Composite optimization models via proximal gradient method with a novel enhanced adaptive stepsize

Pham Thi Hoai[a,*], Nguyen Pham Duy Thai[a]

[a] *Faculty of Mathematics and Informatics, Hanoi University of Science and Technology, 1 Dai Co Viet Road, Hanoi, Vietnam*

Abstract

We first consider the convex *composite optimization models* with the *local Lipschitzness* condition imposed on the gradient of the differentiable term. The classical *proximal gradient method* will be studied with our novel *enhanced adaptive* stepsize selection. To obtain the convergence of the proposed algorithm, we establish a *sufficient decrease type inequality* associated with our new stepsize choice. This allows us to demonstrate the descent of the objective value from some fixed iteration and yield the *sublinear convergence rate* of the new method. Especially, in the case of *locally strong convexity* of the smooth term, our algorithm converges *Q-linearly*. Additionally, we further show that our method can be applied to *nonconvex* composite optimization problems provided that the differentiable function has a globally Lipschitz gradient. Finally, the efficiency of our proposed algorithms is shown by numerical results for numerous applicable test instances in comparison with the other state-of-the-art algorithms.

Keywords: proximal gradient method, nonlinear programming, composite optimization model, locally Lipschitz gradient, convex programming, nonconvex programming, quadratic programming, unconstrained nonlinear programming, constrained nonlinear programming, Lasso problem
2010 MSC: 49J40, 47H04, 47H10

1. Introduction

1.1. Problem description and motivation

Composite optimization models (COM) have arisen from many real-life applications, such as machine learning, signal processing, data science, etc, and have received a lot of attention recently, see e.g., [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. The formulation of (COM) considered in this paper can be described as follows:

$$\min_{x\in\mathbb{R}^n} F(x) = f(x) + g(x), \tag{P}$$

where f and g are functions satisfying *Assumption 1* below.

Assumption 1. *(A1)* $g : \mathbb{R}^n \to (-\infty, +\infty]$ *is a proper and closed convex function.*

(A2) $f : \mathbb{R}^n \to (-\infty, +\infty]$ *is proper and closed such that dom*(f) *is convex, dom*$(g) \subset$ *int(dom*(f)*) and* f *is differentiable on int(dom*(f)*).*

(A3) The optimal solution set X^* *of (P) is nonempty and* F_* *stands for the optimal value of (P).*

One of the conventional methods for solving the problem (P) is *proximal gradient method* (PG) introduced by Fukushima and Mine [14] in 1981 and has now become classical. As a matter of fact, the further origin of the proximal gradient method can be traced back to 1970s with the work of Brucks [15] and Passty [16] in the more general setting of forward backward splitting method. The detailed methodology of the PG method can be found in [5, 6]. It is observed that the optimal condition for the problem (P) relates to the concept of its stationary points. Specifically, if $x^* \in \text{int}(\text{dom}(f))$ is a local optimal solution of (P) then it should be a *stationary point* of (P), i.e., for some $t > 0$

$$x^* = \text{Prox}_{tg}(x^* - t\nabla f(x^*)), \tag{1.1}$$

[*] Corresponding author

Email addresses: `hoai.phamthi@hust.edu.vn` (Pham Thi Hoai), `duythai09092002@gmail.com` (Nguyen Pham Duy Thai) where $\mathrm{Prox}(.)$ is the proximal operator and is defined as the unique optimal solution of the minimization problem

$$\mathrm{Prox}_{tg}(y) = \mathrm{argmin}_{x\in\mathbb{R}^n}\left\{g(x) + \frac{1}{2t}\|x-y\|^2\right\}. \tag{1.2}$$

In the convex situation of (P), i.e., f is convex, the set of stationary points of (P) is coincident with X^*. One can see [5] (Theorem 3.72, 10.7) for more details. Based on the stationary condition (1.1), starting from some $x^0 \in$ int(dom(f)), the well-known PG method to solve problem (P) is designed by generating the sequence $\{x^k\}$ according to the rule

$$x^{k+1} = \mathrm{Prox}_{t_k g}(x^k - t_k\nabla f(x^k)),\ \ k = 0, 1, 2, ..., \tag{1.3}$$

The PG scheme (1.3) is useful if we can compute $\mathrm{Prox}_{t_k g}(x^k - t_k\nabla f(x^k))$ easily by some explicit formulas. Though, there is a list of functions whose *proximal operator* is analytically computable and that list can be found in [5]; for instances, g is the ℓ_1 norm or the indicator function of a closed convex set $C \subset \mathbb{R}^n$. In formula (1.3), $t_k > 0, k = 0, 1, 2, ...$ are defined as *stepsizes* which play a crucial role in the proximal gradient scheme. A suitable stepsize selection can be drawn in the two main points: firstly, it should guarantee the convergence of $\{x^k\}$ to some stationary point of problem (P); secondly, it should navigate x^k to a "good" stationary point. i.e., providing, for example, the low objective value as much as possible with a cheap computational cost. For the class of L_f- smooth function f, i.e.,

$$\|\nabla f(x) - \nabla f(y)\| \le L_f\|x-y\|,\ \forall x, y \in \text{int(dom}(f)),$$

the stepsize t_k in (1.3) can be controlled flexibly by using *constant stepsize* in $\left(0, \frac{2}{L_f}\right)$ or *backtracking line search* rule. Followed by [5] (Theorem 10.21), one gets the *sublinear rate* of convergence, i.e., computational complexity $O(\frac{1}{k})$ of $F(x^k) - F_*$ if f is assumed to be convex and t_k is either in $\left(0, \frac{1}{L_f}\right]$ or taken by backtracking procedure. In the case where f is strongly convex, the convergence rate of $\{x^k\}$ to some $x^* \in X^*$ is proved to be Q-linear. These properties can be seen as the generalization of the convergence results for the gradient descent method solving unconstrained nonlinear optimization problems, i.e., problem (P) with $g = 0$.

Recently, researchers have been concerned about problem (P) *without the global Lipschitzness assumption on* ∇f, see, e.g., [17, 18, 9, 19, 20] because the class of such functions occurs in many applied problems (see e.g., [19, 21, 22] and the references therein). In 2017, Bauschke et al. [17] proposed *NoLips Algorithm* that requires Bregman distances-based computation and constant L in the *Lipschitz-like/convexity condition* (LC). One can see [23] to find the role of non-Euclidean proximal distances of Bregman type in the development and analysis of some typical first order optimization algorithms. The stepsize selection of NoLips is then chosen in $(0, \frac{2-\delta}{L})$. This algorithm is shown in [17] to have the convergent results similar to the ones of the normal PG scheme. Following that, Dragomir et al. [24] give a lower bound to prove that the $O\left(\frac{1}{k}\right)$ convergence rate of the NoLips method is optimal for the class of problems satisfying the relative smoothness assumption. The other recent results on the convergence of the PG method without globally Lipschitz assumption have been studied in Kanzow and Mehlitz [19] and then Jia et al. [9]. Their proposed methods can be applied for the nonconvex setting of (P) with the presence of Kurdyka–Łojasiewicz condition. The stepsize choice is based on backtracking line search procedure. Nevertheless, one knows that there are some restrictions of taking stepsize within $\left(0, \frac{2}{L_f}\right)$ or $\left(0, \frac{2-\delta}{L}\right)$ like: firstly, the process of finding these constants is not easy in general and secondly, if the coefficients L_f or L are large then constant stepsizes will be very small and that may take long executing time. Analogously, the backtracking computation for stepsize selection probably consumes expensive cost and also may cause the stepsize to gradually decrease to a tiny number. To overcome the mentioned drawbacks above, an interesting question should be considered is:

Question 1.1. *Under Assumption 1 and assuming that f is convex and has a locally Lipschitz gradient, is there an adaptive way to find stepsizes explicitly for PG scheme solving problem (P) such that we do not need neither estimating constants like $L_f, L, ...$ nor backtracking line search procedures?*

1.2. **Some recent algorithms considering Question 1.1**

In the specific context of the problem (P) with $g = 0$, two algorithms named AdGD and NGD were proposed by Malitsky and Mishchenko [25] (2019) and Hoai et al. [26] (2024), respectively. Both of them use explicit stepsize strategies based on the local curvature of f. In the general setting of problem (P), to give an answer to Question 1.1, Malitsky and Mishchenko [27] have developed their method AdGD [25] to AdPG (Adaptive Proximal Gradient) for solving the problem (P) recently. The stepsize of AdPG is defined by

$$t_k = t_{k-1} \min \left\{ \sqrt{\frac{2}{3} + \theta_{k-1}}, \frac{1}{\sqrt{\left[\frac{2t_{k-1}^2 \|\nabla f(x^k) - \nabla f(x^{k-1})\|^2}{\|x^k - x^{k-1}\|^2} - 1\right]^+}} \right\}, \tag{AdPG}$$

where $\theta_0 = \frac{1}{3}, \theta_k = \frac{t_k}{t_{k-1}}, k \geq 1$. And for some $t \in \mathbb{R}$, the notation t^+ stands for $\max\{t, 0\}$. The iterates of AdPG are proved to converge to an optimal solution of (P) with the *worst-case sublinear* rate, i.e., the complexity $O(\frac{1}{k})$ of $\min\limits_{1 \leq i \leq k} (F(x^i) - F_*)$. In parallel with this work, Latafat et al. [28] proposed adaPGM that has

$$t_k = t_{k-1} \min \left\{ \sqrt{1 + \frac{t_{k-1}}{t_{k-2}}}, \frac{1}{2\sqrt{\left[t_{k-1}\left(\frac{t_{k-1}\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 - \langle \nabla f(x^k) - \nabla f(x^{k-1}, x^k - x^{k-1}\rangle}{\|x^k - x^{k-1}\|^2}\right)\right]^+}} \right\}, k \geq 1. \tag{adaPGM}$$

Soon after, adaPGM is generalized to be AdaPG^{q,r} in Latafat et al. [29] with

$$t_k = t_{k-1} \min \left\{ \sqrt{\frac{1}{q} + \frac{t_{k-1}}{t_{k-2}}}, \sqrt{\frac{1 - r/q}{\left[\frac{t_{k-1}^2 \|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 + 2t_{k-1}(r-1)\langle \nabla f(x^k) - \nabla f(x^{k-1}, x^k - x^{k-1}\rangle}{\|x^k - x^{k-1}\|^2} - (2r-1)\right]^+}} \right\}, \tag{AdaPG^{q,r}}$$

where $\frac{1}{2} \leq r < p \leq \frac{3+\sqrt{5}}{2}, t_0 = t_{-1} > 0, k \geq 1$. Notably, AdaPG^{q,r} recovers AdPG if $(p, r) = \left(\frac{3}{2}, \frac{3}{4}\right)$ and adaPGM if $(p, r) = \left(1, \frac{1}{2}\right)$ with slight improvements (see [29] for details). The convergence of AdaPG^{q,r} is then established with the worst-case sublinear convergence rate.

1.3. **Contributions**

In this paper, we develop the idea of adaptive stepsize in NGD [26] to be (NPG) used for proximal gradient scheme (1.3) solving problem (P) with locally Lipschitz gradient condition imposed on the smooth term f as follows:

For $0 < c_1 < c_0 < \frac{1}{\sqrt{2}}$ and a convergent positive series $\sum\limits_{k=0}^{+\infty} \gamma_k, t_{-1} = t_0 > 0$

$$\begin{array}{ll} \textbf{If} & \|\nabla f(x^k) - \nabla f(x^{k-1})\| > \dfrac{c_0}{t_{k-1}} \|x^k - x^{k-1}\| \\ \textbf{then} & t_k = c_1 \dfrac{\|x^k - x^{k-1}\|}{\|\nabla f(x^k) - \nabla f(x^{k-1})\|} \\ \textbf{else} & \gamma'_{k-1} = \gamma_{k-1} \qquad \text{if } \dfrac{t_{k-1}}{t_{k-2}} < 1 \text{ then } \gamma'_{k-1} = \min\left\{\gamma_{k-1}, \sqrt{1 + \dfrac{t_{k-1}}{t_{k-2}}} - 1\right\} \\ & t_k = (1 + \gamma'_{k-1}) t_{k-1}. \end{array} \tag{NPG}$$

More precisely, we give a positive response to Question 1.1 with the main contributions including:

- Firstly, NPG recovers NGD [26] in the case $g = 0$ but provides $\sqrt{2}$ times bigger range of step length. In particular, the constants c_0, c_1 belong to $(0, 1/2)$ for NGD but $(0, 1/\sqrt{2})$ for NPG.

- Secondly, we provide *a sufficient decrease type inequality* associated with NPG. This inequality plays a crucial role to prove the decreasing of the objective function from some fixed iteration. And more importantly, it derives the sublinear rate of the proposed method. In addition, in the case of locally strong convexity of f, our sufficient descent lemma helps to deduce the Q-linear rate of the iterates. Observably, the recent algorithms AdPG [27], adaPGM [28] and AdaPG^{q,r} [29] just obtain the *worst-case* sublinear rate. The lack of the descent property of the objective value prevents the existing methods from achieving the sublinear rate (for (P) under Assumption 1) and Q-linear rate if f is complemented the locally strongly convexity.

- Moreover, when ∇f is globally Lipschitz, we further show that our method can be extended to the nonconvex composite optimization models. As a byproduct, one special version solving problem (P) is designed in the case where f is indefinite quadratic with the capability of enlarging stepsize.

- Besides the adaptation in computation of the stepsize selection as the existing methods AdPG, adaPGM, AdaPG^{q,r}, the sequence of our stepsize NPG is confirmed to be increasing to a positive value.
- Finally, we implement our new algorithms in comparison with the recent ones for numerous applicable test instances including: 1. Lasso problems; 2. Minimum length piecewise-linear curve subject to equality constraints; 3. Dual of the entropy; 4. Maximum likelihood estimate of the information matrix; 5. Nonnegative matrix factorization. Data used for testing are randomly generated with diversity dimensions from small to large. The reported results demonstrate the crucial efficiency of the proposed method.

1.4. Structure of the paper

The rest of the paper is structured as follows. After summarizing some necessary preliminaries in Section 2, we propose our new proximal algorithm in Section 3 for solving the convex situation of (P) under the locally Lipschitz condition of ∇f. In the sequel, we consider a nonconvex case of (P) with an other new algorithm. Section 5 presents a particular version of proposed method applied for the indefinite quadratic function f. The numerical experiments on a set of practical examples are stated in Section 6. Lastly, the paper is closed by some conclusions in Section 7.

2. Preliminaries

In this section, we recall some necessary fundamental results which are useful to derive our main contributions in the upcoming sections.

Lemma 2.1. *Under Assumption 1, the sequence $\{x^k\}$ generated by proximal gradient scheme (1.3) for solving the problem (P) has the following properties:*

(i) there exists $\overline{\partial} g(x^{k+1}) \in \partial g(x^{k+1})$ such that $x^{k+1} = x^k - t_k\left(\nabla f(x^k) + \overline{\partial} g(x^{k+1})\right)$;

(ii) for all $x \in \operatorname{int}(\operatorname{dom}(f))$, we have

$$g(x) - g(x^{k+1}) \geq \left\langle x^{k+1} - x, \nabla f(x^k) + \frac{x^{k+1} - x^k}{t_k} \right\rangle. \tag{2.1}$$

Proof. (i) Since $x^{k+1} \in \operatorname*{argmin}_{x\in\mathbb{R}^n} \left\{ g(x) + \frac{1}{2t_k}\left\|x - (x^k - t_k\nabla f(x^k))\right\|^2 \right\}$ then

$$0 \in \partial g(x^{k+1}) + \frac{1}{t_k}\left(x^{k+1} - x^k + t_k\nabla f(x^k)\right).$$

Hence there exists $\overline{\partial} g(x^{k+1}) \in \partial g(x^{k+1})$ such that

$$x^{k+1} = x^k - t_k(\nabla f(x^k) + \overline{\partial} g(x^{k+1})). \tag{2.2}$$

(ii) From (i) and the convexity of g we are easy to get that

$$\begin{aligned} g(x) - g(x^{k+1}) &\geq \left\langle x - x^{k+1}, \overline{\partial} g(x^{k+1}) \right\rangle \\ &= \left\langle x^{k+1} - x, \nabla f(x^k) + \frac{x^{k+1} - x^k}{t_k} \right\rangle. \end{aligned}$$

□

Lemma 2.2 (Opial lemma)**.** *Let $\{x^k\} \subset \mathbb{R}^n$ be a bounded sequence where its cluster points in $X \subset \mathbb{R}^n$ and the real sequence $\{a_k\} \subset \mathbb{R}_+$. If*

$$\|x^{k+1} - x\|^2 + a_{k+1} \leq \|x^k - x\|^2 + a_k, \ \ \forall x \in X, \tag{2.3}$$

then $\{x^k\}$ converges to an element of X.

Proof. One can see the proof of Lemma 2 in [25]. □

3. A new proximal gradient algorithm for the convex case of the problem (P) with locally Lipschitz ∇f

It is worth noting that when f has a globally Lipschitz gradient with constant L_f and t_k is chosen as a fixed number in $\left(0, \frac{2}{L_f}\right)$ or by line search strategy, the common technique establishing the convergence of proximal gradient method (1.3) for solving (P) is related to *the sufficient decrease inequality*, i.e., showing the existence of a positive constant M such that

$$F(x^k) - F(x^{k+1}) \geq M\|x^{k+1} - x^k\|, \;\; k \geq 0, \qquad (\text{ sufficient decrease ieq.})$$

For the proximal gradient algorithms using adaptive stepsizes solving problem (P) in the literature like AdPG [27], adaPGM[28] and AdaPG^{q,r} [29], the obstacle of the locally Lipschitz gradient condition has been overcome by constructing Lyapunov type functions and then obtain the boundedness of the iterates. Since, on the compact set $T = \overline{conv}\left(\{x^0, x^1, ...\} \cup X^*\right)$ all properties of a function f with locally Lipschitz gradient can be operated as those of a globally Lipschitz gradient function. The convergence of their proposed approaches are then deduced by relying on the interesting techniques different from the usual way based on the sufficient decrease ieq. . However, the absence of descent property prevents their algorithms from achieving sublinear convergence rate $O\left(\frac{1}{k}\right)$ of $F(x^k) - F_*$ but only worst-case convergence rate $O\left(\frac{1}{k}\right)$ of $\min\limits_{1\leq i\leq k}\{F(x^i) - F_*\}$ for solving convex problem (P) under locally Lipschitz gradient condition. Notably, our stepsize selection NPG is not only adapted with the local curvature of f but also controllable by using the pre-selected positive convergent series $\sum\limits_{k=0}^{+\infty} \gamma_k$. Then, in contrast of the existing algorithms, our proposed algorithms based on NPG stepsize can establish sufficient decrease ieq. for (P). We will successively explore how to establish this inequality in the subsequent parts of the paper.

Firstly, in this section, we set up NPG for the proximal gradient method to solve problem (P) under *Assumption 1* and *Assumption 2* below.

Assumption 2. *f is convex and has a locally Lipschitz gradient.*

Algorithm 3.1 (NPG1)

```
Step 0. Select $t_0 > 0$, $0 < c_1 < c_0 < \frac{1}{\sqrt{2}}$ and a positive real sequence $\{\gamma_k\}$ such that $\sum\limits_{k=0}^{+\infty} \gamma_k < \infty$. Choose $x^0 \in \mathrm{int}(\mathrm{dom}(f))$,
    $x^1 = \mathrm{Prox}_{t_0 g}(x^0 - t_0\nabla f(x^k)), t_{-1} = t_0$ and set $k = 1$.
Step 1.
    If $\|\nabla f(x^k) - \nabla f(x^{k-1})\| > \frac{c_0}{t_{k-1}}\|x^k - x^{k-1}\|$        (3.1)
    then $t_k = c_1 \frac{\|x^k - x^{k-1}\|}{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}$        (3.2)
    else $\gamma'_{k-1} = \gamma_{k-1}$
        if $\frac{t_{k-1}}{t_{k-2}} < 1$ then $\gamma'_{k-1} = \min\left\{\gamma_{k-1}, \sqrt{1 + \frac{t_{k-1}}{t_{k-2}}} - 1\right\}$        (3.3)
        $t_k = (1 + \gamma'_{k-1})t_{k-1}.$        (3.4)
Step 2. Compute $x^{k+1} = \mathrm{Prox}_{t_k g}(x^k - t_k\nabla f(x^k))$.
Step 3. If $\|x^{k+1} - x^k\| < \epsilon$ then STOP else setting $k := k + 1$ and return to Step 1.
```

Remark 3.1. It is observed that, in the case $g = 0$, Algorithm 3.1 (NPG1) becomes NGD [26] with larger bounds of c_0, c_1. In particular, for NGD, $c_0, c_1 \in \left(0, \frac{1}{2}\right)$ but for NPG1, $c_0, c_1 \in \left(0, \frac{1}{\sqrt{2}}\right)$.

Analogous to existing methods we need to prepare some lemmas which will help us to prove the boundedness of $\{x^k\}$ - a key step to overcome the difficulties generated by the locally Lipschitz continuity of ∇f.

Lemma 3.2. *For all $x \in \mathrm{int}(\mathrm{dom}(f))$ we have*

$$\|x^{k+1} - x\|^2 + 2t_k\left(F(x^k) - F(x)\right) \leq \|x^k - x\|^2 + t_k^2\left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2.$$

Proof. From Lemma 2.1 (ii), for all $x \in \operatorname{int}(\operatorname{dom}(f))$

$$\begin{aligned} 2t_k \left(g(x^{k+1}) - g(x)\right) &\leq 2\left\langle x^{k+1} - x^k + t_k \nabla f(x^k), x - x^{k+1} \right\rangle \\ &= \|x^k - x\|^2 - \|x^{k+1} - x^k\|^2 - \|x^{k+1} - x\|^2 + \\ &\quad + 2t_k \left\langle \nabla f(x^k), x - x^{k+1} \right\rangle. \end{aligned} \tag{3.5}$$

Using the convexity of f and g, we continue evaluating

$$\begin{aligned} \langle \nabla f(x^k), x - x^{k+1} \rangle &= \langle \nabla f(x^k), x - x^k \rangle + \langle \nabla f(x^k) + \overline{\partial} g(x^k), x^k - x^{k+1} \rangle + \langle \overline{\partial} g(x^k), x^{k+1} - x^k \rangle \\ &\leq f(x) - f(x^k) + \left\langle \nabla f(x^k) + \overline{\partial} g(x^k), x^k - x^{k+1} \right\rangle + g(x^{k+1}) - g(x^k). \end{aligned} \tag{3.6}$$

From (3.5) and (3.6), we derive that

$$\|x^{k+1} - x\|^2 + 2t_k \left(F(x^k) - F(x)\right) \leq \|x^k - x\|^2 + R, \tag{3.7}$$

where

$$\begin{aligned} R &= 2t_k \left\langle \nabla f(x^k) + \overline{\partial} g(x^k), x^k - x^{k+1} \right\rangle - \|x^{k+1} - x^k\|^2 \\ &= t_k \left\langle 2\nabla f(x^k) + 2\overline{\partial} g(x^k) - \nabla f(x^k) - \overline{\partial} g(x^{k+1}), x^k - x^{k+1} \right\rangle \\ &= t_k^2 \left\langle \nabla f(x^k) + 2\overline{\partial} g(x^k) - \overline{\partial} g(x^{k+1}), \nabla f(x^k) + \overline{\partial} g(x^{k+1}) \right\rangle \\ &= t_k^2 \left(\left\| \nabla f(x^k) + \overline{\partial} g(x^k) \right\|^2 - \left\| \overline{\partial} g(x^{k+1}) - \overline{\partial} g(x^k) \right\|^2 \right) \\ &\leq t_k^2 \left\| \nabla f(x^k) + \overline{\partial} g(x^k) \right\|^2. \end{aligned} \tag{3.8}$$

The final conclusion is obtained by (3.7) and (3.8). □

Lemma 3.3. *Let $\{t_k\}$ be a sequence of stepsizes generated by Algorithm 3.1 then there exists $k_0 \in \mathbb{N}$ such that*

$$1 + \frac{t_k}{t_{k-1}} \geq \frac{t_{k+1}^2}{t_k^2} \quad \forall k \geq k_0. \tag{3.9}$$

Proof. If $\|\nabla f(x^{k+1}) - \nabla f(x^k)\| > \frac{c_0}{t_k}\|x^{k+1} - x^k\|$ then $t_{k+1} = \frac{c_1\|x^{k+1}-x^k\|}{\|\nabla f(x^{k+1}) - \nabla f(x^k)\|} < \frac{c_1 t_k}{c_0}$ (by (3.2)). Hence $\frac{t_{k+1}}{t_k} < \frac{c_1}{c_0} < 1$ and (3.9) is followed. Conversely, in the case that $\|\nabla f(x^{k+1}) - \nabla f(x^k)\| \leq \frac{c_0}{t_k}\|x^{k+1} - x^k\|$ then by (3.4), $t_{k+1} = (1+\gamma_k')t_k$ and (3.9) is equivalent to

$$\left(\frac{t_{k+1}}{t_k}\right)^2 = (1 + \gamma_k')^2 \leq 1 + \frac{t_k}{t_{k-1}}. \tag{3.10}$$

Moreover, from (3.3), if $\frac{t_k}{t_{k-1}} \geq 1$ then $\gamma_k' = \gamma_k$ and because $\sum_{k=0}^{+\infty} \gamma_k < +\infty$, there is k_0 such that

$$\gamma_k' = \gamma_k \leq \sqrt{2} - 1 \leq \sqrt{1 + \frac{t_k}{t_{k-1}}} - 1 \quad \forall k \geq k_0. \tag{3.11}$$

For the remaining case $\frac{t_k}{t_{k-1}} < 1$, we have

$$\gamma_k' = \min\left\{\gamma_k, \sqrt{1 + \frac{t_k}{t_{k-1}}} - 1\right\} \leq \sqrt{1 + \frac{t_k}{t_{k-1}}} - 1. \tag{3.12}$$

Thus, (3.9) is proved from (3.11) and (3.12).

□

As mentioned above, the bounded property of the sequence $\{x^k\}$ in the following lemma provides us an important key beyond the challenge of locally Lipschitz continuity of ∇f.

Lemma 3.4. *Let $\{x^k\}$ be a sequence generated by Algorithm 3.1 then the following statements hold*

(i) there exists $k_1 \geq k_0$ such that for all $k \geq k_1$,

$$t_k^2 \left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2 \leq \frac{1}{2}\|x^k - x^{k-1}\|^2 + \frac{t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^k)\right); \tag{3.13}$$

(ii) $\{x^k\}$ is bounded.

Proof. *(i)* We have the relation

$$t_k^2 \left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2 = \underbrace{t_k^2 \left\|\nabla f(x^k) - \nabla f(x^{k-1})\right\|^2}_{A} + B, \tag{3.14}$$

where

$$\begin{aligned} B &= 2t_k^2 \left\langle \nabla f(x^k) + \overline{\partial} g(x^k), \nabla f(x^{k-1}) + \overline{\partial} g(x^k)\right\rangle - t_k^2 \left\|\nabla f(x^{k-1}) + \overline{\partial} g(x^k)\right\|^2 \\ &= \frac{t_k^2}{t_{k-1}}\left\langle \nabla f(x^k) + \overline{\partial} g(x^k), x^{k-1} - x^k\right\rangle + \frac{t_k^2}{t_{k-1}}\underbrace{\left\langle \nabla f(x^k) - \nabla f(x^{k-1}), x^{k-1} - x^k\right\rangle}_{\leq 0} \\ &\leq \frac{t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^k)\right). \end{aligned} \tag{3.15}$$

We now prove that there exists $k_1 \geq k_0$ such that

$$A \leq \frac{1}{2}\|x^k - x^{k-1}\|^2 \quad \forall k \geq k_1. \tag{3.16}$$

Indeed, from Algorithm 3.1, if $\left\|\nabla f(x^k) - \nabla f(x^{k-1})\right\| > \frac{c_0}{t_{k-1}}\|x^k - x^{k-1}\|$ then $t_k = \frac{c_1\|x^k - x^{k-1}\|}{\|\nabla f(x^k) - \nabla f(x^{k-1})}\|$ and since $c_1 < \frac{1}{\sqrt{2}}$, we have

$$A = t_k^2\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 = c_1^2\|x^k - x^{k-1}\|^2 < \frac{1}{2}\|x^k - x^{k-1}\|^2.$$

Conversely, if $\|\nabla f(x^k) - \nabla f(x^{k-1})\| \leq \frac{c_0}{t_{k-1}}\|x^k - x^{k-1}\|$ then

$$t_k = (1 + \gamma'_{k-1})t_{k-1} \leq (1 + \gamma_{k-1})\frac{c_0\|x^k - x^{k-1}\|}{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}$$

which follows

$$t_k^2\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 \leq (1 + \gamma_{k-1})^2 c_0^2\|x^k - x^{k-1}\|^2. \tag{3.17}$$

The convergence of $\sum\limits_{k=0}^{+\infty} \gamma_k$ indicates that there exists $k_1 \geq k_0$ satisfying

$$\gamma_{k-1} \leq \frac{1}{\sqrt{2}c_0} - 1 \quad \forall k \geq k_1 \ \left(\frac{1}{\sqrt{2}c_0} - 1 > 0 \text{ since } c_0 < \frac{1}{\sqrt{2}}\right), \tag{3.18}$$

which is equivalent to $(1 + \gamma_{k-1})^2 c_0^2 \leq \frac{1}{2}$ for all $k \geq k_1$. From (3.17) we have (3.16). The combination of (3.14), (3.15) and (3.16) indicates (3.13).

(ii) Using Lemma 3.2 with $x = x^*$ and (3.13), for all $k \geq k_1$ we have

$$\begin{aligned} &\|x^{k+1} - x^*\|^2 + 2t_k\left(F(x^k) - F(x^*)\right) + t_k^2\left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2 \\ &\leq \|x^k - x^*\|^2 + 2t_k^2\left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2 \\ &\leq \|x^k - x^*\|^2 + \|x^k - x^{k-1}\|^2 + 2\frac{t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^k)\right). \end{aligned} \tag{3.19}$$

Nevertheless,

$$\begin{aligned}
t_k^2 \left\|\nabla f(x^k) + \overline{\partial} g(x^k)\right\|^2 &= \left\|t_k \left(\nabla f(x^k) + \overline{\partial} g(x^{k+1})\right) + t_k \left(\overline{\partial} g(x^k) - \overline{\partial} g(x^{k+1})\right)\right\|^2 \\
&= \left\|(x^k - x^{k+1}) + t_k \left(\overline{\partial} g(x^k) - \overline{\partial} g(x^{k+1})\right)\right\|^2 \\
&= \|x^k - x^{k+1}\|^2 + \underbrace{2t_k \left\langle x^k - x^{k+1}, \overline{\partial} g(x^k) - \overline{\partial} g(x^{k+1})\right\rangle}_{\geq 0 \text{ because } g \text{ is convex}} + t_k^2 \underbrace{\|\overline{\partial} g(x^k) - \overline{\partial} g(x^{k+1})\|^2}_{\geq 0} \\
&\geq \|x^k - x^{k+1}\|^2. \qquad (3.20)
\end{aligned}$$

Hence, using inequality (3.20) for the left hand side of (3.19) we obtain that

$$\begin{aligned}
&\|x^{k+1} - x^*\|^2 + 2t_k \left(1 + \frac{t_k}{t_{k-1}}\right)\left(F(x^k) - F(x^*)\right) + \|x^k - x^{k+1}\|^2 \\
&\leq \|x^k - x^*\|^2 + \|x^{k-1} - x^k\|^2 + 2\frac{t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^*)\right). \qquad (3.21)
\end{aligned}$$

Remember that from Lemma 3.3 we derive $2t_k \left(1 + \frac{t_k}{t_{k-1}}\right) \geq \frac{2t_{k+1}^2}{t_k} \ \forall\, k \geq k_1$. Therefore, by (3.21), for all $k \geq k_1$ we have

$$\begin{aligned}
&\|x^{k+1} - x^*\|^2 + \|x^k - x^{k+1}\|^2 + \frac{2t_{k+1}^2}{t_k}\left(F(x^k) - F(x^*)\right) \\
\leq\ &\|x^k - x^*\|^2 + \|x^{k-1} - x^k\|^2 + \frac{2t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^*)\right). \qquad (3.22)
\end{aligned}$$

This inequality follows that

$$\|x^{k+1} - x^*\|^2 + \|x^k - x^{k+1}\|^2 + \frac{2t_{k+1}^2}{t_k}\left(F(x^k) - F(x^*)\right) \leq K, \ \ \forall k \geq k_1 \qquad (3.23)$$

where

$$K = \|x^{k_1} - x^*\|^2 + \|x^{k_1-1} - x^{k_1}\|^2 + \frac{2t_{k_1}^2}{t_{k_1-1}}\left(F(x^{k_1-1}) - F(x^*)\right).$$

The relation (3.23) implies the boundedness of $\{x^k\}$. □

Remark 3.5. From the proof of Lemma 3.4 (eq. (3.11) and (3.18)), we see that if the convergent positive series $\sum_{k=0}^{+\infty} \gamma_k$ is created such that $\gamma_k \leq \min\left\{\frac{1}{\sqrt{2}c_0} - 1, \sqrt{2} - 1\right\}$ for all $k \geq 1$ then $k_1 = 1$ and therefore we obtain (3.22) for all $k \geq 1$. Consequently, by using arguments analogous to those given by Malitsky and Mishchenko [27] we immediately obtain all similar convergent results of NPG1 as that of AdPG [27] such as the *worst-case* sublinear convergence of $\{x^k\}$ to an optimal solution of problem (P).

Nevertheless, one will see in the upcoming parts of the paper, we analyze the convergent results of NPG1 by designing a sufficient decrease inequality (in Corollary 3.10) without globally Lipschitz assumption on $\nabla f(x)$. This technique is different from that of [27, 28, 29]. To get this, in the sequel, we deploy the special properties of $\{t_k\}$ presented in the following lemma.

Lemma 3.6. *Let $\{t_k\}$ be a sequence of stepsizes generated by Algorithm 3.1. Then*

(i) $\{t_k\}$ is lower bounded by a positive number;

(ii) $\{t_k\}$ is convergent and has a positive limit.

Proof. (i) By Lemma 3.4 the set $T = \overline{conv}\left(\{x^0, x^1, ...\} \cup X^*\right)$ is closed and compact. From the local Lipschitz continuity of ∇f, it is easy to see that there exists $L_0 > 0$ satisfying $\|\nabla f(x) - \nabla f(y)\| \leq L_0\|x - y\| \quad \forall x, y \in T$. Thereafter, either $t_1 \geq \frac{c_1}{L_0}$ or $t_1 = (1 + \gamma_0')t_0 \geq t_0$. The induction process derives that

$$t_k \geq \min\{\frac{c_1}{L_0}, t_0\} = \eta > 0 \quad \forall k \geq 0. \qquad (3.24)$$

(ii) If we set $r_k = \ln t_{k+1} - \ln t_k$ and $r_k^+ = \max\{0, r_k\} \geq 0, r_k^- = -\min\{0, r_k\} \geq 0, \forall k \geq 0$ then $r_k = r_k^+ - r_k^-$. On the other hand, from Algorithm 3.1, we observe that $0 < c_1 < c_0 < \frac{1}{\sqrt{2}}$, hence both of (3.2) and (3.4) give

$$r_k = \ln \frac{t_{k+1}}{t_k} \leq \ln(1 + \gamma'_k) \leq \gamma'_k \leq \gamma_k \quad \forall k \geq 0.$$

Thus, $r_k^+ \leq \gamma_k$. Moreover, the series $\sum\limits_{k=0}^{+\infty} \gamma_k$ converges then $\sum\limits_{k=0}^{+\infty} r_k^+ < +\infty$. Noticeably,

$$\ln t_{k+1} - \ln t_0 = \sum_{i=0}^{k} r_i = \sum_{i=0}^{k} (r_i^+ - r_i^-) = \sum_{i=0}^{k} r_i^+ - \sum_{i=0}^{k} r_i^-. \tag{3.25}$$

Hence if the nonnegative series $\sum\limits_{k=0}^{+\infty} r_k^-$ diverges, i.e., $\lim\limits_{k \to +\infty} \sum\limits_{i=0}^{k} r_i^- = +\infty$ then

$$\lim_{k \to +\infty} (\ln t_{k+1}) = -\infty$$

which implies $\lim\limits_{k \to +\infty} t_k = 0$. This result is contradict with the assertion (i). Thus, $\sum\limits_{k=0}^{+\infty} r_k^-$ is convergent and therefore $\lim\limits_{k \to +\infty} t_k = t^* \in (0, +\infty)$ (followed by (3.25)). □

The result in the following lemma gives us an inequality like Lipschitz gradient continuity but with flexible constant for each pair of x^{k-1} and x^k.

Lemma 3.7. *There exists k^* such that*

$$\|\nabla f(x^k) - \nabla f(x^{k-1})\| \leq \frac{c_0}{t_{k-1}} \|x^k - x^{k-1}\|, \ \forall k \geq k^*. \tag{3.26}$$

Proof. Assuming that there is a subsequence $\{k_i\} \subset \mathbb{N}$, $k_i \to +\infty$ such that

$$\|\nabla f(x^{k_i}) - \nabla f(x^{k_i - 1})\| > \frac{c_0}{t_{k_i - 1}} \|x^{k_i} - x^{k_i - 1}\|.$$

By Algorithm 3.1, in this case we have

$$\frac{t_{k_i}}{t_{k_i - 1}} = \frac{c_1 \|x^{k_i} - x^{k_i - 1}\|}{t_{k_i - 1} \|\nabla f(x^{k_i}) - \nabla f(x^{k_i - 1})\|} < \frac{c_1}{c_0} \quad \forall k_i.$$

However, Lemma 3.6 gives

$$\lim_{k_i \to +\infty} t_{k_i} = \lim_{k_i \to +\infty} t_{k_i - 1} = \lim_{k \to +\infty} t_k = t^*.$$

Consequently, $\frac{t^*}{t^*} \leq \frac{c_1}{c_0} < 1$ that is impossible and we obtain the conclusion of the lemma.

□

Remark 3.8. From Lemma 3.7, we immediately obtain the increasing of the sequence $\{t_k\}_{k \geq k^*}$ and $0 < \eta < t_k \leq \max\{t_0, ..., t_{k^*-1}, t^*\} = t_{max}, \ k \geq 0$.

The next lemma plays a crucial role in proving the convergence of Algorithm 3.1 (NPG1).

Lemma 3.9. *For any $x \in \mathrm{int}(\mathrm{dom}(f))$, we have*

$$F(x) - F(x^{k+1}) \geq \frac{1 - c_0}{t_k} \|x^{k+1} - x^k\|^2 + \frac{1}{t_k} \langle x^k - x^{k+1}, x - x^k \rangle, \ \ \textit{for all } k \geq k^*. \tag{3.27}$$

Proof. Because of the convexity of f and Lemma 2.1 (ii) we have

$$\begin{aligned}
F(x) - F(x^{k+1}) &= f(x) + g(x) - f(x^{k+1}) - g(x^{k+1}) \\
&\geq f(x^k) + \left\langle x - x^k, \nabla f(x^k) \right\rangle + \left\langle x^{k+1} - x, \nabla f(x^k) + \frac{x^{k+1} - x^k}{t_k} \right\rangle - f(x^{k+1}) \\
&= f(x^k) - f(x^{k+1}) + \left\langle x^{k+1} - x^k, \nabla f(x^k) \right\rangle + \frac{1}{t_k} \left\langle x^{k+1} - x^k, x^{k+1} - x \right\rangle \\
&\geq \langle \nabla f(x^{k+1}) - \nabla f(x^k), x^k - x^{k+1} \rangle + \frac{1}{t_k} \left\| x^{k+1} - x^k \right\|^2 + \frac{1}{t_k} \left\langle x^{k+1} - x^k, x^k - x \right\rangle .
\end{aligned} \tag{3.28}$$

On the other hand, by using Lemma 3.7, we have the evaluation

$$\begin{aligned}
\left\langle \nabla f(x^{k+1}) - \nabla f(x^k), x^k - x^{k+1} \right\rangle &\geq - \left\| \nabla f(x^k) - \nabla f(x^{k+1}) \right\| \|x^k - x^{k+1}\| \\
&\geq - \frac{c_0}{t_k} \|x^{k+1} - x^k\|^2 \quad \forall k \geq k^*.
\end{aligned} \tag{3.29}$$

The proof is completed by utilizing (3.28) and (3.29). □

It is observed that if we substitute x by x^k in (3.27) of Lemma 3.9 and using Remark 3.8 we immediately get the following corollary known as a sufficient decrease type inequality.

Corollary 3.10 (sufficient decrease type inequality)**.** *For all $k \geq k^*$ we have*

$$F(x^k) - F(x^{k+1}) \geq \frac{1 - c_0}{t_k} \|x^{k+1} - x^k\|^2 \geq \frac{1 - c_0}{t^*} \|x^{k+1} - x^k\|^2 \geq 0, \quad \textit{for all } k \geq k^*. \tag{3.30}$$

Now we are ready to establish the convergent properties of Algorithm 3.1 (NPG1) in the following theorem.

Theorem 3.11 (the convergence of NPG1). *Suppose that problem (P) satisfies Assumptions 1 and 2. Then the following assertions hold for Algorithm 3.1.*

(i) The sequence $\{F(x^k)\}_{k \geq k^*}$ descends to $\lim\limits_{k \to +\infty} F(x^k) = F_*$.

(ii) The sequence $\{x^k\}$ converges to an optimal solution of problem (P).

(iii) For any $x^* \in X^*$ and $k \geq k^* + 1$ we have

$$F(x^k) - F_* = F(x^k) - F(x^*) \leq \frac{D}{2 t_{k^*} (k - k^*)} = O\left(\frac{1}{k}\right), \tag{3.31}$$

where

$$D = \max \left\{ \|x^* - x^{k^*}\|^2, \|x^* - x^{k^*}\|^2 + \frac{t^*(2c_0 - 1)}{1 - c_0} \left(F(x^{k^*}) - F_* \right) \right\}.$$

Proof. (i) By (3.30), the sequence $\{F(x^k)\}_{k \geq k^*}$ is decreasing. On the other hand, it is lower bounded by F_* hence converges to $\overline{F} \geq F_*$. Thus, $F(x^k) - F(x^{k+1}) \to 0$. And consequently, the inequality (3.30) follows

$$\lim_{k \to +\infty} \|x^{k+1} - x^k\| = 0. \tag{3.32}$$

Now, replacing x with x^* in (3.27) of Lemma 3.9 to obtain

$$\begin{aligned}
0 \leq F(x^{k+1}) - F(x^*) &\leq - \frac{1 - c_0}{t_k} \|x^{k+1} - x^k\|^2 - \frac{1}{t_k} \langle x^k - x^{k+1}, x^* - x^k \rangle \\
&\leq \frac{(c_0 - 1)\|x^{k+1} - x^k\|^2 + \|x^{k+1} - x^k\| \|x^k - x^*\|}{t_k}, \quad \text{for all } k \geq k^*.
\end{aligned} \tag{3.33}$$

However, $\{x^k\}$ is bounded (by Lemma 3.4(ii)) and $\lim\limits_{k \to +\infty} t_k = t^*$ (from Lemma 3.6) then combining with (3.32) we deduce that the limit of the right hand side of (3.33) is zero as k tending to infinity. Hence, again, by (3.33) we have

$\lim\limits_{k\to+\infty} F(x^k) = F_*$.

(ii) Taking into account that the sequence $\{x^k\}$ is bounded then for each cluster point $\overline{x}$ of $\{x^k\}$, we can take a subsequence $\{x^{k_i}\}$ such that $x^{k_i} \to \overline{x}$. On the other hand, the closedness of F (from *Assumption 1*) follows its lower semi-continuity and therefore $F(\overline{x}) \le \lim\limits_{k_i\to\infty} F(x^{k_i}) = F_*$, which implies $\overline{x} \in X^*$.

Setting $a_k = \|x^{k-1} - x^k\|^2 + \frac{2t_k^2}{t_{k-1}}\left(F(x^{k-1}) - F(x^*)\right) \ge 0$ and rewrite (3.22) to be

$$\|x^{k+1} - x^*\|^2 + a_{k+1} \le \|x^k - x^*\|^2 + a_k, \ \ \forall x^* \in X^*, \ \ k \ge k_1.$$

Moreover, we have just shown that all cluster points of $\{x^k\}$ belong to X^*. Therefore, applying Lemma 2.2 we obtain that $\{x^k\}$ converges to some element of X^*.

(iii) In (3.30), substituting k by j then summing up it from $j = k^*$ to k we derive that

$$F(x^{k^*}) - F(x^{k+1}) \ge \frac{1-c_0}{t^*}\sum_{j=k^*}^{k}\|x^{j+1} - x^j\|^2. \tag{3.34}$$

This indicates the convergence of $\sum\limits_{j=k^*}^{+\infty}\|x^{j+1} - x^j\|^2$ and

$$\sum_{j=k^*}^{+\infty}\|x^{j+1} - x^j\|^2 \le \frac{t^*}{1-c_0}\left(F(x^{k^*}) - F_*\right). \tag{3.35}$$

Applying (3.27) again, we obtain that

$$\begin{aligned} F(x^*) - F(x^{j+1}) &\ge \frac{1}{2t_j}\left(\|x^{j+1} - x^j\|^2 + 2\left\langle x^j - x^{j+1}, x^* - x^j\right\rangle\right) + \left(\frac{1}{2} - c_0\right)\frac{\|x^j - x^{j+1}\|^2}{t_j} \\ &\ge \frac{1}{2t_j}\left(\|x^* - x^{j+1}\|^2 - \|x^* - x^j\|^2\right) + \left(\frac{1}{2} - c_0\right)\frac{\|x^j - x^{j+1}\|^2}{t_j} \quad \forall j \ge k^*. \end{aligned} \tag{3.36}$$

On the other hand, Remark 3.8 gives $t_j \ge t_{k^*}$ $\forall j \ge k^*$ which helps to infer the following inequality from (3.36)

$$\begin{aligned} 2t_{k^*}\left(F(x^{j+1}) - F(x^*)\right) &\le 2t_j\left(F(x^{j+1}) - F(x^*)\right) \\ &\le \left(\|x^* - x^j\|^2 - \|x^* - x^{j+1}\|^2\right) + (2c_0 - 1)\|x^j - x^{j+1}\|^2 \quad \forall j \ge k^*. \end{aligned} \tag{3.37}$$

Summing (3.37) side by side for $j = k^*$ to $k + k^* - 1$ $(k \ge 1)$, we get that

$$\begin{aligned} 2t_{k^*}\left(\sum_{j=k^*}^{k+k^*-1} F(x^{j+1}) - kF(x^*)\right) &\le \left(\|x^* - x^{k^*}\|^2 - \|x^* - x^{k+k^*}\|^2\right) + \\ &\quad + (2c_0 - 1)\sum_{j=k^*}^{k+k^*-1}\|x^j - x^{j+1}\|^2 \\ &\le D, \end{aligned} \tag{3.38}$$

where, (from (3.35))D is defined by

$$D = \max\left\{\|x^* - x^{k^*}\|^2, \|x^* - x^{k^*}\|^2 + \frac{t^*(2c_0 - 1)}{1 - c_0}\left(F(x^{k^*}) - F_*\right)\right\}.$$

Additionally, the descent of $\{F(x^k)\}_{k\ge k^*}$ induces $\sum\limits_{j=k^*}^{k+k^*-1} F(x^{j+1}) \ge kF(x^{k+k^*})$. Therefore by (3.38), we have

$$F(x^{k+k^*}) - F(x^*) \le \frac{1}{2t_{k^*}}\frac{D}{k} \quad \forall k \ge 1,$$

which means that $F(x^k) - F(x^*) \le \dfrac{D}{2t_{k^*}}\dfrac{1}{k - k^*} = O\left(\dfrac{1}{k}\right) \quad \forall k \ge k^* + 1.$

□

The last result in this section, we prove a stronger convergence rate of Algorithm 3.1 if f is locally strongly convex. The detail is as follows.

Theorem 3.12. *Assuming that $c_0 \leq \frac{1}{2}$ and problem (P) satisfies Assumption 1 and Assumption 2. Additionally, f is locally strongly convex then the sequence $\{x^k\}$ generated by Algorithm 3.1 satisfies*

$$\|x^{k+1} - x^*\|^2 \leq (1 - \sigma t_{k^*})\|x^k - x^*\|^2, \ \ \forall k \geq k^*, \tag{3.39}$$

where $\sigma > 0$ is strong convexity constant of f on the compact set $T = \overline{conv}\left(\{x^0, x^1, ...\} \cup X^*\right)$. Consequently, this result shows the Q-linear convergence rate of $\{x^k\}$.

Proof. The $\sigma-$ strong convexity on T of f implies that

$$f(x) - f(x^k) \geq \langle \nabla f(x^k), x - x^k \rangle + \frac{\sigma}{2}\|x - x^k\|^2, \ \forall x \in T.$$

We update this change and the condition $c_0 \leq \frac{1}{2}$ in the argument of formula (3.28) and (3.36) to obtain the following inequality

$$F(x^*) - F(x^{k+1}) \geq \frac{1}{2t_k}\|x^* - x^{k+1}\|^2 + \left(\frac{\sigma}{2} - \frac{1}{2t_k}\right)\|x^* - x^k\|^2,$$

for all $x^* \in X^*, k \geq k^*$, Remember that $F(x^*) - F(x^{k+1}) \leq 0 \ \forall k$ hence

$$\frac{1}{2t_k}\|x^* - x^{k+1}\|^2 \leq \left(\frac{1}{2t_k} - \frac{\sigma}{2}\right)\|x^* - x^k\|^2, \ \ k \geq k^*. \tag{3.40}$$

By (3.40), Lemma 3.6(i) and Remark 3.8, we have: $\forall k \geq k^*$

$$0 < 1 - \sigma t_k \leq 1 - \sigma t_{k^*} \leq 1 - \sigma\eta < 1,$$

which derives

$$\|x^{k+1} - x^*\|^2 \leq (1 - \sigma t_{k^*})\,\|x^k - x^*\|^2, \ \ k \geq k^*.$$

The last inequality aims the Q-linear convergence rate of $\{x^k\}$. □

4. For a class of the nonconvex case of problem (P)

We now consider problem (P) satisfying *Assumption 1* and other conditions in *Assumption 3* below

Assumption 3. *(i) f has a globally Lipschitz gradient with constant L_f on $\mathrm{int}(\mathrm{dom}(f))$.*

(ii) For $u, v \in \mathrm{int}(\mathrm{dom}(f))$, the function $h_{uv} : [0,1] \to \mathbb{R}$ defined by

$$h_{uv}(t) = f'_t(u + t(v - u)) = \langle \nabla f(u + t(v - u)), v - u \rangle$$

is quasiconvex.

Example 4.1. Suppose that f is either convex or concave. Then f satisfies *Assumption 3 (ii)*. Indeed, the convexity (concavity, resp.) of f follows the convexity (concavity, resp.) of $f(u + t(v - u))$ on the set $\{t \in \mathbb{R} \mid u + t(v - u) \in \mathrm{int}(\mathrm{dom}(f))\} \supset [0, 1]$ (since $\mathrm{int}(\mathrm{dom}(f))$ is convex). As a result, $f'_t(u + t(v - u))$ is increasing (decreasing, resp.) monotone over $[0, 1]$ and therefore quasiconvex on that. In the case where f is a concave function then $F = f + g$ is actually the difference of the two convex functions, or in other words, F belongs to the class of *dc functions*.

Example 4.2. The indefinite quadratic function $f(x) = \frac{1}{2}x^T Ax + b^T x$ (A is a symmetric matrix in $\mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$) satisfies both of *Assumption 1* and *Assumption 3* since $h_{uv}(t) = \langle A(u + t(v - u)) + b, v - u\rangle$ is linear and hence quasiconvex on $[0, 1]$ for any $u, v \in \mathrm{int}(\mathrm{dom}(f)) = \mathbb{R}^n$.

From Example 4.1 and 4.2, we see that the class of problem (P) satisfying *Assumption 1* and *Assumption 3* is nonconvex in general. Subsequently, we propose an other version of Algorithm 3.1 that can be applied for such a kind of problems.

Algorithm 4.1 (NPG2)

Step 0 (Initialization). Select $t_0 > 0$, $0 < c_1 < c_0 < 1$, $x^0 \in \operatorname{int}(\operatorname{dom}(f))$ a tolerance $\epsilon > 0$ and a positive real sequence $\{\gamma_k\}$ such that $\sum\limits_{k=0}^{+\infty} \gamma_k < \infty$. Taking $x^1 = \operatorname{Prox}_{t_0 g}(x^0 - t_0 \nabla f(x^0)), t_{-1} = t_0$ and $k = 1$.

Step 1.

$$\begin{aligned}
&\textbf{If} \quad \|\nabla f(x^k) - \nabla f(x^{k-1})\| > \frac{c_0}{t_{k-1}} \|x^k - x^{k-1}\| \\
&\textbf{then} \quad t_k = c_1 \frac{\|x^k - x^{k-1}\|}{\|\nabla f(x^k) - \nabla f(x^{k-1})\|} && (4.1)\\
&\textbf{else} \quad \gamma'_{k-1} = \gamma_{k-1} \\
&\qquad \text{if } \frac{t_{k-1}}{t_{k-2}} < 1 \text{ then } \gamma'_{k-1} = \min\left\{\gamma_{k-1}, \sqrt{1 + \frac{t_{k-1}}{t_{k-2}}} - 1\right\} && (4.2)\\
&\qquad t_k = (1 + \gamma'_{k-1}) t_{k-1}.
\end{aligned}$$

Step 2. Compute $x^{k+1} = \operatorname{Prox}_{t_k g}(x^k - t_k \nabla f(x^k))$.
Step 3. If $\|x^{k+1} - x^k\| < \epsilon$ **then** STOP **else** setting $k := k + 1$ and return to **Step 1**.

Due to the lack of the convexity of f, the analysis on the convergence of our method in the sequel focus on showing the iterates tending to a stationary point of (P). In particular, we will show $\sum\limits_{k=0}^{+\infty} \|x^{k+1} - x^k\|^2$ is convergent in Theorem 4.6. We first start with some preparing lemmas in the sequel.

Lemma 4.3. *The sequence $\{t_k\}$ in Algorithm 4.1 satisfies $\inf\limits_{k \geq 0} t_k > 0$ and has a positive limit.*

Proof. Similarly to Lemma 3.6 (i), it is clear that $t_k \geq \min\{t_0, \frac{c_1}{L_f}\} > 0$ for all $k \geq 0$. As a result, $\inf\limits_{k \geq 0} t_k > 0$. The remaining conclusion is shown as Lemma 3.6 (ii). □

Lemma 4.4. *For Algorithm 4.1, there exists $\overline{k}$ such that*

$$\|\nabla f(x^k) - \nabla f(x^{k-1})\| \leq \frac{c_0}{t_{k-1}} \|x^k - x^{k-1}\| \;\; \forall k \geq \overline{k}.$$

Proof. The proof is similar to that of Lemma 3.7. □

The following lemma presents the sufficient decrease type inequality - a key step to obtain the convergence results of our algorithms.

Lemma 4.5. *Assuming that problem (P) satisfies Assumption 1 and Assumption 3 then the sequence $\{x^k\}$ generated by Algorithm 4.1 has the following property*

$$F(x^k) - F(x^{k+1}) \geq \frac{1 - c_0}{t_k} \|x^{k+1} - x^k\|^2, \; \forall k \geq \overline{k}.$$

Proof. Invoking the Fundamental Theorem of Calculus, we have

$$\begin{aligned}
f(x^{k+1}) - f(x^k) &= \int_0^1 \left\langle \nabla f(x^k + t(x^{k+1} - x^k)), x^{k+1} - x^k \right\rangle dt \\
&= \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \int_0^1 u_k(t) dt, \; \forall k \geq \overline{k} \qquad (4.3)
\end{aligned}$$

where

$$\begin{aligned}
u_k(t) &= \langle \nabla f(x^k + t(x^{k+1} - x^k)) - \nabla f(x^k), x^{k+1} - x^k \rangle \\
&= h_{x^k x^{k+1}}(t) - \langle \nabla f(x^k), x^{k+1} - x^k \rangle.
\end{aligned}$$

According to *Assumption 3*, the quasiconvexity of $u_k(t)$ in $[0,1]$ follows that

$$\begin{aligned} u_k(t) &\leq \max\{u_k(0), u_k(1)\} = \max\{0, u_k(1)\} \leq |u_k(1)| \\ &= |\langle \nabla f(x^{k+1}) - \nabla f(x^k), x^{k+1} - x^k \rangle|, \ \forall t \in [0,1]. \end{aligned}$$

Thereafter, using Lemma 4.4, we derive that

$$\int_0^1 u_k(t)dt \leq \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2, \ \ \forall k \geq \overline{k}. \tag{4.4}$$

Now, combining (4.3), (4.4) and Lemma 2.1(ii) with $x = x^{k+1}$ we get that

$$\begin{aligned} F(x^k) - F(x^{k+1}) &= f(x^k) - f(x^{k+1}) + g(x^k) - g(x^{k+1}) \\ &\geq -\left\langle x^{k+1} - x^k, \nabla f(x^k) \right\rangle - \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \\ &\qquad + \left\langle x^{k+1} - x^k, \nabla f(x^k) + \frac{x^{k+1} - x^k}{t_k} \right\rangle \\ &= \frac{1 - c_0}{t_k}\|x^{k+1} - x^k\|^2 \ \ \forall k \geq \overline{k}. \end{aligned} \tag{4.5}$$

□

The following theorem provides the convergence results of Algorithm 4.1 (NPG2) under Assumption 1 and Assumption 3 for solving the problem (P).

Theorem 4.6. *Under Assumptions 1 and 3, the following assertions hold for Algorithm 4.1:*

(i) The sequence $\{F(x^k)\}_{k \geq \overline{k}}$ is decreasing and for any $k \geq \overline{k}$, $F(x^{k+1}) < F(x^k)$ unless x^k is a stationary point of problem (P).

(ii) $F(x^k) - F(x^{k+1}) \to 0$ and

$$\min_{\overline{k} \leq k \leq K} \|x^{k+1} - x^k\|^2 \leq \frac{t^*(F(x^{\overline{k}}) - \hat{F})}{(K - \overline{k})(1 - c_0)} = O\left(\frac{1}{\sqrt{K}}\right) \ \ \forall K \geq \overline{k}.$$

(iii) $\sum\limits_{k=0}^{+\infty} \|x^{k+1} - x^k\|^2$ is convergent.

Proof. (i) By (4.5) and $c_0 < 1$, it is clear to see that $F(x^k) \geq F(x^{k+1})$ for all $k \geq \overline{k}$. If $F(x^k) = F(x^{k+1})$ then $x^{k+1} = x^k = \mathrm{Prox}_{t_k g}(x^k - t_k \nabla f(x^k))$ meaning x^k is a stationary point of (P).

(ii) Since problem (P) has a non-empty optimal solution set then the sequence $\{F(x^k)\}_{k \geq \overline{k}}$ is decreasing and lower bounded by F_*. This follows the existence of a finite limit $\hat{F}$ of $\{F(x^k)\}_{k \geq \overline{k}}$ ($\hat{F} \geq F_*$). It means that $F(x^k) - F(x^{k+1}) \to 0$. Moreover, by Lemma 4.3 we have $\{t_k\}_{k \geq \overline{k}}$ increasing to $\lim\limits_{k \to +\infty} t_k = t^*$. On the other hand, inequality (4.5) indicates that

$$\|x^{k+1} - x^k\|^2 \leq \frac{t_k}{1 - c_0}(F(x^k) - F(x^{k+1})) \leq \frac{t^*}{1 - c_0}(F(x^k) - F(x^{k+1})) \ \ \forall k \geq \overline{k}. \tag{4.6}$$

Therefore by summing (4.6) from $k = \overline{k}$ to K we obtain that

$$\sum_{k=\overline{k}}^{K} \|x^{k+1} - x^k\|^2 \leq \frac{t^*}{1 - c_0}(F(x^{\overline{k}}) - F(x^{K+1})) \leq \frac{t^*}{1 - c_0}(F(x^{\overline{k}}) - \hat{F}) \ \ \forall K \geq \overline{k} \tag{4.7}$$

which implies that

$$\min_{\overline{k} \leq k \leq K} \|x^{k+1} - x^k\|^2 \leq \frac{t^*(F(x^{\overline{k}}) - \hat{F})}{(K - \overline{k})(1 - c_0)} = O\left(\frac{1}{\sqrt{K}}\right) \ \ \forall K \geq \overline{k}. \tag{4.8}$$

(iii) It is followed directly from (4.7) that $\sum_{k=\bar{k}}^{+\infty} \|x^k - x^{k+1}\|^2 \leq F(x^{\bar{k}}) - \hat{F}$ and we obtain the desired conclusion.

□

Remark 4.7. (i) Remember that $c_0, c_1 \in \left(0, \frac{1}{\sqrt{2}}\right)$ for Algorithm 3.1 (NPG1) but $c_0, c_1 \in (0,1)$ for Algorithm 4.1 (NPG2).

(ii) Actually, the command (4.2) in Algorithm 4.1 is optional since we do not need it during the proof of the convergence of NPG2. However, through out the implementation for numerical experiments we realize that this step helps the performance of the algorithm be better.

5. Problem (P) with quadratic function f

In this section, we propose an extension of NPG2 called *NPG-quad* solving problem (P) with the quadratic function f, i.e., $f(x) = \frac{1}{2}x^T Ax + b^T x$ as described in Example 4.2. The changes compared with NPG2 are in the two points:

(i) Firstly, c_0, c_1 in $(0, 2)$ (while c_0, c_1 in $(0, 1)$ for NPG2);

(ii) Secondly,

$$t_k(\text{ in (5.2)}) = \frac{c_1\|x^k - x^{k-1}\|^2}{(x^k - x^{k-1})^T A(x^k - x^{k-1})} \geq \frac{c_1\|x^k - x^{k-1}\|}{\|Ax^k - Ax^{k-1}\|} = t_k(\text{ in (4.1)}).$$

These probably make the stepsize of NPG-quad larger and therefore the execution time of it shorter in comparison with NPG1 and NPG2 .

Algorithm 5.1 (NPG-quad)

Step 0 (Initialization). Select $t_0 > 0$, $0 < c_1 < c_0 < 2$, $x^0 \in \text{dom}(g)$, a tolerance $\epsilon > 0$ and a positive real sequence $\{\gamma_k\}$ such that $\sum_{k=0}^{+\infty} \gamma_k < +\infty$. Taking $x^1 = \text{Prox}_{t_0 g}(x^0 - t_0 \nabla f(x^0))$, $t_{-1} = t_0$, and $k = 1$.

Step 1.

$$\textbf{If} \quad (x^k - x^{k-1})^T A(x^k - x^{k-1}) > c_0 \frac{\|x^k - x^{k-1}\|^2}{t_{k-1}} \tag{5.1}$$

$$\textbf{then} \quad t_k = \frac{c_1\|x^k - x^{k-1}\|^2}{(x^k - x^{k-1})^T A(x^k - x^{k-1})} \tag{5.2}$$

$$\textbf{else} \quad \gamma'_{k-1} = \gamma_{k-1}$$

$$\text{if } \frac{t_{k-1}}{t_{k-2}} < 1 \text{ then } \gamma'_{k-1} = \min\left\{\gamma_{k-1}, \sqrt{1 + \frac{t_{k-1}}{t_{k-2}}} - 1\right\} \tag{5.3}$$

$$t_k = (1 + \gamma'_{k-1})t_{k-1}. \tag{5.4}$$

Step 2. Compute $x^{k+1} = \text{Prox}_{t_k g}(x^k - t_k \nabla f(x^k))$.
Step 3. If $\|x^{k+1} - x^k\| < \epsilon$ **then** STOP **else** setting $k := k + 1$ and return to **Step 1**.

Lemma 5.1. *The sequence $\{t_k\}$ generated by Algorithm 5.1 has a positive limit.*

Proof. Analogous to former sections, we are easy to have $t_k \geq \min\left\{t_0, \frac{c_1}{\|A\|}\right\} > 0$ for all $k \geq 0$. Therefore, $\inf_{k \geq 0} t_k > 0$. The computation of t_k by (5.2) or (5.4) provides $\ln\left(\frac{t_{k+1}}{t_k}\right) < \ln(1 + \gamma_k)$. The subsequent arguments are akin to the one of Lemma 3.6 (ii). □

Lemma 5.2. *For Algorithm 5.1, there exists $\tilde{k}$ such that*

$$(x^k - x^{k-1})^T A(x^k - x^{k-1}) \leq c_0 \frac{\|x^k - x^{k-1}\|^2}{t_{k-1}}, \quad \textit{for all } k \geq \tilde{k}. \tag{5.5}$$

Proof. Based on the properties of $\{t_k\}$ in Lemma 5.1 and arguing by contradiction as Lemma 3.7 we have the desired conclusion. □

The last result of the paper is on the convergence of Algorithm 5.1 in the following theorem.

Theorem 5.3. *Supposing problem (P) satisfies Assumption 1 and f has a quadratic form as in Example 4.2. For $\{x^k\}$ generated by Algorithm 5.1, the sequence $\{F(x^k)\}_{k\geq \tilde{k}}$ is decreasing to a finite limit $\tilde{F} \geq F_*$ and $\sum\limits_{k=0}^{+\infty} \|x^{k+1} - x^k\|^2$ is convergent. Additionally,*

$$\min_{\tilde{k}\leq k\leq K} \|x^{k+1} - x^k\|^2 \leq \frac{t^*(F(x^{\tilde{k}}) - \tilde{F})}{(K-\tilde{k})(1-\frac{c_0}{2})} = O\left(\frac{1}{\sqrt{K}}\right) \;\; \forall K \geq \tilde{k}.$$

Proof. We have

$$\begin{aligned} f(x^{k+1}) - f(x^k) &= \int_0^1 \left\langle \nabla f(x^k + t(x^{k+1}-x^k)), x^{k+1}-x^k \right\rangle dt \\ &= \int_0^1 \left\langle A(x^k + t(x^{k+1}-x^k)) + b, x^{k+1}-x^k \right\rangle dt \\ &= \left\langle A(x^{k+1}-x^k), x^{k+1}-x^k \right\rangle \int_0^1 tdt + \left\langle Ax^k + b, x^{k+1}-x^k \right\rangle \\ &= \frac{1}{2}(x^{k+1}-x^k)^T A(x^{k+1}-x^k) + \left\langle \nabla f(x^k), x^{k+1}-x^k \right\rangle. \end{aligned} \tag{5.6}$$

Now plugging (5.6) in $F(x^k) - F(x^{k+1})$ and using Lemma 2.1(ii) to obtain

$$\begin{aligned} F(x^k) - F(x^{k+1}) &= f(x^k) - f(x^{k+1}) + g(x^k) - g(x^{k+1}) \\ &\geq -\frac{1}{2}(x^{k+1}-x^k)^T A(x^{k+1}-x^k) - \langle \nabla f(x^k), x^{k+1}-x^k\rangle + \\ &\qquad + \left\langle x^{k+1}-x^k, \nabla f(x^k) + \frac{x^{k+1}-x^k}{t_k} \right\rangle \\ &= -\frac{1}{2}(x^{k+1}-x^k)^T A(x^{k+1}-x^k) + \frac{1}{t_k}\|x^{k+1}-x^k\|^2. \end{aligned} \tag{5.7}$$

Next, applying Lemma 5.2 for (5.7) we obtain for all $k \geq \tilde{k}$,

$$F(x^k) - F(x^{k+1}) \geq \left(1 - \frac{c_0}{2}\right) \frac{\|x^{k+1}-x^k\|^2}{t_k}. \tag{5.8}$$

The remaining arguments are similar to those of Theorem 4.6. □

Remark 5.4. If f is a concave quadratic function i.e., A is negative semi-definite then the condition (5.1) is false, hence

- $\tilde{k}$ in Lemma 5.2 should be zero;
- t_k is always defined by formula (5.4) and $\{t_k\}_{k\geq 0}$ is increasing to a finite limit;
- the evaluation (5.8) should be

$$F(x^k) - F(x^{k+1}) \geq \frac{\|x^{k+1}-x^k\|^2}{t_k}, \; \forall k \geq 0. \tag{5.9}$$

6. Numerical experiments

In this section, we investigate the performance of our new stepsize for the proximal gradient scheme by comparing NPG1 (Algorithm 3.1), NPG2 (Algorithm 4.1) and NPG-quad (Algorithm 5.1) with the recent algorithms including:

- the AdPG proposed by Malitksy and Mishchenko [27] (Algorithm 3 in [27]);
- the AdaPG^{q,r} from Latafat et al. in [29] using $(q,r) = \left(\frac{3}{2}, \frac{3}{4}\right)$;

- the proximal gradient algorithms with stepsize selection based on an improved version of Armijo's backtracking procedure[1], denoted by ProxGD(s, r) where (s, r) equals $(1.1, 0.5)$ or $(1.2, 0.5)$.

For our algorithms, we use the convergent series $\sum_{k=0}^{+\infty} \gamma_k$ defined by

$$\gamma_{k-1} = \frac{0.1(\ln k)^{5.7}}{k^{1.1}}, \quad \forall k \geq 1,$$

and setting $(c_0, c_1) = (0.7, 0.69)$ for NPG1 , $(c_0, c_1) = (0.99, 0.98)$ for NPG2 and NPG-quad .

The AdPG, ProxGD$(1.1, 0.5)$ and ProxGD$(1.2, 0.5)$ are the top three algorithms in the experiments conducted in [27], with AdPG notably outperforming all others. At the same time, the numerical simulations of [29] indicate that AdaPG$^{\left(\frac{3}{2},\frac{3}{4}\right)}$ is the most effective one for almost cases reported in [29]. Considering all of the above, we believe this section includes a comprehensive comparison of the most effective algorithms in the literature.

We conduct experiments on five typical composite type optimization problems with various sizes for each one. The average results on 10 randomly generated data for each size of considered problems are reported with respect to

1. the number of iterations (*Iter.*);
2. $\|x^{k+1} - x^k\|$ (*Res.*);
3. $F(x^k) - F_*$ (*Obj.*), where F_* is computed as the minimum of $F(x^k)$ over all iterations and all tested algorithms;
4. the running time in seconds (*Time(s)*).

For all implemented algorithms, the stopping criterion is either the residual $\|x^{k+1} - x^k\| \leq 1e - 06$ or the number of iterations over N_{max}. The detailed information is on Tables 1, 2, 3, 4, 5. We emphasize the best results among all by bold characters and the worst results by italic type. We also choose one arbitrary data for each kind of problems to illustrate the performance by Figures 1, 2, 3, 4, 5.

All experiments[2] were implemented in Python and executed on a personal computer equipped with a 12th Gen Intel(R) Core(TM) i7-1260P 2.10 GHz processor, RAM 16.0 GB.

6.1. Lasso problems

The formulation of Lasso problem is formulated as the ℓ_1 regularized least squares

$$\min_{x \in \mathbb{R}^n} \quad \frac{1}{2}\|Ax - b\|^2 + \lambda\|x\|_1, \tag{Lasso}$$

where $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$. The applications of Lasso can be found in statistic, machine learning, signal processing, see e.g., [4, 7, 30]. By using the similar rules in [30], we randomly generate $A \in \mathbb{R}^{m \times n}$ with entries drawn from the normal distribution $\mathcal{N}(0, 1)$. We then construct a sparse solution x^* with 5% approximately non-zero entries, drawn from a mixture distribution $\mathcal{N}(0, 1) \times B(1, 0.05)$ then setting $b = Ax^* + \delta$, where δ is white Gaussian noise with variance 0.01. The regularization term $\lambda = 0.01\|A^T b\|_\infty$. Obviously, Lasso satisfies *Assumptions 1, 2, 3* then both of NPG1 and NPG2 are available for it. Moreover, f is quadratic hence NPG-quad can be applied for solving this problem formally. Figure 1 illustrates the performance of mentioned algorithms for one of randomly generated data with $m = 2048$, $n = 8192$. The obtained average results in Table 1 show the best performance of NPG-quad for almost dimensions of Lasso.

[1] For $s > 1, r < 1$, Armijo's linesearch in finds the largest $t_k = sr^i t_{k-1}$ for $i = 0, 1, ...$ such that $f(x^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \frac{1}{2t_k}\|x^{k+1} - x^k\|^2$.

[2] All codes are available at our repository https://github.com/hoaiphamthi/NPG-for-composite-models.

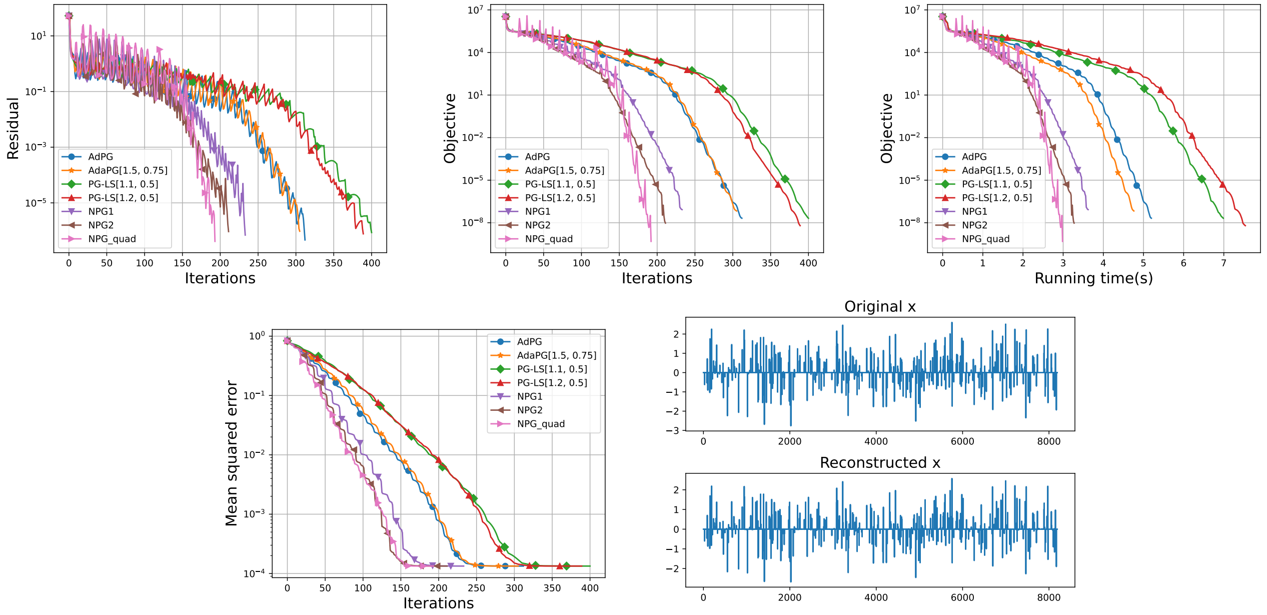

Figure 1: Illustration for one of randomly generated data of Lasso with size $m = 2048, n = 8192$.

Size		Metrics	Average of all datasets						
m	n		AdPG	AdaPG$^{(\frac{3}{2},\frac{3}{4})}$	PG-LS (1.1, 0.5)	PG-LS (1.2, 0.5)	NPG1	NPG2	NPG-quad
512	1024	Iter.	114,4	115,4	*146,7*	138,4	92,1	85,4	**79,7**
		Res.	7,95E-07	*8E-07*	6,76E-07	6,29E-07	7,99E-07	7,16E-07	**6,25E-07**
		Obj.	1,07E-10	1,08E-10	7,05E-11	7,52E-11	*3,45E-10*	1,05E-10	**1,03E-11**
		Time(s)	0,042869	0,039073	0,051572	*0,052111*	0,028284	0,026564	**0,025459**
512	2048	Iter.	307,7	306,2	*402,9*	381,5	235,7	**197,6**	204,7
		Res.	7,26E-07	7,1E-07	8,05E-07	*8,55E-07*	6,1E-07	7,29E-07	**4,25E-07**
		Obj.	*8,72E-09*	5,03E-09	2,72E-09	4,35E-09	7,24E-09	4,99E-09	**9,19E-11**
		Time(s)	0,123599	0,134157	0,215685	*0,25036*	0,129324	**0,107936**	0,119032
512	4096	Iter.	5923,4	5478,9	*8311,4*	8269,7	5690	4534,1	**3066,5**
		Res.	9,65E-07	9,52E-07	9,68E-07	9,43E-07	*9,8E-07*	9,73E-07	**9,22E-07**
		Obj.	6,5E-06	7,08E-06	1,2E-06	5,69E-07	*9,81E-06*	5,56E-06	**6,86E-08**
		Time(s)	6,289767	6,476083	10,43385	*12,00841*	6,641218	5,188084	**3,442947**
1024	2048	Iter.	118,8	123,2	*153,6*	144,8	102	90,9	**89,6**
		Res.	*8,18E-07*	7,58E-07	6,45E-07	5,9E-07	7,94E-07	7,82E-07	**5,68E-07**
		Obj.	3,23E-10	2,3E-10	1,97E-10	1,55E-10	*9,11E-10*	2,64E-10	**3,34E-11**
		Time(s)	**0,094191**	0,101175	0,152634	*0,177035*	0,105597	0,105269	0,109908
1024	4096	Iter.	282,6	282	*366,6*	342,2	221,7	**187,7**	188,8
		Res.	7,57E-07	6,93E-07	*9,1E-07*	7,5E-07	7,24E-07	7,46E-07	**5,91E-07**
		Obj.	1,13E-08	5,93E-09	4,27E-09	6E-09	*1,89E-08*	1,11E-08	**9,4E-11**
		Time(s)	0,942224	1,029607	1,429591	*1,506059*	0,769672	**0,657381**	0,675595
1024	8192	Iter.	5422,5	5197,2	*7953*	7839,9	5431,7	4345,8	**2967,5**
		Res.	**9,42E-07**	9,69E-07	9,7E-07	9,45E-07	9,61E-07	*9,78E-07*	9,43E-07
		Obj.	1,76E-05	1,49E-05	2,34E-06	1,65E-06	*1,84E-05*	1,14E-05	**4,27E-07**
		Time(s)	45,3179	43,95711	78,10607	*83,31711*	44,58026	36,09447	**25,52683**
2048	4096	Iter.	107	111,5	*135,6*	129,3	97,5	86,6	**79,2**
		Res.	7,76E-07	*8,06E-07*	7,48E-07	7,19E-07	7,43E-07	7,57E-07	**5,46E-07**
		Obj.	4,13E-10	3,98E-10	5,13E-10	3,07E-10	*1,37E-09*	3,69E-10	**1,16E-10**
		Time(s)	1,008659	1,057126	1,350391	*1,420677*	0,856957	0,763448	**0,698618**
2048	8192	Iter.	289,1	288,2	*380,7*	361,1	226,8	199,6	**183,5**
		Res.	7,52E-07	7,46E-07	7,85E-07	*8,42E-07*	7,67E-07	7,11E-07	**5,15E-07**
		Obj.	*3,93E-08*	3,5E-08	1,31E-08	1,33E-08	3,65E-08	2,58E-08	**5,18E-10**
		Time(s)	5,192244	5,059049	7,395414	*7,782476*	3,8797	3,397991	**3,145621**

Table 1: Average results for Lasso problem ($N_{max} = 15000$).

6.2. Minimum length piecewise-linear curve subject to equality constraints

We consider an other optimization problem from [31, Example 10.4], where the objective is minimizing the length of the piecewise-linear curve connecting the points $(0,0),(1,x_1),...,(n,x_n)$ such that $Ax=b$, where $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$. The problem therefore can be formed as

$$\min \sqrt{1+x_1^2}+\sum_{i=1}^{n-1}\sqrt{1+(x_{i+1}-x_i)^2} \quad \text{s.t.} \quad Ax=b. \tag{Min-length}$$

It is seen that Min-length[3] satisfies *Assumption 1,2,3* and we can use NPG1 and NPG2 to solve it exactly. In the implementation, all members of A are randomly generated by normal distribution $\mathcal{N}(0,1)$. Taking $b=Ax^*$, where $x^* \sim \mathcal{N}(0,1)$. Figure 2 provides the line graphs of one randomly generated data with $m=2000, n=10000$. Table 2 includes the average computation results for various sizes of Min-length. Notably, both NPG1 and NPG2 outperform the remaining ones with the big deviation in term of computational time, residual, objective value and the number of iterations. The speed of NPG1 can be seen as the best among all for Min-length.

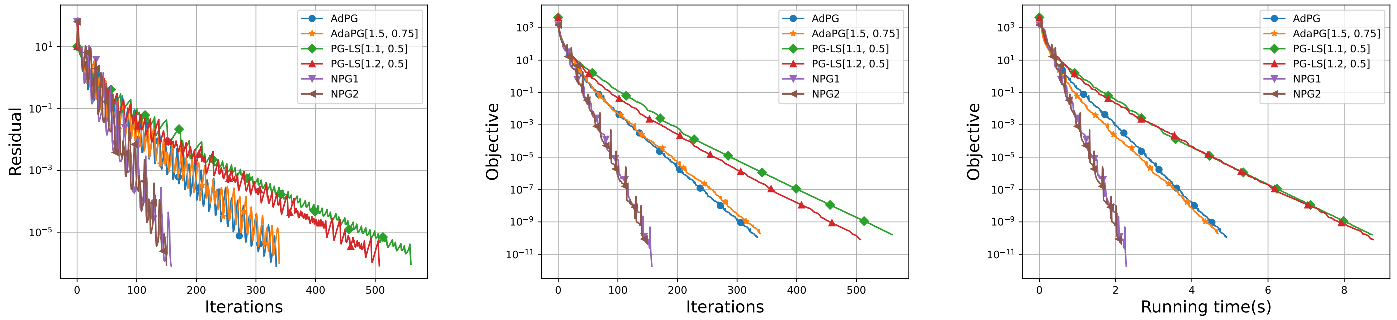

Figure 2: Illustrations for one of randomly generated data of Min-length with $m=2000, n=10000$.

6.3. Dual of the entropy maximization problems

We consider the entropy maximization problem subject to linear constraints [31, Section 5.1.6] which is

$$\min \sum_{i=1}^{n} x_i \log x_i \quad \text{s.t.} \quad Ax \leq b, \ \sum_{i=1}^{n} x_i = 1, \ \text{ and } \ x_i > 0, i=1,...,n, \tag{6.1}$$

where $A=[a^1,a^2,...,a^n] \in \mathbb{R}^{m\times n}$, with $a^i \in \mathbb{R}^m$ is the $i-$th column of A and $b \in \mathbb{R}^m$. Its dual problem is

$$\min e^{-\mu-1}\sum_{i=1}^{n} e^{-(a^i)^T\lambda}+b^T\lambda+\mu, \text{ s.t. } \lambda \in \mathbb{R}^m_+, \ \mu \in \mathbb{R}. \tag{Dual-max-entropy}$$

It is observed that Problem Dual-max-entropy[4] matches *Assumption 1, 2* but *Assumption 3*. Therefore the use of NPG1 is straightforward for it. We still run NPG2 for Dual-max-entropy as a heuristic approach. We use the similar rule of generating data as [27]. Specifically, a $m \times n$ matrix A with entries are generated from $\mathcal{N}(0,1)$, $b=Ax^*$ with a ℓ_1-normalized x^* sampled from the uniform distribution $\mathcal{U}[0.1,1)$. Results are depicted in Table 3 and Figure 3. It is shown that the performance of NPG2 is more significantly efficient than the remaining ones.

[3] Min-length is a case of problem (P) with $f(x)=\sqrt{1+x_1^2}+\sum_{i=1}^{n-1}\sqrt{1+(x_{i+1}-x_i)^2}$ and $g(x)=\iota_C$ (the indicator function of C) with $C=\{x\in\mathbb{R}^n \mid Ax=b\}$.

[4] Dual-max-entropy is a case of problem (P) with $f(\lambda,\mu)=e^{-\mu-1}\sum_{i=1}^{n}e^{-(a^i)^T\lambda}+b^T\lambda+\mu$ and $g(\lambda,\mu)=\iota_C$ (the indicator function of C) with $C=\mathbb{R}^m_+\times\mathbb{R}$ and ∇f does not global Lipschitz on C.

Size		Metrics	Average of all datasets					
m	n		AdPG	AdaPG$^{(\frac{3}{2},\frac{3}{4})}$	PG-LS (1.1, 0.5)	PG-LS (1.2, 0.5)	NPG1	NPG2
50	5000	Iter.	45399,2	46076,7	*50000*	*50000*	**22437,5**	30189,1
		Res.	3,72E-06	3,42E-06	*1,63E-05*	1,51E-05	9,88E-07	**9,84E-07**
		Obj.	8,92E-08	1,15E-07	*7,16E-06*	6,71E-06	2,96E-08	**0**
		Time(s)	11,0176	11,94366	13,94243	*15,4938*	**5,38424**	7,227166
500	5000	Iter.	1035,1	1060,1	1623,9	*1631,4*	442,3	**404,3**
		Res.	*9,44E-07*	8,97E-07	8,82E-07	8,68E-07	8,1E-07	**8,04E-07**
		Obj.	3,51E-10	*3,53E-10*	2,29E-10	1,92E-10	1,92E-10	**7,22E-11**
		Time(s)	1,00071	1,040429	1,694592	*1,876264*	0,402679	**0,368048**
2000	5000	Iter.	120,4	122,4	*165,1*	163,7	**70,1**	85,5
		Res.	**6,07E-07**	6,31E-07	6,82E-07	*7,87E-07*	6,45E-07	6,34E-07
		Obj.	1,32E-11	1,07E-11	*1,36E-11*	1,28E-11	**6,37E-13**	9,73E-12
		Time(s)	1,07501	1,107907	1,508984	*1,609584*	**0,601122**	0,727446
100	10000	Iter.	49008,7	49202,7	*50000*	*50000*	**26747,2**	36521,4
		Res.	8,29E-06	9,51E-06	3,84E-05	*3,97E-05*	9,86E-07	**9,81E-07**
		Obj.	3,67E-07	4,76E-07	*2,68E-05*	2,49E-05	5,31E-08	**0**
		Time(s)	21,82338	23,03032	25,43881	*29,56451*	**12,71977**	17,39741
1000	10000	Iter.	1052,9	1089,5	*1614,2*	1609,5	451,3	**409**
		Res.	*9,47E-07*	8,7E-07	**6,35E-07**	7,61E-07	8,42E-07	8,56E-07
		Obj.	3,79E-10	2,99E-10	*4,05E-10*	3,41E-10	1,06E-10	**5,66E-11**
		Time(s)	7,133321	7,081205	11,93754	*13,21689*	2,975329	**2,691825**
2000	10000	Iter.	330,1	343,1	*526*	500,3	153,2	**146**
		Res.	8,38E-07	*8,42E-07*	6,99E-07	**5,91E-07**	7,2E-07	6,44E-07
		Obj.	*1,23E-10*	9,39E-11	1,04E-10	1,15E-10	**1,75E-11**	2,42E-11
		Time(s)	4,892416	4,825672	8,328903	*8,78495*	2,213989	**2,131633**

Table 2: Average results for Min-length problem ($N_{max} = 50000$).

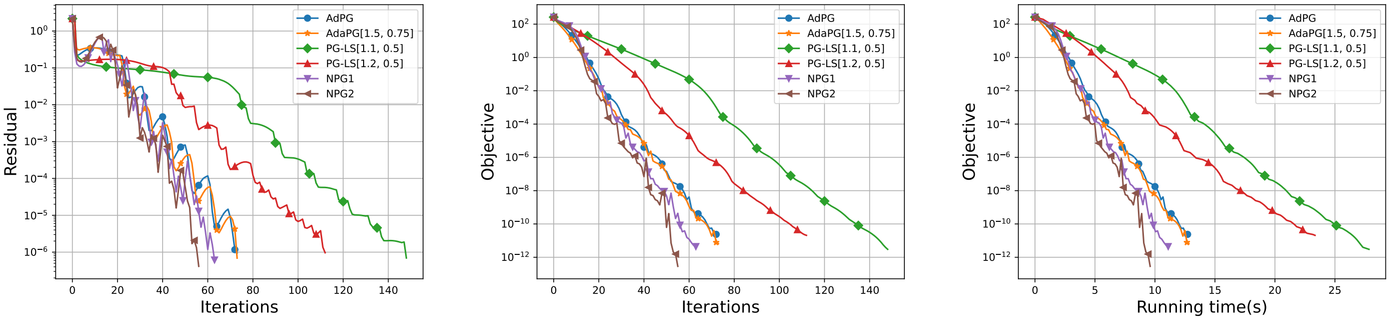

Figure 3: Illustrations for one of randomly generated data of Dual-max-entropy with $m = 4000, n = 5000$.

6.4. Maximum likelihood estimate of the information matrix

This problem (see [31, Eq. (7.5)]) aims to estimate the inverse of a covariance matrix Y of a multivariate random variable subject to the eigenvalue bounds given some samples of the random variable. The problem can be formulated as

$$\min \quad f(X) = -\log\det(X) + \mathrm{tr}(XY) \quad \text{s.t.} \quad X \in \mathbb{S}_n \text{ and } lI \preceq X \preceq uI. \qquad \text{(Max-likelyhood)}$$

Here $\mathbb{S}_n$ denotes the space of real symmetric matrices of dimension $n \times n$, and $A \preceq B$ indicates that $B - A$ is positive semi-definite. Observably, Max-likelyhood[5] satisfies *Assumption 1,2,3* then NPG1 and NPG2 are exact methods to

[5] Max-likelyhood is a case of problem (P) with $f(X) = -\log\det(X) + \mathrm{tr}(XY)$ and $g(X) = \iota_C$ (the indicator function of C) with $C = \{X \in \mathbb{S}_n \mid lI \preceq X \preceq uI\}$.

Size		Metrics	Average of all datasets					
m	n		AdPG	AdaPG$^{(\frac{3}{2},\frac{3}{4})}$	PG-LS (1.1, 0.5)	PG-LS (1.2, 0.5)	NPG1	NPG2
100	500	Iter.	32,7	31,6	*80*	51,1	30,6	**29,1**
		Res.	**4,69E-07**	*5,93E-07*	5,9E-07	4,72E-07	5,67E-07	4,75E-07
		Obj.	3,85E-14	1,37E-13	1,19E-13	1,04E-13	*3,1E-13*	**1,57E-14**
		Time(s)	0,013965	0,013373	*0,032767*	0,021095	0,011194	**0,010558**
500	2000	Iter.	35,3	33,3	*83,7*	54,8	33,4	**31,9**
		Res.	7,44E-07	6,83E-07	*7,9E-07*	5,96E-07	6,3E-07	**4,97E-07**
		Obj.	2,19E-13	9,17E-14	*7,72E-13*	1,6E-13	7,51E-13	**5,93E-14**
		Time(s)	0,328293	0,306634	*0,803013*	0,545206	0,303972	**0,29314**
2000	4000	Iter.	50,1	48,7	*102,1*	70,2	47,5	**45,9**
		Res.	5,68E-07	**4,8E-07**	*8,26E-07*	7,53E-07	5,02E-07	**4,8E-07**
		Obj.	**7,23E-14**	6,69E-13	8,67E-13	8,19E-13	*1,73E-12*	7,17E-13
		Time(s)	3,447821	3,267871	*7,103997*	5,344434	3,184719	**3,14644**
4000	5000	Iter.	79,6	76,4	*151,7*	116,1	73,1	**60,4**
		Res.	6,28E-07	5,52E-07	*7,63E-07*	7,42E-07	6,56E-07	**4,28E-07**
		Obj.	6,27E-12	4,01E-12	4,53E-12	*6,53E-12*	2,94E-12	**1,39E-12**
		Time(s)	14,65683	13,33189	*28,72655*	24,29627	12,93086	**10,5829**

Table 3: Average results for Dual-max-entropy problem ($N_{max} = 200$).

solve Max-likelyhood. The dataset for the implementation is generated analogously to [27] as follows. We initially generate a random vector $y \in \mathbb{R}^n$ with entries from $\mathcal{N}(0, 10)$ and $\delta_i \in \mathbb{R}^n$ with entries from $\mathcal{N}(0, 1)$, and then set $y_i = y + \delta_i$, $i = 1, \ldots, M$. The covariance matrix of the samples $y_1, ..., y_M$ is $Y = \frac{1}{M}\sum_{i=1}^{M} y_i y_i^T$. The obtained results are shown in Table 4 and Figure 4. It is seen that for Max-likelyhood, both of NPG1 and NPG2 provide better results compared to the others with the big deviation. And most of cases NPG2 performs best.

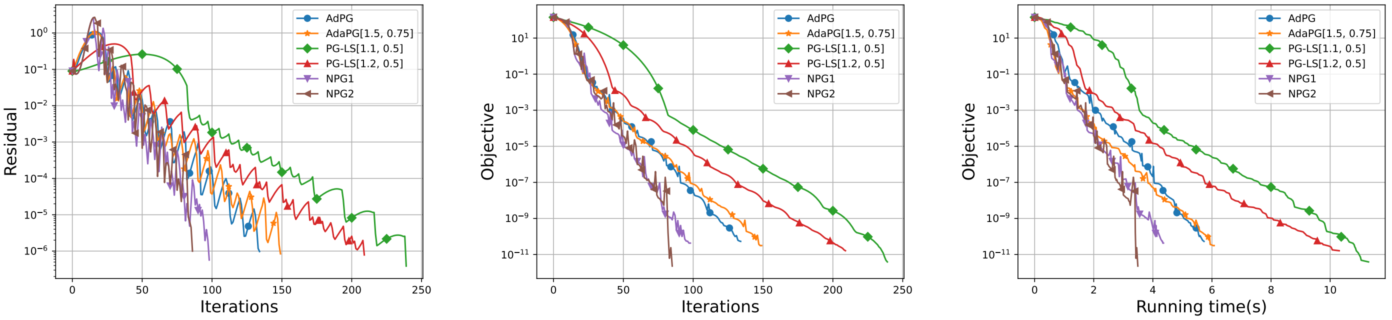

Figure 4: Illustrations for one of randomly generated data of Max-likelyhood with $n = 100, l = 0.1, u = 10, M = 500$.

6.5. Nonnegative matrix factorization

One of efficient approaches to solve recommendation system problems [32] is based on nonnegative matrix factorization[6]

$$\min f(U, V) = \frac{1}{2}\|UV^T - A\|_F^2, \ \text{ s.t. } U \in \mathbb{R_+}^{m\times r}, V \in \mathbb{R_+}^{n\times r}, \tag{NMF}$$

where $A \in \mathbb{R}^{m\times n}$ is a low-rank matrix, $\|\cdot\|_F$ stands for Frobenius norm. This problem does not satisfy *Assumption* 2 and *Assumption* 3. Therefore our algorithms can be seen as heuristic methods for it. Akin to [27], we create A by multiplying matrices B and $C^\top$, where $B \in \mathbb{R_+}^{m\times r}$ and $C \in \mathbb{R_+}^{n\times r}$ have entries drawn from a normal distribution $\mathcal{N}(0, 1)$. All negative entries of B and C are replaced with zero. The numerical results are reported in Table 5 and

[6] NMF is a case of problem (P) with $f(U, V) = \frac{1}{2}\|UV^T - A\|_F^2$ and $g(U, V) = \iota_C$ (the indicator function of C) with $C = \mathbb{R_+}^{m\times r} \times \mathbb{R_+}^{n\times r}$.

Size	Metrics	Average of all datasets					
n, l, u, M		AdPG	AdaPG$^{\left(\frac{3}{2},\frac{3}{4}\right)}$	PG-LS (1.1, 0.5)	PG-LS (1.2, 0.5)	NPG1	NPG2
100, 0.1, 10, 50	Iter.	1661,5	1709,7	*2439*	2364,5	1259,7	**1171,8**
	Res.	*9,58E-07*	9,35E-07	8,68E-07	9,16E-07	9,21E-07	**8,59E-07**
	Obj.	4,27E-09	4,78E-09	1,94E-09	2,74E-09	*6,45E-09*	**8,4E-10**
	Time(s)	60,86365	64,50515	102,6512	*105,7513*	42,38316	**42,36841**
100, 0.1, 10, 500	Iter.	133,7	136,2	*219,2*	197,7	103,5	**93,6**
	Res.	7,15E-07	7,4E-07	6,76E-07	*7,42E-07*	**5,66E-07**	6,45E-07
	Obj.	2,69E-11	2,18E-11	1,29E-11	*2,93E-11*	1,7E-11	**7,07E-12**
	Time(s)	5,251226	5,310266	*9,504244*	8,734855	3,744619	**3,415316**
100, 0.1, 10, 1000	Iter.	57,9	56,7	*103,9*	83,8	58	**49,7**
	Res.	5,69E-07	5,34E-07	4,91E-07	**4,44E-07**	*7,56E-07*	6,19E-07
	Obj.	3,55E-12	2,71E-12	*5,88E-12*	4,99E-12	**9,55E-13**	2,19E-12
	Time(s)	2,160025	2,011673	*4,011467*	3,743083	1,986301	**1,70523**
30, 0.1, 1000, 50	Iter.	5210,2	5355,8	*7612,8*	7518,9	4684,2	**3295,8**
	Res.	9,69E-07	9,45E-07	*2,05E-06*	1,86E-06	**9,3E-07**	9,49E-07
	Obj.	4,28E-09	3,88E-09	*1,34E-07*	1,2E-07	4,69E-09	**1,54E-09**
	Time(s)	7,804621	7,996857	12,49123	*12,80073*	5,995387	**4,278019**
50, 0.1, 1000, 100	Iter.	1644,2	1669,7	*2589,4*	2545,9	1193,8	**954,1**
	Res.	9,4E-07	*9,55E-07*	**8,62E-07**	9,01E-07	8,7E-07	8,67E-07
	Obj.	8,07E-10	9,69E-10	4,87E-10	3,91E-10	*1,35E-09*	**1,52E-10**
	Time(s)	12,11536	12,1823	22,13427	*23,77569*	8,789019	**7,06114**

Table 4: Average results for Max-likelyhood problem ($N_{max} = 20000$).

illustrated by Figure 5. For this problem, NPG1 and NPG2 alternately are proved to be the most effective methods compared to the others.

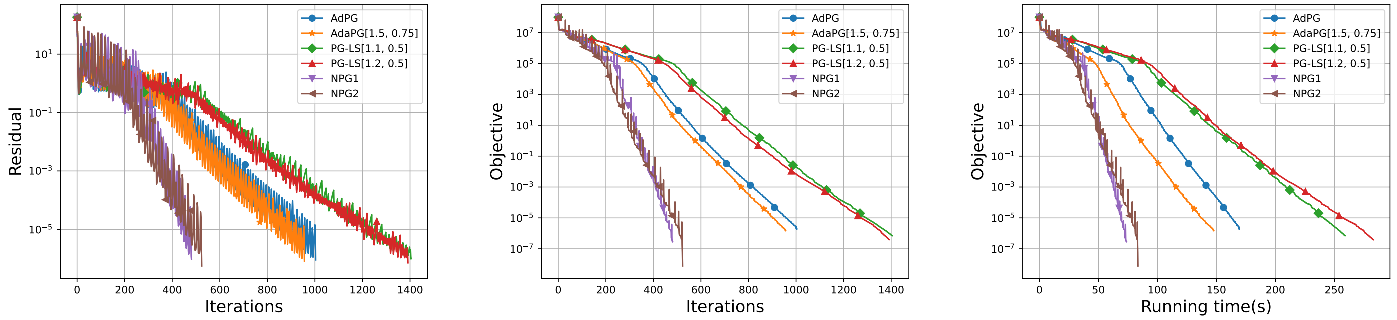

Figure 5: Illustrations for one of randomly generated data of NMF problem with $m = 3000, n = 3000, r = 30$.

7. Conclusions

In this paper, we propose an efficient explicit stepsize NPG applied for the proximal gradient (PG) scheme. In particular, Algorithm 3.1 (NPG1) is the combination of proximal gradient scheme with NPG to solve the convex situation of the problem (P) under the locally Lipschitz gradient condition imposed on f. The iterates is proved to converge to an optimal solution of (P) with the computational complexity $O\left(\frac{1}{k}\right)$ of $F(x^k) - F_*$ and the Q-linear rate if f has locally strong convexity property. These convergence results are based on the descent of our proposed method. Moreover, our stepsize NPG is investigated with a class of nonconvex f satisfying global Lipschitz gradient condition to have the second version in Algorithm 4.1 (NPG2), where the size of steplength can be bigger. In quadratic case of f, NPG is improved significantly in length for solving (P) in Algorithm 5.1 (NPG-quad). Our stepsize selection is computed quickly by a closed formulas without linesearch computation or estimating some constant (like Lipschitz constant of

Size			Metrics	Average of all datasets					
m	r	n		AdPG	AdaPG$^{(\frac{3}{2},\frac{3}{4})}$	PG-LS (1.1, 0.5)	PG-LS (1.2, 0.5)	NPG1	NPG2
500	20	1000	Iter.	537	518	*801,4*	746,3	**302,6**	308,5
			Res.	*9,07E-07*	8,3E-07	8,8E-07	8,84E-07	7,93E-07	**6,6E-07**
			Obj.	*1,93E-07*	1,5E-07	6,04E-08	8,27E-08	1,04E-07	**1,37E-08**
			Time(s)	5,063259	4,854466	8,131989	*8,150044*	**2,640108**	2,687595
1000	20	500	Iter.	543,9	526,3	*777,7*	751,7	**300,5**	309,9
			Res.	8,42E-07	7,99E-07	*9,04E-07*	8,7E-07	8,57E-07	**7,76E-07**
			Obj.	*1,44E-07*	1,09E-07	7,8E-08	5,78E-08	**2,05E-08**	2,85E-08
			Time(s)	4,05312	4,435143	7,149508	*7,690501*	2,394326	**2,33219**
2000	20	3000	Iter.	506,7	491	*731,9*	699,9	**301**	302,6
			Res.	8,33E-07	8,48E-07	*9,29E-07*	9,01E-07	**7,56E-07**	7,69E-07
			Obj.	*4,86E-07*	3,76E-07	2,13E-07	1,65E-07	1,49E-07	**1,44E-07**
			Time(s)	41,5473	39,88436	68,38917	*73,55071*	**24,66209**	24,99118
3000	20	2000	Iter.	509,8	483,8	*716,2*	672,7	**290,1**	305
			Res.	8,26E-07	8,34E-07	8,26E-07	*8,82E-07*	8,15E-07	**6,7E-07**
			Obj.	*4,56E-07*	3,36E-07	1,08E-07	2,11E-07	2,65E-07	**6,11E-08**
			Time(s)	46,16344	43,86382	72,49023	*76,1551*	**26,58725**	27,59743
3000	20	3000	Iter.	498,1	476,6	*701*	671,9	**275,3**	276,9
			Res.	**7,95E-07**	7,99E-07	8,39E-07	*8,97E-07*	8,74E-07	8,11E-07
			Obj.	*4,63E-07*	4,28E-07	1,49E-07	2,44E-07	2,08E-07	**5,89E-08**
			Time(s)	62,75717	59,84711	98,75072	*104,7538*	33,81778	**33,39832**
500	30	1000	Iter.	982,7	970,4	*1493,6*	1422,9	633,6	**598,5**
			Res.	*9,38E-07*	8,39E-07	8,85E-07	9,01E-07	**8,37E-07**	8,84E-07
			Obj.	*4,76E-07*	3,26E-07	1,85E-07	1,38E-07	4,04E-07	**6,43E-08**
			Time(s)	9,372398	8,894667	14,90402	*15,70727*	5,521334	**5,151777**
1000	30	500	Iter.	1026,1	976,6	*1502,3*	1430,2	603,3	**587,3**
			Res.	9E-07	*9,02E-07*	8,93E-07	8,57E-07	**7,87E-07**	8,63E-07
			Obj.	*4,28E-07*	4,08E-07	1,78E-07	1,09E-07	2,44E-07	**3,35E-08**
			Time(s)	7,677596	7,96714	13,22346	*13,87682*	4,588455	**4,575719**
2000	30	3000	Iter.	876,2	872,2	*1247,9*	1200,2	**435,5**	467,2
			Res.	8,75E-07	8,49E-07	8,78E-07	8,77E-07	*8,94E-07*	**7,64E-07**
			Obj.	*1,49E-06*	1,02E-06	2,88E-07	3,06E-07	3,27E-07	**1,1E-07**
			Time(s)	74,42385	73,70845	121,1278	*128,7483*	**37,27746**	39,75814
3000	30	2000	Iter.	907,4	860,5	*1280*	1247,7	**439,6**	469,3
			Res.	8,95E-07	8,43E-07	*9,1E-07*	9,06E-07	**7,71E-07**	8,06E-07
			Obj.	*1,47E-06*	1,28E-06	5,77E-07	6,12E-07	4,89E-07	**1,3E-07**
			Time(s)	85,76037	85,5649	156,2934	*181,0738*	**51,33857**	54,80522
3000	30	3000	Iter.	914,1	902,2	*1303,2*	1252,5	**457,9**	504
			Res.	8,81E-07	*9,33E-07*	8,8E-07	8,89E-07	8,74E-07	**7,46E-07**
			Obj.	*1,7E-06*	1,65E-06	4,86E-07	7,59E-07	**1,84E-07**	3,96E-07
			Time(s)	141,176	139,0688	225,5331	*237,0593*	**66,16939**	72,081

Table 5: Average results for NMF problem ($N_{max} = 5000$).

gradient) to ensure the convergence of the PG algorithm. Moreover, the increasing of the sequence of our stepsizes from some fixed iteration opens the ability to speed up the corresponding PG algorithms. The deep experiments on a variety of test instances with various sizes show the crucial efficiency of the proposed method compared to the recent ones. Future research includes deploying our adaptive stepsize for the composite models in the absence of both convexity and global Lipschitz gradient assumptions on f.

References

[1] M. Ahookhosh, A. Themelis, P. Patrinos, A bregman forward-backward linesearch algorithm for nonconvex composite optimization: superlinear convergence to nonisolated local minima, SIAM J. Optim. 31 (1) (2021) 653–685.

[2] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM J. Imaging Sci. 2 (2009) 183–202.

[3] A. Beck, M. Teboulle, Gradient-based algorithms with applications to signal recovery problems, in: D. Palomar, Y. Eldar (Eds.), Convex Optimization in Signal Processing and Communications, Cambridge University Press, Cambridge, 2009, pp. 139–162.

[4] A. Beck, Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB, Society for Industrial and Applied Mathematics, USA, 2014.

[5] A. Beck, First Order Methods in Optimization, Society for Industrial and Applied Mathematics, USA, 2017.

[6] D. Bertsekas, Nonlinear Programming, 3rd Edition, Athena Scientific, 2016.

[7] P. Combettes, J.-C. Pesquet, Proximal splitting methods in signal processing, in: H. Bauschke, R. Burachik, P. Combettes, V. Elser, D. Luke, H. Wolkowicz (Eds.), Fixed-Point Algorithms for Inverse Problems in Science and Engineering, Springer, New York, 2011, pp. 185–212.

[8] H. Liu, T. Wang, Z. Liu, Some modified fast iterative shrinkage-thresholding algorithms with a new adaptive non-monotone step-size strategy for nonsmooth and convex minimization problems, Comput. Optim. Appl. 83 (2022) 651–691.

[9] X. Jia, C. Kanzow, P. Mehlitz, Convergence analysis of the proximal gradient method in the presence of the kurdyka–Łojasiewicz property without global lipschitz assumptions, SIAM J. Optim. 33 (4) (2023) 3038–3056.

[10] A. Themelis, L. Stella, P. Patrinos, Forward-backward envelope for the sum of two nonconvex functions: further properties and nonmonotone linesearch algorithms, SIAM J. Optim. 28 (3) (2018) 2274–2303.

[11] P. Combettes, V. Wajs, Signal recovery by proximal forward-backward splitting, Multiscale Model. Simul. 4 (4) (2005) 1168–1200.

[12] T. Quoc, L. Liang, K. Toh, A new homotopy proximal variable-metric framework for composite convex minimization, Mathematics of Operation Research 47 (1) (2021) 508–539.

[13] N. Hallak, M. Teboulle, An adaptive lagrangian-based scheme for nonconvex composite optimization, Mathematics of Operation Research 48 (4) (2023) 2337–2352.

[14] M. Fukushima, H. Mine, A generalized proximal point algorithm for certain non-convex minimization problems, Int. J. Syst. Sci. 12 (8) (1981) 989–1000.

[15] R. Bruck, On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in hilbert space, J. Math. Anal. Appl. 61 (1977) 159–164.

[16] G. Passty, Ergodic convergence to a zero of the sum of monotone operators in hilbert space, J. Math. Anal. Appl. 72 (1979) 383–390.

[17] H. Bauschke, J. Bolte, M. Teboulle, A descent lemma beyond lipschitz gradient continuity: first-order methods revisited and applications, Mathematics of Operations Research 42 (2) (2017) 330–348.

[18] J. Bolte, S. Sabach, M. Teboulle, Y. Vaisbourd, First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems, SIAM J. Optim. 28 (3) (2018) 2131–2151.

[19] C. Kanzow, P. Mehlitz, Convergence properties of monotone and nonmonotone proximal gradient methods revisited, J. Optim. Theory Appl. 195 (2) (2022) 624–646.

[20] A. D. Marchi, A. Themelis, Proximal gradient algorithms under local lipschitz gradient continuity, J. Optim. Theory Appl. 194 (2022) 771–794.

[21] X. Jia, C. Kanzow, P. Mehlitz, G. Wachsmuth, An augmented lagrangian method for optimization problems with structured geometric constraints, Math. Program. 199 (2023) 1365–1415.

[22] S. Wright, R. Nowak, M. Figueiredo, Sparse reconstruction by separable approximation, IEEE Trans. Signal Process. 57 (7) (2009) 2479–2493.

[23] M. Teboulle, A simplified view of first order methods for optimization, Math. Program. 170 (1) (2018) 67–96.

[24] R. Dragomir, A. Taylor, A. d'Aspremont, J. Bolte, Optimal complexity and certification of bregman first-order methods, Math. Program. 194 (2022) 41–83.

[25] Y. Malitsky, K. Mishchenko, Adaptive gradient descent without descent, in: ICML, Vol. 119, 2020, pp. 6702–6712.

[26] P. Hoai, N. Vinh, N. Chung, A novel stepsize for gradient descent method, Operations Research Letters (2024) 107072doi:https://doi.org/10.1016/j.orl.2024.107072.

[27] Y. Malitsky, K. Mishchenko, Adaptive proximal gradient method for convex optimization, arXiv preprint (2024). URL https://arxiv.org/pdf/2308.02261.pdf

[28] P. Latafat, A. Themelis, L. Stella, P. Patrinos, Adaptive proximal algorithms for convex optimization under local lipschitz continuity of the gradient, arXiv preprint (2023).
URL https://arxiv.org/abs/2301.04431

[29] P. Latafat, A. Themelis, P. Patrinos, On the convergence of adaptive first order methods: proximal gradient and alternating minimization algorithms, in: Proceedings of Machine Learning Research, Vol. 242, 2024, pp. 197–208.

[30] M. Figueiredo, R. Nowak, S. Wright, Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, IEEE Journal of Selected Topics in Signal Processing 1 (4) (2007) 586–597.

[31] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, 2004.

[32] P. Symeonidis, A. Zioupos, Matrix and Tensor Factorization Techniques for Recommender Systems, Springer Briefs in Computer Science, 2016.