \langle \nabla f(y), x - y \rangle + \frac{L}{2}\|x - y\|^2 \right\}.$$

Remark 2.12 Under the assumption of this section, the mapping T_L is a single-valued mapping on $\mathbb{R}^n$ and it's a $3/2$ averaged operator.

Definition 2.13 (Gradient mapping operator) *Define the gradient mapping operator $\mathcal{G}_L$ on all $y \in \mathbb{R}^n$:*

$$\mathcal{G}_L(y) := L(y - T_L y).$$

Lemma 2.14 (The proximal gradient inequality) *For all $y \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, it has:*

$$F(x) - F(T_L y) - \langle L(y - T_L y), x - y \rangle - \frac{\mu}{2}\|x - y\|^2 - \frac{L}{2}\|y - T_L y\|^2 \geq 0.$$

The proof of the lemma appeals to Lemma 2.9 with the $L+\mu$ strong convexity of $\widetilde{\mathcal{M}}_F^{\beta^{-1}}(\cdot; y)$.

2.3 Nesterov's estimating sequence technique

The derivation of APG was originally conceived by Nesterov's estimating sequence technique. We emphasize that the technique derives the algorithm and proves its convergence rate.

The method is widespread in the literatures, and the ideas behind it are tremendously useful. Güler [20] used it to derive an accelerated proximal point method, which was instrumental to develop the Catalyst Acceleration framework. Nesterov [29] used it to conceive

the accelerated cubic regularized Newton's method. In (6.1.19) of Nesterov's book [32], it's used to derive a method of accelerated mirror descent. Finally, in Geovani N. et al [19], they used it to derive an accelerated Newton's method for convex composite objective function.

The definition of the estimating sequence that follows is based on our own understanding of the estimating sequence.

Definition 2.15 (Nesterov's estimating sequence) *For all $k \geq 0$, let $\phi_k : \mathbb{R}^n \to \mathbb{R}$ be a sequence of functions. We call this sequence of functions a Nesterov's estimating sequence when it satisfies conditions:*

(i) *There exists another sequence $(x_k)_{k\geq 0}$ such that for all $k \geq 0$ it has $F(x_k) \leq \phi_k^* := \min_x \phi_k(x)$.*

(ii) *There exists a sequence of $(\alpha_k)_{k\geq 0}$ where $\alpha_k \in (0, 1)$ $\forall k \geq 0$ such that for all $x \in \mathbb{R}^n$ it has $\phi_{k+1}(x) - \phi_k(x) \leq -\alpha_k(\phi_k(x) - F(x))$.*

3 Unifying NAG, and weakening the sequence assumption for convergences

This section is based on our unpublished draft paper. The Nesterov's acceleration scheme which was originally proposed in 1983 [28] is a celebrated first order method for solving the minimization problem. Here, NAG stands for Nesterov's Accelerated Gradient. It refers to a general class of momentum method where the gradient are evaluated on an extrapolated iterate using the previous two iterates. Initialize $x_1 = y_1$ and $t_0 = 1$, the algorithm finds $(x_k)_{k\geq 1}$ for all $k \geq 1$ by:

$$x_{k+1} = y_k - L^{-1}\nabla F(y_k), \tag{3.1}$$

$$t_{k+1} = 1/2\left(1 + \sqrt{1 + 4t_k^2}\right), \tag{3.2}$$

$$\theta_{k+1} = (t_k - 1)/t_{k+1}, \tag{3.3}$$

$$y_{k+1} = x_{k+1} + \theta_{k+1}(x_{k+1} - x_k). \tag{3.4}$$

If the minimizer x^* exists, then the optimality gap $F(x_k) - F(x^*)$ decreases at a rate of $\mathcal{O}(1/k^2)$, a faster rate compared to gradient descent which is $\mathcal{O}(1/k)$. It's consider optimal in some sense and see Chapter 2 of Nesterov's book [32] for details.

3.1 Contributions

Inspired specifically by the technique of Nestrov's estimating sequence [32], firstly we present a unified framework of Accelerated Proximal Gradient (APG) which we call Relaxed Weak Accelerated Proximal Gradient (R-WAPG) in Section 3.3. It has the ability to upper bound $F(x_k) - F(x^*)$ for sequences $(t_k)_{k\geq 0}$ that follows a rule much weaker than Nesterov's update rule. In addition to a convergence claim of $F(x_k) - F(x^*)$ for a much more flexible choice of $(t_k)_{k\geq 1}$. Secondly, we present an alternative to restarting that performs well empirically inspired by a small detail in the convergence proof of R-WAPG. It also has descriptive power to describe several variants of FISTA in the literatures.

Our contributions are two folds, theoretical and practical. Our results are based the assumption $F = f + g$ where $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex proper and closed, and f is an L-Lipschitz smooth and $\mu \geq 0$ strongly convex function.

A summary of our main results follow. Nesterov's acceleration extrapolate $y_{k+1} = x_{k+1} + \theta_{k+1}(x_{k+1} - x_k)$ where $\theta_{k+1} = (t_k - 1)/t_{k+1} \in (0, 1)$ is the "momentum". The choices for θ_k varies for different variants of the accelerated proximal gradient algorithm. In Chambolle, Dossal [9], it has $t_k = (n + a - 1)/a$ for all $a > 2$ which gives weak convergence of the iterates x_k in Hilbert space. In Chapter 10 of Beck's Book [3], a variant called V-FISTA can achieve the faster linear convergence rate: $\mathcal{O}((1 - \sqrt{\mu/L})^k)$ on the optimality gap for $\mu > 0$ strongly convex F. V-FISTA has $\theta_t = (\sqrt{\kappa} - 1)/(\sqrt{\kappa} + 1)$ where $\kappa = \mu/L$.

We relax the traditional choice of the sequence θ_k in Equation 3 and showed an upper bound of the optimal gap. Let $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ be two sequences that satisfy

$$\begin{aligned} &\alpha_0 \in (0, 1], \\ &\alpha_k \in (\mu/L, 1) \quad (\forall k \geq 1), \\ &\rho_k := \frac{\alpha_{k+1}^2 - (\mu/L)\alpha_{k+1}}{(1 - \alpha_{k+1})\alpha_k^2} \quad \forall (k \geq 0). \end{aligned}$$

Our first main result shows that if $\theta_{k+1} = (\rho_k \alpha_k (1 - \alpha_k)/(\rho_k \alpha_k^2 + \alpha_{k+1}))$, using the R-WAPG we proposed in Definition 3.5 with Proposition 3.6, 3.15, we can show that the gap $F(x_k) - F(x^*)$ is bounded by:

$$\mathcal{O}\left(\left(\prod_{i=0}^{k-1} \max(1, \rho_k)\right) \prod_{i=1}^{k} (1 - \alpha_i)\right).$$

Our second main result shows that for any $\alpha_k \geq 0$ the choice of sequence $\alpha_k = a/(a + k)$ results in $\rho_k > 1$ for all $k \in \mathbb{N}$ such that R-WAPG reduces to a variant of FISTA proposed in Chambolle, Dossal [9], and we are able to show the same convergence rate in Theorem 3.19. When $\rho_k = 1, \mu = 0$, R-WAPG reduces perfectly to FISTA by Beck [3], if $\mu > 0, \rho_k = 1$,

it reduces to the V-FISTA by Beck [3]. In Theorem 3.21, it demonstrates that R-WAPG frameworks gives a linear convergence claim for all fixed momentum method where $\alpha_k := \alpha \in (\mu/L, 1)$ and F is $\mu > 0$ strongly convex. Finally, we did the tedious work and present three equivalent forms of R-WAPG in Section 3.4 that are comparable to notable Euclidean variants of FISTA and beyond. This shows the descriptive power of our R-WAPG framework.

Our practical contribution is an algorithm inspired by a detail in our convergence proof which we call it "Parameter Free R-WAPG" (See Algorithm 1). The algorithm is parameter free, meaning that it doesn't require knowing L, μ in advance, and it determines the value of θ_t by estimating the local concavity using iterates y_k, y_{k+1} from the Bregman divergence of f with minimal computational cost. We conducted ample amount of numerical experiments to show that it has a favorable convergence rate in practice and behaves similarly to the FISTA with monotone restart.

Section 3.2 states and proves Proposition 3.3, an inequality based on a single generic iterative step that acts similar to a descent lemma. Section 3.3 formulates the full R-WAPG in Definition 3.5 and defines the R-WAPG sequence in Definition 3.4. Proposition 3.6 states the convergence rate of R-WAPG through Proposition 3.3 and the R-WAPG sequence. Section 3.4 gives three equivalent representations of the R-WAPG algorithms that are comparable to instances of APG found in the literatures. Section 3.5 formulates FISTA, and V-FISTA sequences as instance of the R-WAPG sequences. The section proves the convergence rate of several variants of FISTA using the equivalent forms introduced in Section 3.4 and the convergence rate developed Section 3.3. Finally, in Section 4 gives a formulation of a parameter free version of R-WAPG algorithm and showcase the numerical experiments for regression, LASSO.

3.2 Stepwise formulation of weak accelerated proximal gradient

The goal of this section is to build the R-WAPG algorithm which is described in the Definition 3.5 of the next section.

Definition 3.2 which describes what happens at a single iteration of the R-WAPG algorithm. It defines a procedure of generating x_{k+1}, v_{k+1} given any x_k, v_k. Proposition 3.3 states the inequality that describes a decreasing quantity that involves $F(x_k), F(x_{k+1})$ at each single iteration.

Assumption 3.1 Given x_k, y_k, v_k where $k \in \mathbb{Z}_+$, we define the following quantities

$$g_k := L(y_k - T_L y_k), \tag{3.5}$$

$$l_F(x; y_k) := F(T_L y_k) + \langle g_k, x - y_k \rangle + \frac{1}{2L}\|g_k\|^2, \tag{3.6}$$

$$\epsilon_k := F(x_k) - l_F(x_k; y_k), \tag{3.7}$$

Observe that by convexity of F, $\epsilon_k \geq 0$ for all $x_k, L > 0$. To see, use Theorem 2.14 and let $y = y_k, x = x_k$ which gives:

$$\begin{aligned} F(x_k) - F(T_L y_k) - \langle L(y_k - T_L y_k), x_k - y_k \rangle - \frac{L}{2}\|y_k - T_L y_k\|^2 - \frac{\mu}{2}\|x_k - y_k\|^2 \geq 0 \\ \iff F(x_k) - F(T_L y_k) - \langle g_k, x_k - y_k \rangle - \frac{1}{2L}\|g_k\|^2 \geq 0. \end{aligned}$$

The proposition follows provides upper bound to $F(x_{k+1})$ in relations to $F(x_k)$.

Definition 3.2 (Stepwise weak accelerated proximal gradient)
Assume $0 \leq \mu < L$. Fix any $k \in \mathbb{Z}_+$. For any $(v_k, x_k), \alpha_k \in (0,1), \gamma_k > 0$, let $\hat{\gamma}_{k+1}$, and vectors y_k, v_{k+1}, x_{k+1} be given by:

$$\hat{\gamma}_{k+1} = (1 - \alpha_k)\gamma_k + \mu\alpha_k, \tag{3.8}$$

$$y_k = (\gamma_k + \alpha_k \mu)^{-1}(\alpha_k \gamma_k v_k + \hat{\gamma}_{k+1} x_k), \tag{3.9}$$

$$g_k = \mathcal{G}_L y_k, \tag{3.10}$$

$$v_{k+1} = \hat{\gamma}_{k+1}^{-1}(\gamma_k(1 - \alpha_k)v_k - \alpha_k g_k + \mu\alpha_k y_k), \tag{3.11}$$

$$x_{k+1} = T_L y_k. \tag{3.12}$$

Proposition 3.3 (Stepwise Lyapunov)
Let $k \in \mathbb{Z}_+$, $R_k \in \mathbb{R}$. Given any v_k, x_k and $\gamma_k > 0$ and $v_{k+1}, x_{k+1}, y_k, \hat{\gamma}_{k+1}, \alpha_k$ that satisfies Definition 3.2. Define:

$$R_{k+1} := \frac{1}{2}\left(L^{-1} - \frac{\alpha_k^2}{\hat{\gamma}_{k+1}}\right)\|g_k\|^2 + (1 - \alpha_k)\left(\epsilon_k + R_k + \frac{\mu\alpha_k\gamma_k}{2\hat{\gamma}_{k+1}}\|v_k - y_k\|^2\right). \tag{3.13}$$

Then for all $x^* \in \mathbb{R}^n$, we have:

$$F(x_{k+1}) - F(x^*) + R_{k+1} + \frac{\hat{\gamma}_{k+1}}{2}\|v_{k+1} - x^*\|^2 \leq (1 - \alpha_k)\left(F(x_k) - F(x^*) + R_k + \frac{\gamma_k}{2}\|v_k - x^*\|^2\right). \tag{3.14}$$

The proof of the above proposition made extensive use of definition of R_{k+1}, and it appeals to Theorem 2.14.

3.3 R-WAPG and its convergence rates

In this section we propose Relaxed Weak Accelerated Proximal Gradient (R-WAPG), see Definition 3.5. R-WAPG algorithm generates iterates (x_k, y_k, v_k) and admits an upper bound on $F(x_k) - F^*$ described in Proposition 3.3. Definition 3.4 introduces the concept of an R-WAPG sequences: $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ which is crucial. The sequences parameterize the R-WAPG algorithm stated in Definition 3.5, it connects the step-wise formulation of R-WAPG (Definition 3.2) and can describe the convergence claim of R-WAPG in Proposition 3.6. In the next section, it continues to play a crucial role in describing several equivalent forms of the R-WAPG algorithm, and their corresponding convergence claim.

Definition 3.4 (R-WAPG sequences)
Assume $0 \leq \mu < L$. The sequences $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ are valid for R-WAPG if all the following holds:

$$\begin{aligned} &\alpha_0 \in (0, 1], \\ &\alpha_k \in (\mu/L, 1) \quad (\forall k \geq 1), \\ &\rho_k := \frac{\alpha_{k+1}^2 - (\mu/L)\alpha_{k+1}}{(1-\alpha_{k+1})\alpha_k^2} \quad \forall (k \geq 0). \end{aligned}$$

*We call $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ the **R-WAPG Sequences**.*

We now give Relaxed Weak Accelerated Proximal Gradient in details.

Definition 3.5 (Relaxed weak accelerated proximal gradient (R-WAPG))
Choose any $x_1 \in \mathbb{R}^n, v_1 \in \mathbb{R}^n$. Let $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ be given by Definition 3.4. The algorithm generates a sequence of vector $(y_k, x_{k+1}, v_{k+1})_{k\geq 1}$ for $k \geq 1$ by the procedures:

For $k = 1, 2, 3, \cdots$

$$\begin{aligned} \gamma_k &:= \rho_{k-1} L \alpha_{k-1}^2, \\ \hat{\gamma}_{k+1} &:= (1-\alpha_k)\gamma_k + \mu\alpha_k = L\alpha_k^2, \\ y_k &= (\gamma_k + \alpha_k \mu)^{-1}(\alpha_k \gamma_k v_k + \hat{\gamma}_{k+1} x_k), \\ g_k &= \mathcal{G}_L y_k, \\ v_{k+1} &= \hat{\gamma}_{k+1}^{-1}(\gamma_k(1-\alpha_k)v_k - \alpha_k g_k + \mu\alpha_k y_k), \\ x_{k+1} &= T_L y_k. \end{aligned}$$

Here is the main result of this section.

Proposition 3.6 (R-WAPG convergence claim)
Fix any arbitrary $x^* \in \mathbb{R}^n, N \in \mathbb{N}$. Let vector sequence $(y_k, v_k, x_k)_{k\geq 1}$ and R-WAPG sequences α_k, ρ_k be given by Definition 3.5. Define $R_1 = 0$ and suppose that for $k = 1, 2, \ldots, N$, we have R_k recursively given by:

$$R_{k+1} := \frac{1}{2}\left(L^{-1} - \frac{\alpha_k^2}{\hat{\gamma}_{k+1}}\right)\|g_k\|^2 + (1-\alpha_k)\left(\epsilon_k + R_k + \frac{\mu\alpha_k\gamma_k}{2\hat{\gamma}_{k+1}}\|v_k - y_k\|^2\right).$$

Then for all $k = 1, 2, \ldots, N$:

$$\begin{aligned} &F(x_{k+1}) - F(x^*) + \frac{L\alpha_k^2}{2}\|v_{k+1} - x^*\|^2 \\ &\leq \left(\prod_{i=0}^{k-1}\max(1,\rho_i)\right)\left(\prod_{i=1}^{k}(1-\alpha_i)\right)\left(F(x_1) - F(x^*) + \frac{L\alpha_0^2}{2}\|v_1 - x^*\|^2\right). \end{aligned}$$

3.4 Equivalent representations of R-WAPG

This section reduces Definition 3.5 into simpler forms that are comparable to what commonly appears in the literatures. In the literatures, variants of Accelerated Proximal Gradient algorithm such as FISTA, V-FISTA has different representations. This shows that R-WAPG provides a unified framework. These equivalent representations are listed in Definition 3.7, 3.9 and 3.11. These forms are equivalent under a subset of initial conditions.

They are comparable to existing APG algorithms in the literatures such as Exercise 12.1 in Ryu, Yin [38], Similar Triangle from Lee et al. [23], Ahn Sra [1] and momentum form of (2.2.19) in Nesterov [32]. Specific instances of Accelerated Proximal Gradient algorithm that has the same form as the Definition 3.7, Definition 3.9 and Definition 3.11 in the literatures are stated in the remarks that follows the definitions.

Definition 3.7 (R-WAPG intermediate form)
Assume $\mu < L$ and let $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ given by Definition 3.4. Initialize any x_1, v_1 in $\mathbb{R}^n$. For $k \geq 1$, the algorithm generates sequence of vector iterates $(y_k, v_{k+1}, x_{k+1})_{k\geq 1}$ by the procedures:

For $k = 1, 2, \ldots$

$$
\begin{aligned}
y_k &= \left(1 + \frac{L - L\alpha_k}{L\alpha_k - \mu}\right)^{-1} \left(v_{k+1} + \left(\frac{L - L\alpha_k}{L\alpha_k - \mu}\right) x_k\right), \\
x_{k+1} &= y_k - L^{-1}\mathcal{G}_L y_k, \\
v_{k+1} &= \left(1 + \frac{\mu}{L\alpha_k - \mu}\right)^{-1} \left(v_k + \left(\frac{\mu}{L\alpha_k - \mu}\right) y_k\right) - \frac{1}{L\alpha_k}\mathcal{G}_L y_k.
\end{aligned}
$$

Remark 3.8 This form of APG is rarely identified in the literatures. The closest algorithm that fits the form but with $\mu = 0$ is Chapter 12 of in Ryu and Yin's Book [38], right after Theorem 17. We created this form which makes the math that follows simpler. The inspiration of using this as an intermediate representation was inspired by solving Exercise 12.1 in the same Ryu and Yin's Book.

Definition 3.9 (R-WAPG similar triangle form)
Given any (x_1, v_1) *in* $\mathbb{R}^n$. *Assume* $\mu < L$. *Let the sequence* $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ *be given by Definition 3.4. For* $k \geq 1$, *the algorithm generates sequences of vector iterates* $(y_k, v_{k+1}, x_{k+1})_{k\geq 1}$ *by the procedures:*

For $k = 1, 2, \ldots$

$$
\begin{aligned}
y_k &= \left(1 + \frac{L - L\alpha_k}{L\alpha_k - \mu}\right)^{-1} \left(v_k + \left(\frac{L - L\alpha_k}{L\alpha_k - \mu}\right) x_k\right), \\
x_{k+1} &= y_k - L^{-1}\mathcal{G}_L y_k, \\
v_{k+1} &= x_{k+1} + (\alpha_k^{-1} - 1)(x_{k+1} - x_k).
\end{aligned}
$$

Remark 3.10 The word similar triangle form can be traced back to several literatures. The term "Method of Similar Triangle" was used for Algorithm (6.1.19) in Nesterov's book [32], but without the necessary graphical illustrations to clarify it. Equation (2), (3), (4) in [9] is a similar triangle formulation of FISTA with $\mu = 0$. To see graphical visualization on why such term is used to describe the APG algorithm in the literatures, see (3.1, 4.1) in Lee et al. [23] and Ahn and Sra [1].

Definition 3.11 (R-WAPG momentum form) *Given any* $y_1 = x_1 \in \mathbb{R}^n$, *and sequences* $(\rho_k)_{k\geq 0}, (\alpha_k)_{k\geq 0}$ *Definition 3.4. The algorithm generates iterates* x_{k+1}, y_{k+1} *For* $k = 1, 2, \cdots$ *by the procedures:*

For $k = 1, 2, \ldots$

$$x_{k+1} = y_k - L^{-1}\mathcal{G}_L y_k,$$
$$y_{k+1} = x_{k+1} + \frac{\rho_k \alpha_k (1-\alpha_k)}{\rho_k \alpha_k^2 + \alpha_{k+1}}(x_{k+1} - x_k).$$

In the special case where $\mu = 0$*, the momentum term can be represented without relaxation parameter* ρ_k*:*

$$(\forall k \geq 1) \quad \frac{\rho_k \alpha_k (1-\alpha_k)}{\rho_k \alpha_k^2 + \alpha_{k+1}} = \alpha_{k+1}(\alpha_k^{-1} - 1).$$

Remark 3.12 This format fits with (2.2.19) in Nesterov's book [32], however, the sequence $(\alpha_k)_{k\geq 0}$ would be given by a different rule. See Theorem 3.19 and Lemma 3.16 to see a specific choice of $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ such this equivalent form of R-WAPG is in fact two possible variants of the FISTA algorithm.

The following propositions state the relations between the three representation of R-WAPG and R-WAPG as stated in Definition 3.5.

Proposition 3.13 simplifies Definition 3.5 and finds a representation without using auxiliary sequence $\gamma_k, \hat{\gamma}_k$. Definition 3.7 states the first simplified form of the R-WAPG algorithm which we call: "R-WAPG intermediate form". Following a similar pattern, Proposition 3.14, 3.15 demonstrates two more equivalent representations of the R-WAPG intermediate form (Definition 3.7) which are formulated into Definition 3.9, 3.11. Convergence results from Proposition 3.6 applies to all these equivalent forms of R-WAPG. In brief, different equivalent reformulations are summarized as following:

Definition 3.5 $\iff$ Definition 3.7	By Proposition 3.13.
Definition 3.7 $\iff$ Definition 3.9	By Proposition 3.9.
Definition 3.9 $\implies$ Definition 3.11.	By Proposition 3.15.

Proposition 3.13 (First equivalent representation of R-WAPG)
If the sequence $(y_k, v_k, x_k)_{k\geq 1}$ *is produced by R-WAPG (Definition 3.5), then the iterates can be expressed without* $(\gamma_k)_{k\geq 1}, (\hat{\gamma}_k)_{k\geq 2}$*, and for all* $k \geq 1$ *namely*

$$y_k = \left(1 + \frac{L - L\alpha_k}{L\alpha_k - \mu}\right)^{-1}\left(v_k + \left(\frac{L - L\alpha_k}{L\alpha_k - \mu}\right)x_k\right),$$
$$x_{k+1} = y_k - L^{-1}\mathcal{G}_L y_k,$$
$$v_{k+1} = \left(1 + \frac{\mu}{L\alpha_k - \mu}\right)^{-1}\left(v_k + \left(\frac{\mu}{L\alpha_k - \mu}\right)y_k\right) - \frac{1}{L\alpha_k}\mathcal{G}_L y_k.$$

Proposition 3.14 (Second equivalent representation of R-WAPG)
Let iterates $(y_k, x_k, v_k)_{k\geq 1}$ and sequence $(\alpha_k, \rho_k)_{k\geq 0}$ be given by Definition 3.7. Then for all $k \geq 1$, iterate y_k, x_{k+1}, v_{k+1} satisfy:

$$\begin{aligned} y_k &= \left(1 + \frac{L - L\alpha_k}{L\alpha_k - \mu}\right)^{-1} \left(v_k + \left(\frac{L - L\alpha_k}{L\alpha_k - \mu}\right) x_k\right), \\ x_{k+1} &= y_k - L^{-1}\mathcal{G}_L y_k, \\ v_{k+1} &= x_{k+1} + (\alpha_k^{-1} - 1)(x_{k+1} - x_k). \end{aligned}$$

Proposition 3.15 (Third equivalent representation of R-WAPG)
Let sequence $(\alpha_k, \rho_k)_{k\geq 0}$ and iterates $(x_k, v_k, y_k)_{k\geq 1}$ given by R-WAPG intermediate form (Definition 3.9). Then for all $k \geq 1$, the iterates $(x_{k+1}, y_{k+1})_{k\geq 1}$ satisfy:

$$\begin{aligned} x_{k+1} &= y_k - L^{-1}\mathcal{G}_L y_k, \\ y_{k+1} &= x_{k+1} + \frac{\rho_k \alpha_k (1 - \alpha_k)}{\rho_k \alpha_k^2 + \alpha_{k+1}} (x_{k+1} - x_k). \end{aligned}$$

If in addition, $v_1 = x_1$ then

$$y_1 = \left(1 + \frac{L - L\alpha_1}{L\alpha_1 - \mu}\right)^{-1} \left(v_1 + \left(\frac{L - L\alpha_1}{L\alpha_1 - \mu}\right) x_1\right) = x_1.$$

If in addition $\mu = 0$, then the momentum term admits a simpler representation

$$(\forall k \geq 1) \quad \frac{\rho_k \alpha_k (1 - \alpha_k)}{\rho_k \alpha_k^2 + \alpha_{k+1}} = \alpha_{k+1}(\alpha_k^{-1} - 1).$$

3.5 R-WAPG unifies existing accelerations scheme

In addition to various equivalent forms of the R-WAPG algorithm, the R-WAPG sequences are much more flexible. They generalize many existing sequences used in accelerated proximal gradient schemes.

This section demonstrates that several variants of FISTA in the literatures reduces to the R-WAPG method by setting up the R-WAPG sequences with additional assumptions. This section will also demonstrate that the convergence claim (Proposition 3.6) holds, and it derives convergence rates consistent with results in the literatures. The table below shows the R-WAPG convergence results specialized into various settings.

Algorithm	μ	α_k	ρ_k	Convergence of $F(x_k) - F^*$
R-WAPG in Definition 3.5	$\mu \geq 0$	$\alpha_k \in (\mu/L, 1)$	$\rho_k > 0$	$\mathcal{O}\left(\prod_{i=0}^{k-1} \max(1, \rho_i)(1-\alpha_{i+1})\right)$ (Proposition 3.6)
Chambolle, Dossal 2015 [9]	$\mu = 0$	$0 < \alpha_k^{-2} \leq \alpha_{k+1}^{-1} - \alpha_{k+1}^{-2}$	$\rho_k \geq 1$	$\mathcal{O}(\alpha_k^2)$ (Theorem 3.19)
V-FISTA Beck (10.7.7) [3]	$\mu > 0$	$\alpha_k = \sqrt{\mu/L}$	$\rho_k = 1$	$\mathcal{O}\left((1-\sqrt{\mu/L})^k\right)$, (Theorem 3.21, remark)
R-WAPG in Definition 3.5	$\mu > 0$	$\alpha_k = \alpha \in (\mu/L, 1)$	$\rho_k = \rho > 0$	$\mathcal{O}\left(\max(1-\alpha, 1-\mu/(\alpha L))^k\right)$ (Theorem 3.21)

The lemma follows characterizes momentum sequences in the literatures using Definition 3.4.

Lemma 3.16 (R-WAPG sequences as inverted FISTA sequence) *Let R-WAPG sequence $(\rho_k)_{k\geq 0}, (\alpha_k)_{k\geq 0}$ given by Definition 3.4. If $\mu = 0, \rho_k \geq 1\ \forall k \geq 0$, and $\alpha_0 = 1$, then:*

(i) $\alpha_k^{-2} \geq \alpha_{k+1}^{-2} - \alpha_{k+1}^{-1}\ \forall k \geq 0$

(ii) *Let $t_k := \alpha_k^{-1}$, then $0 < t_{k+1} \leq (1/2)\left(1 + \sqrt{1+4t_k^2}\right)\ \forall k \geq 0$, hence the name: "Inverted FISTA sequence".*

(iii) $\prod_{i=1}^{k} \max(1, \rho_{k-1})(1-\alpha_k) = \alpha_k^2 \quad (\forall k \geq 1)$.

Remark 3.17 The sequence t_k is exactly the same as in Theorem 3.1 of Chambolle, Dossal [9].

Lemma 3.18 (Constant R-WAPG sequences) *Suppose $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ are R-WAPG sequences given by Definition 3.4 and assume $L > \mu > 0$. Define $q := \mu/L$. Then $\forall r \in \left(\sqrt{q}, \sqrt{q^{-1}}\right)$, the constant sequence $\alpha_k := r\sqrt{q}$ has the following:*

(i) *For any $r \in \left(\sqrt{q}, \sqrt{q^{-1}}\right)$, the constant sequence $\alpha_k := \alpha \in (q, 1)$ and $\rho_k := \rho = \left(1 - r^{-1}\sqrt{q}\right)\left(1 - r\sqrt{q}\right)^{-1} > 0$, hence it's a pair of valid R-WAPG sequence.*

(ii) *The momentum terms θ_{t+1} in Definition 3.11, which we denoted by θ is the constant: $\theta = (1 - r^{-1}\sqrt{q})(1 - r\sqrt{q})(1-q)^{-1}$*

(iii) *When $r = 1$, $\theta = (1-\sqrt{q})(1+\sqrt{q})^{-1}$.*

(iv) *For all $r \in \left(1, \sqrt{q^{-1}}\right)$, $\rho > 1$; for all $r \in \left(\sqrt{q}, 1\right]$ $\rho \leq 1$.*

(v) *For all $r \in \left(\sqrt{q}, \sqrt{q^{-1}}\right)$, $\max(\rho, 1)(1-\alpha) = \max\left(1 - r\sqrt{q}, 1 - r^{-1}q\right)$.*

Theorem 3.19 (FISTA first variant Chambolle, Dossal 2015) *Fix arbitrary $a \geq 2$. Define $\forall k \geq 1$ the sequence $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ by*

$$\alpha_k = a/(k+a),$$
$$\rho_k = \frac{(k+a)^2}{(k+1)(k+a+1)}.$$

Consider the algorithm given by:

Initialize any $y_1 = x_1$.
For $k = 1, 2, \ldots,$ update:

$$x_{k+1} := y_k + L^{-1}\mathcal{G}_L(y_k),$$
$$\theta_{t+1} := \alpha_{k+1}(\alpha_k^{-1} - 1),$$
$$y_{k+1} := x_{k+1} + \theta_{k+1}(x_{k+1} - x_k).$$

If $\mu = 0$, then $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ is a valid pair of R-WAPG sequence from Definition 3.4 and the above algorithm is a valid form of R-WAPG.

Assume minimizer x^* exists for function F. Then algorithm produces $(x_k)_{k\geq 0}$ such that $F(x) - F(x^*)$ convergences at a rate of $\mathcal{O}(\alpha_k^2)$.

Remark 3.20 This algorithm described here is exactly the same algorithm being analyzed in the paper by Chambolle, Dossal [9].

Theorem 3.21 (Fixed momentum APG) *Assume $L > \mu > 0$, let a pair of constant R-WAPG sequence: $(\alpha_k)_{k\geq 0}, (\rho_k)_{k\geq 0}$ be given by Lemma 3.18. Define $q := \mu/L$ and for any fixed $r \in \left(\sqrt{q}, \sqrt{q^{-1}}\right)$, let $\alpha_k := \alpha = r\sqrt{q}$ be the constant R-WAPG sequence. Consider the algorithm with a constant momentum specified by the following:*

Define $\theta = \left(1 - r^{-1}\sqrt{q}\right)(1 - r\sqrt{q})(1-q)^{-1}$.
Initialize $y_1 = x_1$; for $k = 1, 2, \ldots, N$, update:

$$x_{k+1} = y_k + L^{-1}\mathcal{G}_L y_k,$$
$$y_{k+1} = x_{k+1} + \theta(x_{k+1} - x_k).$$

Then the algorithm generates $(x_k)_{k\geq 1}$ such that $F(x) - F(x^*)$ convergences at a rate of $\mathcal{O}\left(\max(1 - r\sqrt{q}, 1 - r^{-1}\sqrt{q})^k\right)$.

Remark 3.22 When $r = 1$, the algorithm described above is exactly the same as the V-FISTA algorithm specified in (10.7.7) of Beck's book [3].

4 The method of Free R-WAPG

This section introduces an algorithm of our creation inspired by the remark of Proposition 3.3. Algorithm 1 estimates the μ constant as the algorithm executes and pools the information using the Bregman Divergence of the smooth part function f.

Algorithm 1 Free R-WAPG

1: **Input:** $f, g, x_0, L > \mu \geq 0, \in \mathbb{R}^n, N \in \mathbb{N}$
2: **Initialize:** $y_0 := x_0; L := 1; \mu := 1/2; \alpha_0 = 1;$
3: **Compute:** $f(y_k);$
4: **for** $k = 0, 1, 2, \cdots, N$ **do**
5: **Compute:** $\nabla f(y_k); x^+ := [I + L^{-1}\partial g](y_k - L^{-1}\nabla f(y_k));$
6: **while** $L/2\|x^+ - y\|^2 < D_f(x^+, y)$ **do**
7: $L := 2L;$
8: $x^+ = [I + L^{-1}\partial g](y_k - L^{-1}\nabla f(y_k));$
9: **end while**
10: $x_{k+1} := x^+;$
11: $\alpha_{k+1} := (1/2)\left(\mu/L - \alpha_k^2 + \sqrt{(\mu/L - \alpha_k^2)^2 + 4\alpha_k^2}\right);$
12: $\theta_{k+1} := \alpha_k(1 - \alpha_k)/(\alpha_k^2 + \alpha_{k+1});$
13: $y_{k+1} := x_{k+1} + \theta_{k+1}(x_{k+1} - x_k);$
14: **Compute:** $f(y_{k+1})$
15: $\mu := (1/2)(2D_f(y_{k+1}, y_k)/\|y_{k+1} - y_k\|^2) + (1/2)\mu;$
16: **end for**

Line 5-8 estimates upper bound for the Lipschitz constant and find x^+, the next iterates produced by proximal gradient descent on previous y_k; Line 9 updates x_{k+1} to be x^+, a successful iterate identified by the Lipschitz line search routine; Line 10 updates the R-WAPG sequence α_k for the iterates y_{k+1}; Line 13 updates μ using the Bregman Divergence of f from iterates y_{k+1}, y_k.

Assume L given is an upper bound of the Lipschitz smoothness constant of f, then the algorithm calls $f(\cdot)$ two times, and $\nabla f(\cdot)$ once per iteration. The algorithm computes $\nabla f(y_k)$ once for x^+, $f(y_{k+1})$ once for Bregman Divergence because $f(y_k)$ is evaluated from the previous iteration, and $f(x^+)$ once for Lipschitz constant line search condition. We note that $f(y_0)$ is computed before the start of the for loop. And finally, it evaluates proximal of g at $y_k - L^{-1}\nabla f(y_k)$ once.

4.1 Numerical experiments

This section showcases results for the numerical experiments conducted using FR-WAPG algorithm (Algorithm 1), and compare with other APG algorithms in the literatures: the V-

FISTA, and M-FISTA algorithm described in Section (10.7.7, 10.7.6) by Beck [3]. We implemented in Julia [6] and FR-WAPG (Algorithm 1) with V-FISTA, M-FISTA. The equivalences highlighted in Proposition 3.15 allows us to compare the sequence of iterates $(x_k)_{k\geq 1}, (y_k)_{k\geq 0}$ for R-WAPG, VISTA and M-FISTA.

We measure the aggregate statistics of the base two logarithms of the normalized optimality gap (NOG), at each iteration k with the same initialization conditions given for all algorithms. Given x_k, and minimum F^*, we define the normalized optimality gap:

$$\delta_k := \log_2 \left(\mathbf{NOG}_k := \frac{F(x_k) - F^*}{F(x_0) - F^*} \right).$$

Since it's not the case that F^* is always known in prior, we will the minimum of all $F(x_k)$ across all algorithms, all iterations k as the surrogate for F^* when the true value is unavailable.

We consider the norm of the gradient mapping $\|\mathcal{G}_L(y_k)\| < \epsilon$ as a termination conditions for all test algorithms. The L can change during each iteration if it's obtained through the specified Lipschitz line search routine.

4.1.1 Simple convex quadratic

Consider $\min_{x\in\mathbb{R}^n}\{F(x) := f(x) + 0\}$ where

$$f(x) = (1/2)\langle x, Ax\rangle.$$

A is set to be a positive semi-definite diagonal matrix, so the problem admits unique minimizer $x^* = \mathbf{0}$ with minimum being zero. We apply Algorithm 1, M-FISTA, and V-FISTA. The following parameters are used to set up the problem:

(i) N, the dimension of the problem which defines $A \in \mathbb{R}^{N\times N}$, a diagonal matrix given by $N-1$ linearly spaced with equal increment on the interval $[\mu, L]$, and an extra number 0, i.e: $A = \mathrm{diag}(0, \mu+(L-\mu)(N-1)^{-1}, \mu+2(L-\mu)(N-1)^{-1}, \cdots, \mu+(N-2)(L-\mu)^{-1}, L)$.

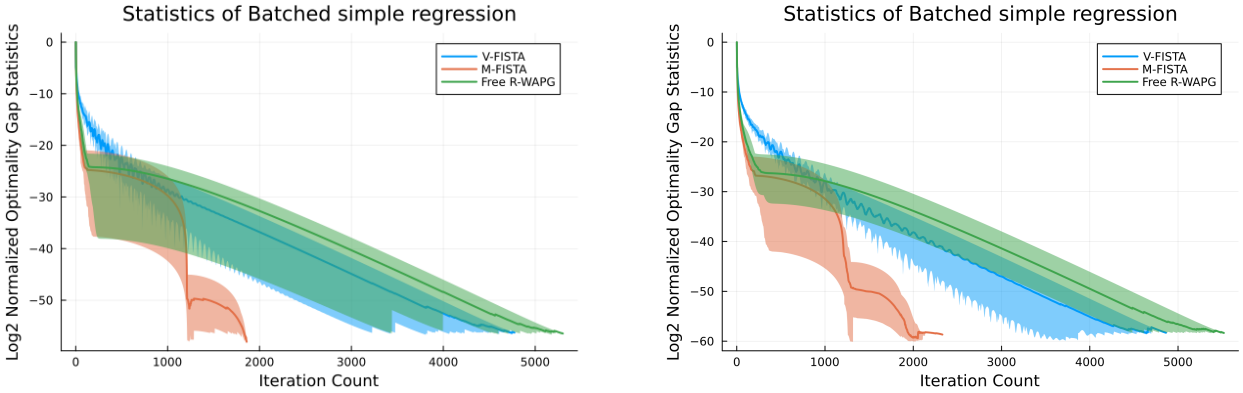
(ii) The strong convexity and Lipschitz smoothness constant has $0 < \mu < L$. They are given in prior to construct the problem.

(iii) $\epsilon > 0$ is the tolerance value, and it's set to be 10^{-10}.

(iv) $x_0 \sim \mathcal{N}(I, \mathbf{0})$ is a vector, and it's the initial condition for all the algorithm. In this case the initial guess is fixed for all R-WAPG, M-FISTA and M-FISTA, but it's randomly generated by the zero mean standard normal distribution for each element in the vector.

The parameter $L = 1, \mu = 10^{-5}$ are given in prior to produce the diagonal matrix A, and we conduct many experiments for $N = 256$ and $N = 1024$. For all R-WAPG, M-FISTA and V-FISTA the same random initial guess is used for all test algorithms and 30 experiments are repeated with a different random initial guess each time. The maximum, minimum and median values of δ_k are measured for all algorithms at each iteration and plotted as a ribbon. Results are shown in Figure 1. The solid line in the ribbon is the median value of δ_k across all experiment, the ribbon gives the maximum, minimum value of δ_k for each iteration across all experiments. FR-WAPG initially behaves similar to M-FISTA, but as the iteration goes on, it started to behave like V-FISTA.

The most surprising feature here is the monotone descent, however, it's being numerical verified that the method is not monotone in general, it just looks monotone on the figure.



(a) $N = 256$, simple convex quadratic.

(b) $N = 1024$, simple convex quadratic.

Figure 1: Simple convex quadratic experiments results for V-FISTA, M-FISTA, and R-WAPG.

Another quantity that maybe interesting other than δ_k would be the estimated value of μ during at each iteration k. One individual experiment is carried out for the R-WAPG algorithm and the value of μ at each iteration is being recorded as well. Figure 2 showcases the results. In this experiment, the values oscillate and converges to the true μ value. Observe that the iteration when the estimates are nearing the true value corresponds to the iteration when the algorithm plateau away from its initial fast descent.

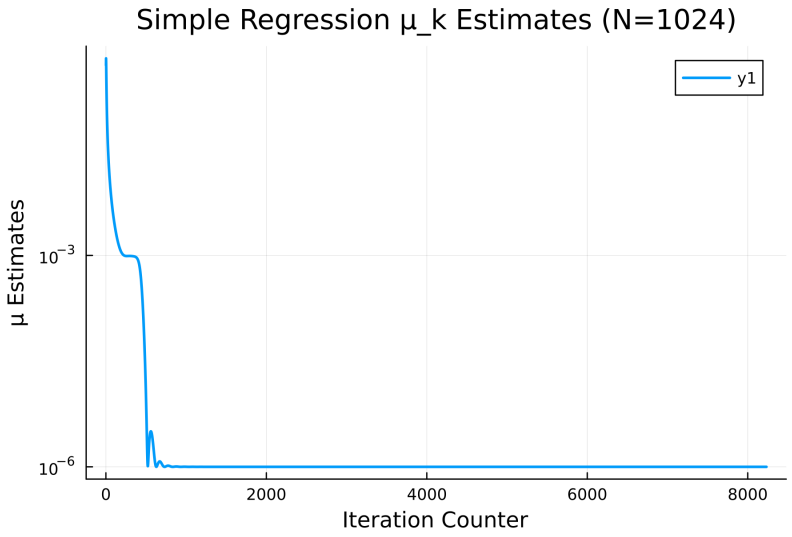


Figure 2: $N = 1024$, the μ estimates produced by Algorithm 1 (R-WAPG) is recorded.

4.1.2 LASSO

This section present results of numerical experiment for solving the (least absolute shrinkage and selection operator) LASSO problem proposed by Tibshirani [41]. The problem of LASSO has smooth, nonsmooth additive parts given by:

$$\min_{x\in\mathbb{R}^n}\left\{\frac{1}{2}\|Ax-b\|^2+\lambda\|x\|_1\right\}.$$

The smooth part is $f(x)=\frac{1}{2}\|Ax-b\|^2$ and the nonsmooth is $g(x)=\lambda\|x\|_1$. The objective function is coersive and the exact minimum, or minimizers are unknown. We perform numerical experiments using V-FISTA, M-FISTA and FR-WAPG on this problem. The setup of our parameters now follow.

(i) M, N are constants. They define matrix $A \in \mathbb{R}^{M\times N}$ which has entries of i.i.d random variable taken from a standard normal distribution.

(ii) L, μ, are Lipschitz constant and the strong convexity constant for the smooth part of the objective which are not known prior. Hence, they are estimated by A by $\mu = 1/\|(A^TA)^{-1}\|$ and $L = \|A^TA\|$ prior to experiment to assist the V-FISTA algorithm.

(iii) $x^+ = [1\ -1\ 1\ \cdots]^T \in \mathbb{R}^N$, it's a vector with alternating $1, -1$ in it.

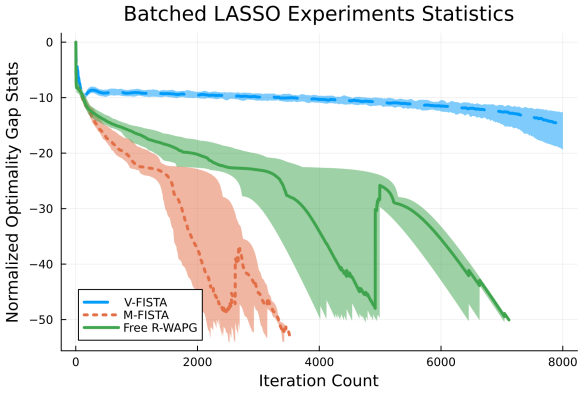
(iv) Given x^+, it has $b = Ax^+ \in \mathbb{R}^M$.

(v) Given A, estimations for L, μ are given by $L = \|A^T A\|$, $\mu = \|(A^T A)^{-1}\|^{-1}$.

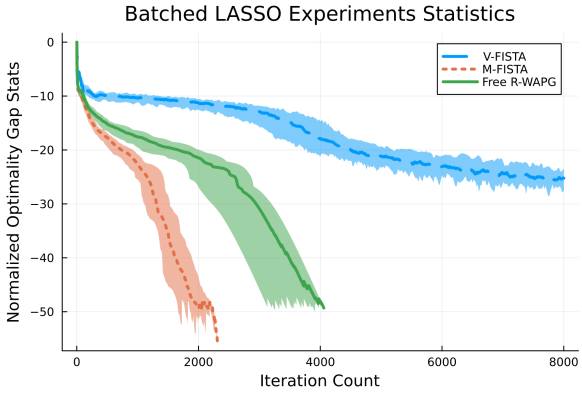
(vi) $x_0 \in \mathbb{R}^N$ denotes the initial guess consist of i.i.d random variable realized from the standard normal distribution.

(vii) $\epsilon = 10^{-6} > 0$ sets the tolerance for all test algorithm on $\|\mathcal{G}_L(x_k)\|$.

Experiments were conducted using V-FISTA, M-FISTA and FR-WAPG with $(M, N) = (64, 256)$ and $(M, N) = (64, 128)$ respectively. A random matrix A is realized and is fixed and the for all test algorithms and all repetitions. The same experiment are repeated 30 times on all algorithms with a fixed random initial condition x_k realized from the distribution. The aggregate statistics of δ_k are collected for all repetitions, and then grouped by the respective algorithm. The results are showcased in Figure 3. The bump on the curve is due to a subset of test instances of the 30 repetition where the algorithms take larger number of iterations to terminate.



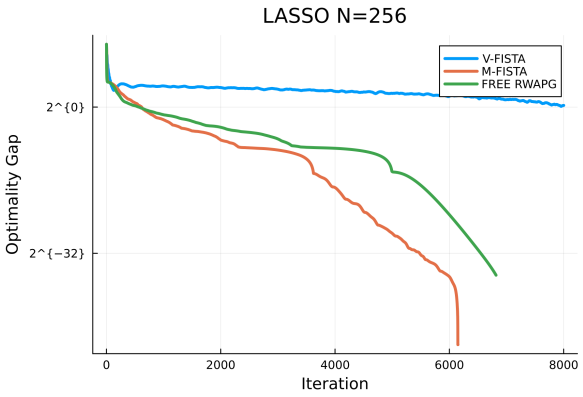
(a) LASSO experiment with $M = 64, N = 256$. Plots of minimum, maximum, and median δ_k with estimated F^*.



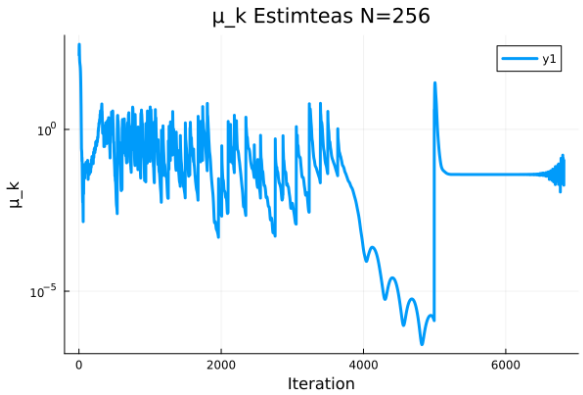
(b) LASSO experiment with $M = 64, N = 128$. Plots of minimum, maximum, and median δ_k with estimated F^*.

Figure 3: LASSO experiments.

Another quantity of interest is the estimates of μ on each iteration of the algorithm. A single experiment were conducted and the estimates and δ_k are in Figure 4



(a) Single lasso experiment plot of δ_k with.



(b) The μ estimated by test algorithms for one LASSO experiment.

Figure 4: A single LASSO experiment results, with $M = 64, 256$.

For this specific experiment showed in the figure, the estimated value of μ, L which we feed into V-FISTA are $\mu = 7.432363627613958 \times 10^{-18}$ and $L = 2321.737206983643$. One of the most important feature is that the estimate μ doesn't converge to the true value, but it didn't affect the convergence of δ_k.

4.2 Future works for R-WAPG

The R-WAPG framework neglected a detail in the literatures. Our future works would extend the frameworks and include this detail.

4.2.1 Nesterov's idea of strong convexity transfer

We consider the case that $F = f + g$ where, f is L Lipschitz smooth, $\mu_f \geq 0$ strongly convex and g is closed convex. In Nesterov's 2013 paper [30], he considers accelerated minimization problem $\phi = f + \Psi$ with f L_f smooth and Ψ being $\mu_\Psi \geq 0$ strongly convex.

This detail on itself is not necessarily interesting, since without lost of generality, we can always do the splitting $f + \mu_\Psi/2\|\cdot\|^2$ and $\Psi - \mu_\Psi/2\|\cdot\|^2$ instead so that the smooth part is $\mu_\Psi \geq 0$ strongly convex. It's an interesting detail we should consider because:

(i) A strongly convex nonsmooth parts still gives linear convergence by Theorem 6 in Nesterov's 2013 [30].

(ii) The strong convexity constant can be easily estimated via a routine specified as (5.14) in Nesterov 2013 [30] and the complexity of the estimation is bounded precisely.

(i) indicates that our theories of R-WAPG is incomplete, since it doesn't predict linear convergence when g is strongly convex. (ii) indicates that splitting the strongly convex objective so that it's with the proximal operator gives computational advantage.

Furthermore, Nesterov was not the only person who thought about it. In Algorithm 5, Chambolle, Pock [10] captures several variants of FISTA, and it assumes that $F = f + g$ where f, g has strong convexity constant $\mu_f \geq 0, \mu_g \geq 0$ respective so F is $\mu := \mu_f + \mu_g \geq 0$ strongly convex.

Upon first inspection and preliminary readings, we lay out the followings that may contribute to include strong convexity in the non-smooth part into our R-WAPG framework. The R-WAPG sequence α_k, ρ_k will have to describe the sequence in Chambolle, Pock [10] Algorithm 5 which has

$$q\alpha_{k+1}^2 = (1 - \alpha_{k+1})q\alpha_k^2 + q\alpha_{k+1}^2.$$

Where $q = L^{-1}(\mu_f + \mu_g)/(1 + L^{-1}\mu_g)$. Additionally, observe that in the special case where $f \equiv 0$, then proximal gradient becomes proximal point on the strongly convex objective $\mu_g \geq 0$ alone and its convergence rate is included in the outer loop convergence analysis in the Catalyst Acceleration Framework by Lin et al. [24], this provided us with a hint on an estimating sequence approach. Of course, Lemma 2.14 requires a complete rework to include $\mu_g \geq 0$. The recursive quantity R_k would also require a complete rework because we invented the quantity R_k just so the proof of Proposition 3.3 falls through. The quantity is obtained from the Nesterov's estimating sequence of APG, to include such a change, the estimating sequence required to engineer R_k, which is the recursive quantity will also require rework. The only part that may not require rework is the Definition 3.2 because the R-WAPG sequence is generic. We are quite certain about these required changes to achieve the open question because we invented R-WAPG.

5 Catalyst accelerations

We introduce assumptions and notations for the Catalyst Meta Acceleration Frameworks.

Assumption 5.1 Given any $\beta > 0$ and $y \in \mathbb{R}^n$, and $F : \mathbb{R}^n \to \overline{\mathbb{R}}$ is $\mu \geq 0$ strongly convex and closed. Assume that minimizer exists for F and the minimum is F^*. For all $x, y \in \mathbb{R}^n, \beta > 0$ define the model function:

$$\mathcal{M}_F^{\beta^{-1}}(x; y) := F(x) + \frac{\beta}{2}\|x - y\|^2.$$

We define the Moreau Envelope at $y \in \mathbb{R}^n$ to be $\mathcal{M}^*_{F,\beta^{-1}}(y) := \min_{x\in\mathbb{R}^n} \mathcal{M}_F^{\beta^{-1}}(x;y)$ and the resolvent operator for subgradient of F to be $\mathcal{J}_{\beta^{-1}F}$, i.e: which is also called the proximal operator.

We invent the notations because there are no consistent notations in the literatures and several of the Catalyst papers all use different notations.

Definition 5.2 (Absolute termination criterion C1) *Take F as given by Assumption 5.1. For $\epsilon > 0, \kappa > 0$ and $x \in \mathbb{R}^n$, the absolute criterion C1 characterizes the set of inexact proximal iterates by:*

$$\mathcal{J}^{\epsilon}_{\kappa^{-1}F}(x) := \left\{ y \in \mathbb{R}^n \;\middle|\; \mathcal{M}_F^{1/\kappa}(y;x) - \mathcal{M}^*_{F,1/\kappa}(x) \leq \epsilon \right\}.$$

Remark 5.3 Setting $\epsilon = 0$, we have the exact definition of the exact resolvent given as $\mathcal{J}_{\beta^{-1}F}y = \mathcal{J}^0_{\beta^{-1}F}y$.

The remainder of this section is organized as the follow. Section 5.1 gives the main ideas behind the Catalyst Acceleration Framework. It follows Lin et al.'s first paper [24] on Catalyst Acceleration and summarizes the convergence analysis of the algorithm. Section 5.2 introduces Lin et al.'s second paper [25] on Catalyst which considers the improved relative error condition stated in Definition 5.15. The relative error condition improves the total complexity and inspired nonconvex gradient based 4WD Catalyst by Paquette et al. [34].

5.1 Introduction to Catalyst

Inspired by Accelerated Proximal Point Method (APPM) from Güler [20], and Inexact Proximal Point Method (IPPM) of Rockafellar 1976 [36], Lin et al. [24] proposed a generic method taking inspirations from the convergence claims of APPM to accelerate Variance Reduced Method (VRM). VRMs are incremental method that is not slower than full gradient descent in complexity. See Gower's guide [18] for more information on variance reduced methods in machine learning.

In brief, VRM is a type of incremental methods for solving a large sum problem: $F(x) = \sum_{i=1}^N f_i(x)$ in machine learning. See Bertsekas's surveys [4, 5] for more context. VRM can be deterministic, or stochastic. When it's stochastic, the theories focus on the expected optimality gap: $\mathbb{E}[F(x_k) - F^*]$. Let's assume for simplicity of discussion that it's deterministic, so from now on we focus on: $F(x_k) - F^*$ instead.

VRM stabilizes the estimate of gradient using information from all or a subset of gradients evaluated at previous iterates. In each iteration, it accesses the gradient of a few samples to attain a new estimate of the gradient with minimum additional calculations. This gives better

complexity than full gradient descent over all. Compare to traditional stochastic gradient, VRM produces smaller variance of the estimated gradient near the minimizer which gives faster convergence rate. Major examples of VRMs include SVRG by Xiao, Zhang [42], Finito by Defazio et al. [12], SAG by Schmidt et al. [39], and SAGA by [11].

The following definition introduces the first Catalyst algorithm in [24].

Definition 5.4 (Lin's Universal Catalyst Acceleration)
Let $F : \mathbb{R} \to \overline{\mathbb{R}}$ be $\mu \geq 0$ strongly convex and closed, $x_0 \in \mathbb{R}^n$ be the initial condition and $\kappa > 0$, $\alpha_0 \in (0, 1]$ are fixed parameters for the algorithm. Let $(\epsilon_k)_{k\geq 0}$ be an absolute error sequence chosen for the evaluation for inexact proximal point method.

Initialize $x_0 = y_0$. Then the algorithm generates $(x_k, y_k)_{k\geq 0}$ for all $k \geq 1$ by the procedures:

$$\begin{aligned}
&\text{find } x_k \in \mathcal{J}^{\epsilon_k}_{\kappa^{-1}F} y_{k-1},\\
&\text{find } \alpha_k \in (0,1) \text{ such that } \alpha_k^2 = (1-\alpha_k)\alpha_{k-1}^2 + (\mu/(\mu+\kappa))\alpha_k,\\
&y_k = x_k + \frac{\alpha_{k-1}(1-\alpha_{k-1})}{\alpha_{k-1}^2 + \alpha_k}(x_k - x_{k-1}).
\end{aligned}$$

Remark 5.5 The above algorithm is Algorithm 1 from the first paper on Catalyst Acceleration by Lin et al. [24]. The explicit formula for α_k is the larger root of solving the quadratic equation given by:

$$\alpha_k = \frac{1}{2}\left(q - \alpha_{k-1}^2 + \sqrt{(q+\alpha_{k-1})^2 + 4\alpha_{k-1}}\right),$$

where $q = \mu/(k+\mu)$. The choices for κ, ϵ_k will come later.

Notations now follows. With a fixed regularization parameter $\kappa > 0$, the outer loop (i.e: Definition 5.4) produces $(y_k, x_k)_{k\geq 1}$ denote it by $\mathbb{A}$. Typically, an iterative scheme (e.g: VRM) with a known complexity finds inexact proximal iterates $x_k \in \mathcal{J}^{\epsilon_k}_{\kappa^{-1}F}(y_{k-1})$. We refer to it as the inner loop and denote it by $\mathbb{M}$. Without loss of generality assume it generates some iterates $(z_{k,t})_{t\geq 0}$ where k is the corresponding iteration counter of the outer loop and t is the iteration counter of the inner loop.

The choice of absolute error sequence $(\epsilon_k)_{k\geq 0}$ determines the iteration complexity of both $\mathbb{A}, \mathbb{M}$. The overall iteration complexity of Catalyst algorithm counts the total number of inner loop iteration at the k th iteration of outer loop. The term "iteration complexity" refers to the number of iterations required to achieve a desired accuracy, a concept related to the convergence rate of the algorithm. We adopt iteration complexity in the sections that follow. Since F is convex, we focus on the convergence rate of the optimality gap $F(x_k) - F^*$

for $\mathbb{A}$ and convergence of the model function $\mathcal{M}_F^{\kappa^{-1}}(\cdot, y_{k-1})$ for $\mathbb{M}$ given y_{k-1}, ϵ_k and initial guess $z_{k,0}$.

5.1.1 Outer loop iteration complexity

The error sequence $(\epsilon_k)_{k\geq 0}$ controls the convergence rate of $F(x_k) - F^*$ of the outer loop $\mathbb{A}$. Depending on either $\mu > 0$, or $\mu = 0$, the choice of $(\epsilon_k)_{k\geq 0}$ differs. The theorems that follow state the error sequence required for the outer loop to retain optimal convergence rate, they are Theorem 3.3, 3.1 in Lin et al. [24].

Theorem 5.6 (Outer loop convergence strongly convex) *For $\mathbb{A}$ with regularization parameter $\kappa > 0$. Assume that F is $\mu > 0$ strongly convex. Choose $\alpha_0 = \sqrt{q}$ with $q = \mu/(\kappa + \mu)$ and the error sequence*

$$\epsilon_k = \frac{2}{9}(F(x_0) - F^*)(1-\rho)^k \quad \textit{with} \quad \rho < \sqrt{q}.$$

Then the $\mathbb{A}$ generates $(x_k)_{k\geq 0}$ such that

$$F(x_k) - F^* \leq C(1-\rho)^{k+1}(F(x_0) - F^*) \quad \textit{with} \quad C = \frac{8}{(\sqrt{q} - \rho)^2}.$$

Remark 5.7 Suggested by Lin et al., ρ is at the discretion of the practitioner, take for example $\rho = 0.9\sqrt{q}$ would work.

Theorem 5.8 (Outer loop convergence not necessarily strongly convex) *For $\mathbb{A}$ with regularization parameter $\kappa > 0$. Assume that F is convex but with strong convexity constant $\mu \geq 0$. Choose $\alpha_0 = (\sqrt{5} - 1)/2$ and the error sequence*

$$\epsilon_k = \frac{2(F(x_0) - F^*)}{9(k+2)^{4+\eta}} \quad \textit{with} \quad \eta > 0.$$

Take x^* to be a minimizer of F. Then algorithm $\mathbb{A}$ generates $(x_k)_{k\geq 0}$ such that it has a convergence rate of

$$F(x_k) - F^* \leq \frac{8}{(k+2)^2}\left(\left(1 + \frac{2}{\eta}\right)^2 (F(x_k) - F^*) + \frac{\kappa}{2}\|x_0 - x^*\|^2\right).$$

Remark 5.9 Suggested by Lin et al., $\eta > 0$ is at the discretion of the practitioners, for an example, $\eta = 0.1$ would work.

Theorem 5.6, 5.8 gives the largest possible absolute error sequence $(\epsilon_k)_{k\geq 0}$ such that the convergence claim holds. Of course, the results still hold if we take smaller values of $(\epsilon_k)_{k\geq 0}$ as suggested by the above theorems, but it will increase the iteration complexity of $\mathbb{M}$.

Observe the absolute error sequence $(\epsilon_k)_{k\geq 0}$ requires prior knowledge of F^*. It's poses no theoretical concerns, but it's of upmost practical concern since F^* is not accessible in practice prior to executing the algorithm. In the work by Lin et al. [24], the example algorithm given is in appendix item D.3 called Accelerated MISO Prox. It automatically builds a lower bound estimates on F^* in the outer loop as the algorithm executes.

An inexact version of the proximal gradient inequality (similar to Theorem 2.14) stated as Lemma A.7 in [24] derives the convergence of $\mathbb{A}$. It is instrumental for formulating an inexact variant of the estimating sequence $\phi_k^* \geq F(x_k) + \xi_k$. The convergence proof (outer and inner loop together) from Lin was inspired by Schmidt's Inexact Proximal Gradient method [40]. The technique of estimating sequence introduced back in Definition 2.15 did the heavy lifting, but it results in depressingly long proof making it unsuitable for exposition here. Significant pieces of theoretical innovations using Nesterov's estimating sequence are covered in details in our most recent Fall Winter 2024 MATH 590 report. The parts that come will complement the report to focus on the big picture of Catalyst Acceleration.

5.1.2 Inner loop complexity

The iteration complexity of $\mathbb{M}$ relates to the outer loop when warm start is used. The Catalyst Paper [24] suggested the use of $z_{k,0} = x_{k-1}$ as the warm start condition. With Assumption 5.10, and the suggested warm start condition, Lin et al. derived the upper bound of the iteration complexity of $\mathbb{M}$. These results are stated as Proposition 3.2, 3.3 of their text. Here, 5.14, 5.12 restates them in our notations. These two theorems relate the convergence of $\mathbb{M}, \mathbb{A}$ and gives a convergence rate of $F(x_k) - F^*$ expressed using the total number of iteration underwent by $\mathbb{M}$.

Assumption 5.10 (Linear convergence of inner loop) For any $k \in \mathbb{N}$, $y \in \mathbb{R}^n$. Suppose $\mathbb{M}$ generates iterates $(z_{k,t})_{t\geq 0}$ for the inner loop iteration such that there exists $C_{\mathbb{M}} > 0$, and it has:

$$\mathcal{M}_F^{\kappa^{-1}}(z_{k,t}, y) - \mathcal{M}_{F,\kappa^{-1}}^*(y) \leq C_{\mathbb{M}}(1 - \tau_{\mathbb{M}})^t \left(\mathcal{M}_F^{\kappa^{-1}}(z_{k,0}) - \mathcal{M}_{F,\kappa^{-1}}^*(y)\right).$$

Remark 5.11 VRMs listed earlier satisfy this assumption. Details about the consequence of this assumption is in Section 5.3.

Proposition 5.12 (Inner loop complexity strongly convex) *Under the same settings of Theorem 5.6, suppose that*

(i) $\mathbb{M}$ *has linear convergence rate as specified in Assumption 5.10,*

(ii) $\mathbb{M}$ *is initialized with* $z_{k,0} = x_{k-1}$ *for all* $k \geq 2$.

Then, the precision ϵ_k *is achieved within at most a number of iteration* $T_{\mathbb{M}} \leq \widetilde{\mathcal{O}}(1/\tau_{\mathbb{M}})$. *Here* $\widetilde{\mathcal{O}}$ *hides logarithmic complexity in* μ, κ *and other constants.*

Remark 5.13 Here, $T_{\mathbb{M}}$ is a constant associated with just $\mathbb{M}$, it's not related to k, ϵ_k.

Proposition 5.14 (Inner loop convergence not necessarily strongly convex) *Under the settings of Theorem 5.8, suppose that:*

(i) $\mathbb{M}$ *has linear convergence rate as specified in Assumption 5.10,*

(ii) *the initial guess for* $\mathbb{M}$ *is* $z_{0,k} = x_{k-1}$,

(iii) F *has bounded level set.*

Then there exists $T_{\mathbb{M}} \leq \widetilde{\mathcal{O}}(1/\tau_{\mathbb{M}})$ *such that for any* $k \geq 1$, *it requires at most* $T_{\mathbb{M}} \log(k+2)$ *iterations for* $\mathbb{M}$ *to achieve accuracy* ϵ_k.

For a proof of Proposition 5.14, 5.12, see Appendix item B1, B2 in Lin et al. [24]. We are now ready to derive the convergence rate measured by the number of total iteration experience by $\mathbb{M}$. Let $k \in \mathbb{N}$ be the total number of iterations experienced by the outer loop $\mathbb{A}$, let m be the total number of iterations underwent by $\mathbb{M}$. If $\mu > 0$, then Proposition 5.12 gives the total number of inner iteration upper bound: $m \leq T_{\mathbb{M}} k$. Substituting $k \geq m/T_{\mathbb{M}}$ into Theorem 5.6, it gives description of convergence rate of the algorithm measured by the total number of iteration experience by $\mathbb{M}$:

$$\begin{aligned} F(x_k) - F^* &\leq \mathcal{O}\left((1-\rho)^k\right) \leq \mathcal{O}\left((1-\rho)^{m/T_{\mathbb{M}}}\right) \leq \mathcal{O}\left((1-\rho/T_{\mathbb{M}})^m\right) \\ &\leq \widetilde{\mathcal{O}}\left(\tau_{\mathbb{M}}\sqrt{\mu}/(\mu+\kappa)\right). \end{aligned}$$

The second inequality on the first line made use of the fact that $1 + x \leq (1 + x/n)^n$ for all $n \geq 1$ and $|x| \leq n$. The optimal value of κ is suggested by choosing the best $\kappa > 0$ that minimizes the above upper bound.

If $\mu = 0$, using Proposition 5.14 the total number of inner loop iteration executed by $\mathbb{M}$ at the k th iteration of $\mathbb{A}$ is bounded via:

$$m \leq \sum_{i-1}^{k} kT_{\mathbb{M}} \log(i+2) \leq kT_{\mathbb{M}} \log(k+2) \leq T_{\mathbb{M}} k(k+2) \leq \mathcal{O}(T_{\mathbb{M}} k^2).$$

Therefore, using Theorem 5.8, the convergence rate as measure by the total number of inner iteration is given by:

$$F(x_k) - F^* \leq \mathcal{O}(k^{-2}) \leq \mathcal{O}\left(m^{-2}T_{\mathbb{M}}\right) \leq \widetilde{\mathcal{O}}\left(m^{-2}\tau_{\mathbb{M}}^{-1}\right).$$

The last inequality is $T_{\mathbb{M}} < \widetilde{\mathcal{O}}(1/\tau_{\mathbb{M}})$ from Proposition 5.14.

5.2 The second Catalyst Acceleration paper

Lin et al.'s second paper on Catalyst Acceleration [25] describes the new option of relative termination criterion (which we stated here in Definition 5.15) for $\mathbb{M}$ to evaluate the inexact solution of $\mathcal{J}_{\kappa^{-1}F}y_k$. The relative termination condition has advantages: it gives easier convergence analysis, it doesn't need knowledge of F^*, and it can generalize to the nonconvex case. In addition, they also improved warm start strategies for $\mathbb{M}$ at the end of Section 3 for absolute termination criterion in Definition 5.2. Both these ideas together provides a tighter and easier complexity analysis for the Catalyst Acceleration.

In this section, we summarize results regarding criterion C2 in Lin et al.'s second Catalyst paper [25]. The following definition defines relative termination criteria for $\mathbb{M}$.

Definition 5.15 (Relative termination criterion C2) *Take F as given by Assumption 5.1. Given any $\delta \in (0,1]$, $\kappa > 0$ and $x \in \mathbb{R}^n$, the relative criterion C2 of the inexact resolvent is defined by:*

$$\widetilde{\mathcal{J}}^{\delta}_{\kappa^{-1}F}(x) := \left\{ z \in \mathbb{R}^n \;\middle|\; \mathcal{M}^{\kappa^{-1}}_{F}(z;x) - \mathcal{M}^{*}_{F,\kappa^{-1}}(z;x) \leq \frac{\kappa\delta}{2}\|x-z\|^2 \right\}.$$

Remark 5.16 Observe that, if we set $\epsilon = \delta\kappa/2\|x-z\|^2$ then $\widetilde{\mathcal{J}}^{\delta}_{\kappa^{-1}F}(x) = \mathcal{J}^{\epsilon}_{\kappa^{-1}F}(x)$. The relative inexact condition can be interpreted as an adaptive inexactness condition. Observe that $\mathcal{J}_{\kappa^{-1}F}(x) \subseteq \widetilde{\mathcal{J}}^{\delta}_{\kappa^{-1}F}(x)$ hence the set is nonempty.

Stated by Lin, the following lemma provides the sufficient condition for evaluating inexact proximal point up to accuracy ϵ.

Lemma 5.17 (Sufficient condition for C1) *Consider smooth plus nonsmooth objective $F := f + g$ with $g : \mathbb{R}^n \to \overline{\mathbb{R}}, F : \mathbb{R}^n \to \mathbb{R}$ closed and convex, and F is $\mu \geq 0$ strongly convex and L-Lipschitz smooth. With arbitrary $x \in \mathbb{R}^n$ fixed, the model function is additive composite of the form:*

$$\mathcal{M}^{\kappa^{-1}}_{F}(z;x) := \underbrace{f(z) + \frac{\kappa}{2}\|z-x\|^2}_{=:f_\kappa(z)} + g(z).$$

For any $z \in \mathbb{R}^n$, define proximal gradient point

$$\bar{z} = \text{prox}_{\eta g}(z - \nabla f_\kappa(z)) \quad \textit{with} \quad \eta = 1/(\kappa + L).$$

Then it has

$$\frac{1}{\eta}\|z - \bar{z}\| \leq \sqrt{2\kappa\epsilon} \implies \bar{z} \in \mathcal{J}^{\epsilon}_{\kappa^{-1}F}(x).$$

Remark 5.18 This is Lemma 2 in Lin et al. [25].

5.2.1 Consequences of the inner loop termination criteria

Lemma 5.17 describes a sufficient conditions to verify the membership of $\bar{z} \in \mathcal{J}^{\epsilon}_{\kappa^{-1}F}(x)$ through the proximal gradient operator on $F := f_\kappa(z) + g(z)$ which has practical importance because it translates criterion C1 into an implementable criterion in the algorithm with the proximal gradient operator. It contributes to theoretical analysis because it bounds on the true error on the gradient of Moreau Envelope at the point x.

The following results are proved in Lin et al. second Catalyst paper [25] in Lemma 2 and the discussions that follow it. Given any $\epsilon > 0$ if $z \in \mathcal{J}^{\epsilon}_{\kappa^{-1}F}(x)$, define the absolute inexact gradient mapping $\mathcal{G}^{\epsilon}_{\kappa^{-1}F}(z) := \kappa(x - z)$. Then it satsifies:

$$\|z - \mathcal{J}_{\kappa^{-1}F}(x)\| \leq \sqrt{\frac{2\epsilon}{\kappa}} \iff \left\|\mathcal{G}^{\epsilon}_{\kappa^{-1}F}(z) - \nabla \mathcal{M}^{*}_{F,\kappa^{-1}}(x)\right\| \leq \sqrt{2\kappa\epsilon}.$$

The above says that the gradient mapping $\mathcal{G}^{\epsilon}_{\kappa^{-1}F}(z)$ approximates the gradient of Moreau Envelope of F at x. Similarly, if $z \in \widetilde{\mathcal{J}}^{\delta}_{\kappa^{-1}F}(x)$, define the relative inexact gradient mapping $\widetilde{\mathcal{G}}^{\delta}_{\kappa^{-1}F}(x) := \kappa(x - z)$, we have:

$$\begin{aligned} \|z - \mathcal{J}_{\kappa^{-1}F}(x)\| &\leq \sqrt{\delta}\|x - z\| \leq \sqrt{\delta}(\|x - \mathcal{J}_{\kappa^{-1}F}(x)\| - \|z - \mathcal{J}_{\kappa^{-1}F}(x)\|), \\ \left\|\widetilde{\mathcal{G}}^{\delta}_{\kappa^{-1}F}(x) - \nabla \mathcal{M}^{*}_{F,\kappa^{-1}}(x)\right\| &\leq \delta'\|\nabla \mathcal{M}^{*}_{F,\kappa^{-1}}(x)\| \quad \text{with } \delta' = \sqrt{\delta}/\left(1 - \sqrt{\delta}\right). \end{aligned}$$

These results are instrumental to proving the convergence of $\mathbb{M}$ and giving convergence claim of $\mathbb{A}$ under certain assumption on $(\delta_k)_{k\geq 0}, (\epsilon_k)_{k\geq 0}$. For a more general, rigorous and comprehensive development of the characterizations on inexact oracles, see Devolder et al. [13]. It's discusses a Nesterov accelerated algorithm using estimating sequence with inexact oracles.

Besides Definition 5.2, 5.15, the "Fixed Budget" termination criterion terminates $\mathbb{M}$ after a fixed number of iteration given termination criterion C1, C2. Unfortunately the bound is often too loose making it impractical.

Interestingly, the new termination criterion C2 improves the complexity of inner loop $\mathbb{M}$ and outer loop $\mathbb{A}$, and it gives a tighter bound with fewer assumptions. The theorems and commentaries that follow will illustrate.

Definition 5.19 (Catalyst Acceleration all in one) *Suppose that F is a $\mu \geq 0$ strongly smooth under Assumption 5.1. Initialize any $x_0 \in \mathbb{R}^n$, $\kappa > 0$. Given the relative error sequence $(\delta_k)_{k\geq 0}$, or equivalently some fixed budget number of iterations $T_{\mathbb{M}}$, or absolute error sequence $(\epsilon_k)_{k\geq 0}$:*

Initialize $y_0 = x_0, q = \mu/(\kappa+\mu), k = 1$, and if $\mu > 0$, set $\alpha_0 = \sqrt{q}$ otherwise $\alpha_0 = 1$.
***While** desirable accuracy is not reached, **do**:*

(i) *Finds $x_k \approx \mathcal{J}_{\kappa^{-1}F} y_{k-1}$ using warm start.*

(ii) *Pick one of the following fixed termination criterion:*

(a) *Determine the number of fixed budget depending on $T_{\mathbb{M}}$, terminate the above subroutine if fixed budget reached.*

(b) *C1 are satisfied through ϵ_k.*

(c) *C2 are satisfied through δ_k.*

(iii) *Find $\alpha_k \in (0,1)$ such that $\alpha_k^2 = (1-\alpha_k)\alpha_{k-1}^2 + q\alpha_k$.*

(iv) *Compute extrapolation $y_k = x_k + \beta_k(x_k - x_{k-1})$ with*

$$\beta_k = \alpha_{k-1}(1-\alpha_{k-1})/(\alpha_{k-1}^2 + \alpha_k).$$

(v) *Increment $k := k+1$*

End while

Assumption 5.20 (Warm start conditions for the inner loop using C2)
For minimization sub problem $\mathcal{M}_F^{\kappa^{-1}}(z; y_k)$ solved by $\mathbb{M}$, we use the following warm start condition for $z_{k,0}$ under termination criterion C2 (Definition 5.15):

(i) If F is smooth then choose $z_k = y_k$.

(ii) Else it's composite, define $f_k(x) = f(x) + \frac{\kappa}{2}\|x - y_k\|^2$ and do

$$z_{k,0} = \text{prox}_{\eta g}\left(y_k - \eta \nabla f_\kappa(y_k)\right) \quad \text{with } \eta = 1/(\kappa + L).$$

Remark 5.21 See pg 13 of Lin et al. for more detail about warm starting strategies of $\mathbb{M}$ under Termination criterion C1.

Theorem 5.22 (Outer loop complexity under criterion C2) *For the iterates $(x_k)_{k\geq 0}$ generated by algorithm in Definition 5.19, we have*

(i) *If $\mu > 0$, choose $\alpha_0 = \sqrt{q}$, $\delta_k = \sqrt{q}/(2-\sqrt{q})$. Then the iterates $(x_k)_{k\geq 0}$ satisfies $F(x_k) - F^* \leq \mathcal{O}\left(1 - \sqrt{q}/2\right)^k$.*

(ii) *If $\mu = 0$, choose $\alpha_0 = 1$, $\delta_k = 1/(k+1)^2$ satisfies $F(x_k) - F^* \leq \mathcal{O}(k^{-2})$.*

Remark 5.23 This is Proposition 8, 9 in Lin et al.'s second Catalyst paper [25]. For a precise description of the upper bound, see Theorem 8.

Theorem 5.24 (Inner loop complexity with relative error C2)
Consider $\mathbb{M}$ under the settings of Assumption 5.10 and Assumption 5.20, let the relative error sequence $(\delta_k)_{k\geq 0}$ be given by Theorem 5.22, and let $\mathbb{A}$ be defined as in Definition 5.19. Then the sequence $(z_{k,t})_{t\geq 0}$ generated by $\mathbb{M}$ has $T_{k+1} = \min\left\{t \geq 0 : z_{k,t} \in \widetilde{\mathcal{J}}^{\delta}_{\kappa^{-1}F} y_k\right\}$ satisfies:

$$T_{k+1} \leq \tau_{\mathbb{M}}^{-1} \log\left(4C_{\mathbb{M}} \frac{L+\kappa}{\kappa} \frac{2-\sqrt{q}}{q}\right) \quad \text{when} \quad \mu > 0,$$
$$T_{k+1} \leq \tau_{\mathbb{M}}^{-1} \log\left(4C_{\mathbb{M}} \frac{L+\kappa}{\kappa} (k+2)^2\right) \quad \text{when} \quad \mu = 0.$$

Remark 5.25 This is Corollary 16 in Lin et al. [25] but phrased in our notations.

Using Theorem 5.24, 5.24, it's possible to derive the total complexity of the algorithm as counted by the total number of iterations of $\mathbb{M}$ at the k iteration of $\mathbb{A}$. Stated in Proposition 17, 18 in Lin et al. [25] is the following results. The total number of iteration $N_{\mathbb{M}}$ experienced by $\mathbb{M}$ for $\mathbb{A}$ to produce x_k such that $F(x_k) - F^* \leq \epsilon$ when using termination criterion C2, warm start and error sequence strategies in Theorem 5.22 has:

(i) $N_{\mathbb{M}} \leq \widetilde{\mathcal{O}}\left(\tau_{\mathbb{M}}^{-1}\sqrt{\kappa/\epsilon}\log(\epsilon^{-1})\right)$ if $\mu = 0$,

(ii) $N_{\mathbb{N}} \leq \widetilde{O}\left((\tau_{\mathbb{M}}\sqrt{q})^{-1}\log(\epsilon^{-1})\right)$ if $\mu > 0$.

The key improvement here compared to using absolute inexactness for $\mathbb{M}$ as stated by Theorem 5.6, 5.8 is that $(\delta_k)_{k\geq 0}$ doesn't require knowledge on F^*. This innocent detail helps to develop a nonconvex variant of Catalyst call 4WD Catalyst which is the focus of the paper by Paquette [34]. Corollary 16 in Lin et al.'s second Catalyst paper [25] gives complexity for $\mathbb{M}$ Using termination criterion C2 with a warm start specialized for $C2$. It's important to note that the upper complexity bounds are much better, and it doesn't require bounded level set compared to Theorem 5.14.

5.3 Potential future research

Now, we have enough context to understand the potential future directions of research regarding the Catalyst Acceleration framework.

5.3.1 Necoara et al.'s comments on Catalyst Acceleration

Necoara et al. [27] considered optimization problem $f^* = \min_{x \in X} f(x)$ with $f : X \subseteq \mathbb{R}^n \to \mathbb{R}$ and X closed and convex. They extended the linear convergence results of accelerated gradient method such as Nesterov's acceleration with proximal gradient into a wider setting than strong convexity.

As a secondary objective, they introduced the relations between several weaker characterizations of strong convexity on a closed convex set. Definition 1, 2, 3, 4, 5, introduced the ideas of: Quasi-strong convexity, Quadratic under approximation, Quadratic gradient growth, quadratic functional growth (recall that it was stated as a consequence of strong convexity in Lemma 2.9), and Global Error bound conditions.

For our interests, we focus on one of their comments on the Catalyst by Lin et al. [24]. They said, and here we quote:

> It is also worth to investigate the complexity bounds obtained when wrapping linearly-convergent algorithms under the proposed conditions in an outer-loop using a generic acceleration scheme such as Catalyst [24]. This would allow to extend to non-strongly convex settings a wide range of incremental optimization algorithms designed for large finite-sum problems arising in machine learning and statistics.

We note that, their idea of extension can be applied to inner loop $\mathbb{M}$, or out loop $\mathbb{A}$, either, or both.

Finally, it's worth noting that in the years before and after Necoara et al.'s paper and Catalyst, there were parallel developments around the ideas of extending Nesterov's accelerations and Catalyst to the nonconvex settings. Paquette et al. [34] were inspired by [16] and showed that the Catalyst algorithm based on the gradient termination criterion analogous similar to C2 (see equation (13), (14) in their paper) can adapt to nonconvex settings and the norm of the gradient converges. However, it's at the expense of one additional evaluation of the proximal point method, and it requires evaluating the function value once per outer loop iteration. They used an analogous inner loop complexity to Assumption 5.10 which is stated as Equation (16) in their paper. Convergences to stationarity, convergence of iterates are not claimed.

Attaining stronger convergence results on iterates, convergence to stationarity and function value for NAG in general requires intense amount of works. In fact, these type of convergences claims are nontrivial in the convex settings in general as well. For example, see Bauchke et al. [2], when the minimizer of F doesn't exist. To limit the scope, we focus on Nesterov's acceleration which differs from inertial momentum from Polyak. Polyak momentum (also referred to as heavy-ball inertia in the literatures) differs from Nesterov's acceleration by evaluating the gradient at the current iterates and does the extrapolation after the gradient evaluation. The relationship between Polyak and Nesterov momentum acceleration remains non-trivial. In fact, both type of acceleration can present at the same time, see Maulén and Peypouquet [26]. It is sometime referred to as "Hessian Damping" in the literatures. There are three pillars that supported Catalyst Accelerations:

(i) Variance reduced algorithm with known complexities.

(ii) Theories of inexact proximal point method and their convergences.

(iii) Nesterov's accelerations or some other first order acceleration method with inexact oracles.

To the best of our interpretation of Catalyst is through the theories of inexact proximal point because NAG is not the unque option and other acceleration techniques can be used as well. For example, see Cai et al. [8] for inexact Halpern acceleration for solving variational inequalities. Therefore, the theories of inexact proximal point and the algorithm used to evaluate lies deeper in the theoretical aspect of Catalyst.

We now list new developments for Nesterov's acceleration in the nonconvex settings with stronge convergence claim. Preprint by Hermant et al. [21] present results that the fixed momentum variant of NAG has linear convergence on the class of strongly quasar convex functions. Preprint by Bu, Mesbashi [7] discusses using an idea called weak estimating sequence to show the linear convergence of NAG for weakly quasi-strongly-convex functions. The perspectives on how to generalize NAG to the nonconvex settings are diverse, and the issue is compounded by the fact that many variants of NAG also exist.

For theories of inexact proximal point method see Khanh et al. [22] for an algorithmic framework of inexact proximal point method on weakly convex function. The class of weakly convex function is particularly interesting because it has general convergence claim in Ghadimi and Lan [16], and in Chapter 4 of the book preprint by Recht and Wright [35]. It's unclear how their convergence theories can be combined with NAG in the non-convex case, but it's worth investigating.

5.3.2 Our ideas on future works of Catalyst Acceleration

Restricting ourselves to the convexity, this is our question:

> What if Assumption 5.10 is false?

This is a detail in the Catalyst framework, and it may not be obvious to the readers that the complexity assumption is sufficient to accommodate objective functions F to be the class of all L-Lipschitz smooth function. The remaining of this section will:

(i) elucidate the relation of F being L-Lipschitz smooth and Assumption 5.10;

(ii) show that removing the assumption will yield non-trivial answers for the total complexity of Catalyst because theorems Nesterov's lower bound complexity results for black box smooth/nonsmooth objectives can't be applied to Catalyst as a whole;

(iii) show that some other the existing literatures that seems to hold the answer but actually doesn't fully address the question;

(iv) discuss literatures relevant to the question.

The concern starts by considering the necessary assumption in VRMs. Defazio et al. [11] SAGA, Finito et al. [12] requires Lipschitz continuous gradient on each individual f_i in the objective function $F = \sum_{i=1}^{n} f_i$. Schmidt et al. [39] also requires Lipschitz smoothness for the gradient of the objective function. Xiao, Zhang, SVRG [42] requires Lipschitz gradient assumption as well. This seemingly raises a crucial question: "Where does the Lipschitz continuous gradient of the smooth part make its presence in the Catalyst Framework?"

A careful reader will realize that Assumption 5.10 encapsulates the Lipschitz smoothness assumption of VRMs because the proximal point sub-problem is smooth and strongly convex and lower bound from Theorem 2.1.13 from Nesterov's book [32] applies and it has a linear convergence rate. Removing the smoothness assumption of F, $\mathbb{M}$ solves subproblem $\mathcal{M}_{\kappa^{-1}F}(\cdot, y_k)$ which is only strongly convex. Then, lower bound on convergence rate of any first order algorithm becomes $\mathcal{O}(1/k)$ by Theorem 3.2.5 from Nesterov's book.

From a complexity theory point of view, Catalyst acceleration didn't attempt at narrowing the lower complexity bound for functions that are convex and nonsmooth. At the same time, the theories of lower bound on blackbox optimization also cannot be applied for Catalyst Accelerations because they are based on different computational oracles. It is possible that, for any blackbox nonsmooth optimization problem, the same lower bound convergence rate $\mathcal{O}\left(1/\sqrt{k}\right)$ as stated in Theorem 3.2.1 in Nesterov's book [32] is true. Unfortunately it is

nontrivial to see it as true because $\mathbb{A}$ optimizes Moreau Envelope $\mathcal{M}^*_{\kappa^{-1}F}(x)$ through inexact evaluation and the Moreau Envelope has Lipschitz gradient. For $\mathbb{A}$ Theorem 3.2.1 doesn't apply. So, here is the refined question:

> If we assume that F is any convex function (potentially nonsmooth with gradient that are not locally Lipschitz), would there exist any kind of regularization parameters $(\kappa_k)_{k\geq 0}$, error sequence ϵ_k such that the worst case total complexity of Catalyst retains convergence rate equals or faster than $\mathcal{O}\left(1/\sqrt{k}\right)$ (The lower bound of all first order blackbox convex optimization)? I.e: Does the idea of an inexact proximal point computational oracle lift the theoretical lower bound complexity on first order blackbox convex optimization?

If the answer is that it can be faster than the lower bound, then we had extended the Catalyst Framework to convex function without Lipschitz smooth gradient, otherwise, it leads to the unintuitive result that lower bound $\mathcal{O}(1/\sqrt{k})$ still holds which will prove a concept of universality of the lower bound results for blackbox convex optimizations.

To the best of our knowledge, nobody has yet answered the question directly in the literature, but we found some relevant literatures nonetheless. In Nesterov 2014 [31], he considered accelerated gradient for function with Hölder Continuous gradient. Nesterov continued and expanded on the results obtained by Devolder et al. [13] on inexact oracles for blackbox optimizations which showed that the inexact oracle for Lipschitz smooth function can be viewed as exact for another perturbed function with Hölder Continuous gradient. He proposed three first order algorithms where the last one, a variant of accelerated gradient method that converges optimally for all functions with Hölder Continuous gradient without knowing the Holder continuity constant in advance. His work is not considered directly relevant to Catalyst because it's not yet obvious because it's missing an inexact accelerated proximal point, and he uses the theories of inexact oracles from a purely theoretical standpoint.

A more recent development in Yang and Toh [43] exposes the theories of Inexact Accelerated Bregman Proximal Point method in the context of applying it for optimal transport. They showed that the lower bound remains $\mathcal{O}(1/k)$, and in the special case when the kernel function is strongly convex relative to some norm, the optimal convergence rate of $\mathcal{O}(1/k^2)$ is reached. These results are consistent with Nesterov's book chapter 6 [32] and Dragomir et al. [14].

Of course, the class of convex function with Hölder Continuous gradient (i.e: the class of function that is relatively smooth to some Bregman kernel) doesn't encapsulate the class of all convex functions where inexact proximal point is applicable. The original question would remain open.

To tackle the problem worst case complexity problem in general, it's worth mentioning

Performance estimation problem (PEP). PEP puts the set of all convex function as their iterates, gradient, and function value and the algorithm all together into Semi-definite programming constraints (SPD) and it solves for the worst case convergence. Proposed by Drori and Teboulle [15] is the method of PEP for worst case analysis for convex programming, they showed that QCQP (Quadratic Constrained Quadratic Program) for formulation of the first order worst case iteration complexity problem can be reduced to a SPD such that it's tight. And Goujaud et al. [17] gives a Python library with APIs[1] to produce worst case convergence by simply implementing the algorithm and the program formulates the SDP and solves it in the back. This library supports inexact gradient, exact/inexact proximal point, and many other computational oracles. These tools help us to formulate/verify complexity results if our questions on the total complexity of Catalyst can be put into the PEP framework.

References

[1] K. Ahn and S. Sra, *Understanding Nesterov's acceleration via proximal point method*, in Symposium on Simplicity in Algorithms, SIAM, June 2022, pp. 117–130.

[2] H. H. Bauschke, M. N. Bui, and X. Wang, *Applying FISTA to optimization problems (with or) without minimizers*, Mathematical Programming, 184 (2020), pp. 349–381.

[3] A. Beck, *First-order Methods in Optimization*, MOS-SIAM Series in Optimization, SIAM, 2017.

[4] D. P. Bertsekas, *Incremental proximal methods for large scale convex optimization*, Mathematical Programming, 129 (2011), pp. 163–195.

[5] ——, *Incremental gradient, subgradient, and proximal methods for convex optimization: A survey*, Dec. 2017.

[6] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, *Julia: A fresh approach to numerical computing*, SIAM Review, 59 (2017), pp. 65–98.

[7] J. Bu and M. Mesbahi, *A Note on Nesterov's Accelerated Method in Nonconvex Optimization: a Weak Estimate Sequence Approach*, June 2020. arXiv:2006.08548.

[8] X. Cai, A. Alacaoglu, and J. Diakonikolas, *Variance Reduced Halpern Iteration for Finite-Sum Monotone Inclusions*, Oct. 2023. arXiv:2310.02987 [cs, math].

[1] APIs: A plural acronym for Application Programming Interface.

[9] A. Chambolle and C. Dossal, *On the convergence of the iterates of the "Fast iterative shrinkage/thresholding algorithm"*, Journal of Optimization Theory and Applications, 166 (2015), pp. 968–982.

[10] A. Chambolle and T. Pock, *An introduction to continuous optimization for imaging*, Acta Numerica, 25 (2016), pp. 161–319.

[11] A. Defazio, F. Bach, and S. Lacoste-Julien, *SAGA: A fast increment al gradient method with support for non-strongly convex composite objectives*, Dec. 2014.

[12] A. Defazio, J. Domke, and T. Caetano, *Finito: A faster, permutable incremental gradient method for big data problems*, in Proceedings of the 31st International Conference on Machine Learning, PMLR, June 2014, pp. 1125–1133.

[13] O. Devolder, F. Glineur, and Y. Nesterov, *First-order methods of smooth convex optimization with inexact oracle*, Mathematical Programming, 146 (2014), pp. 37–75.

[14] R.-A. Dragomir, A. B. Taylor, A. d'Aspremont, and J. Bolte, *Optimal complexity and certification of Bregman first-order methods*, Mathematical Programming, 194 (2022), pp. 41–83.

[15] Y. Drori and M. Teboulle, *Performance of first-order methods for smooth convex minimization: a novel approach*, Mathematical Programming, 145 (2014), pp. 451–482.

[16] S. Ghadimi and G. Lan, *Accelerated gradient methods for nonconvex nonlinear and stochastic programming*, Mathematical Programming, 156 (2016), pp. 59–99.

[17] B. Goujaud, C. Moucer, F. Glineur, J. M. Hendrickx, A. B. Taylor, and A. Dieuleveut, *PEPit: computer-assisted worst-case analyses of first-order optimization methods in Python*, Mathematical Programming Computation, 16 (2024), pp. 337–367.

[18] R. M. Gower, M. Schmidt, F. Bach, and P. Richtárik, *Variance-reduced methods for machine learning*, Proceedings of the IEEE, 108 (2020), pp. 1968–1983.

[19] G. N. Grapiglia and Y. Nesterov, *Accelerated regularized newton methods for minimizing composite convex functions*, SIAM Journal on Optimization, 29 (2019), pp. 77–99.

[20] O. Guler, *New proximal point algorithms for convex minimization*, SIAM Journal on Optimization, 2 (1992), pp. 649–664.

[21] J. Hermant, J.-F. Aujol, C. Dossal, and A. Rondepierre, *Study of the behaviour of Nesterov Accelerated Gradient in a non convex setting: the strongly quasar convex case*. May 2024.

[22] P. D. Khanh, B. S. Mordukhovich, V. T. Phat, and D. B. Tran, *Inexact proximal methods for weakly convex functions*, Journal of Global Optimization, (2025).

[23] J. Lee, C. Park, and E. Ryu, *A Geometric structure of acceleration and its role in making gradients small fast*, in Advances in Neural Information Processing Systems, vol. 34, 2021, pp. 11999–12012.

[24] H. Lin, J. Mairal, and Z. Harchaoui, *A universal catalyst for first-order optimization*, in Procedings of Advances in Neural Information Processing Systems, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds., vol. 28, 2015.

[25] ——, *Catalyst acceleration for first-order convex optimization: from theory to practice*, Journal of Machine Learning Research, 18 (2018), pp. 1–54.

[26] J. J. Maulén and J. Peypouquet, *A speed restart scheme for a dynamics with hessian-driven damping*, Journal of Optimization Theory and Applications, 199 (2023), pp. 831–855.

[27] I. Necoara, Y. Nesterov, and F. Glineur, *Linear convergence of first order methods for non-strongly convex optimization*, Mathematical Programming, 175 (2019), pp. 69–107.

[28] Y. Nesterov, *A method for solving the convex programming problem with convergence rate O(1/kˆ2)*, Proceedings of the USSR Academy of Sciences, (1983).

[29] Y. Nesterov, *Accelerating the cubic regularization of Newton's method on convex problems*, Mathematical Programming, 112 (2008), pp. 159–181.

[30] ——, *Gradient methods for minimizing composite functions*, Mathematical Programming, 140 (2013), pp. 125–161.

[31] Y. Nesterov, *Universal gradient methods for convex optimization problems*, Mathematical Programming, 152 (2015), pp. 381–404.

[32] Y. Nesterov, *Lectures on Convex Optimization*, vol. 137 of Springer Optimization and Its Applications, Springer International Publishing, 2018.

[33] W. Noel, *Nesterov's method for convex optimization*, SIAM Review, 65, pp. 539–562.

[34] C. Paquette, H. Lin, D. Drusvyatskiy, J. Mairal, and Z. Harchaoui, *Catalyst for gradient-based nonconvex optimization*, in Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, PMLR, Mar. 2018, pp. 613–622.

[35] B. Recht and S. Wright, *Optimization for Modern Data Analysis*.

[36] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM Journal on Control and Optimization, 14 (1976), pp. 877–898.

[37] R. T. Rockafellar, *Convex Analysis*, Princeton Landmarks in mathematics and physics, Princeton Univ. Press, Princeton, NJ, 10. print. and 1. paperb. print ed., 1997.

[38] E. K. Ryu and W. Yin, *Large-scale Convex Optimization: Algorithms & Analyses via Monotone Operators*, Cambridge University Press, 2022.

[39] M. Schmidt, N. Le Roux, and F. Bach, *Minimizing finite sums with the stochastic average gradient*, Mathematical Programming, 162 (2017), pp. 83–112.

[40] M. Schmidt, N. L. Roux, and F. Bach, *Convergence rates of inexact proximal-gradient methods for convex optimization*, Dec. 2011. arXiv:1109.2415.

[41] R. Tibshirani, *Regression shrinkage and selection via the Lasso*, Journal of the Royal Statistical Society. Series B (Methodological), 58 (1996), pp. 267–288.

[42] L. Xiao and T. Zhang, *A proximal stochastic gradient method with progressive variance reduction*, SIAM Journal on Optimization, 24 (2014), pp. 2057–2075.

[43] L. Yang and K.-C. Toh, *Bregman proximal point algorithm revisited: A new inexact version and its inertial variant*, SIAM Journal on Optimization, 32 (2022), pp. 1523–1554.

[44] C. Ying and P. Jong-Shi, *Modern Nonconvex Nondifferentiable Optimization*, vol. 1 of MOS-SIAM Series on Optimization, MOS-SIAM, 2021.