Noname manuscript No.
(will be inserted by the editor)

New inexact adaptive proximal gradient algorithms for nonconvex composite optimization problems

Pham Thi Hoai [1] · **Nguyen Dang Hao**[1] · **Jen-Chih Yao** [2]

Received: date / Accepted: date

Abstract In this paper, we propose new *inexact adaptive* proximal gradient algorithms for solving nonconvex composite optimization problems, where the objective is the sum of a differentiable nonconvex function and a convex non-differentiable function. A new relative error criterion to compute the proximal operator inexactly has been proposed together with new adaptive strategies for selecting stepsizes used in inexact proximal gradient steps. In particular, both of relative error criterion and adaptive stepsizes in our new method are dynamically adjusted quickly at each iteration by using information of the current step. Standard theoretical convergence results for our proposed algorithms are obtained for both nonconvex and convex cases of composite optimization models. Specifically, for a wide class of nonconvex objective functions we show that any cluster point of the iterates is a stationary point of the considered problem. For the convex case, we prove that the sequence generated by our algorithms converge to a global optimal solution of the considered problem with sublinear convergence rate (from a fixed iteration). To the best of our knowledge, no *inexact* proximal gradient algorithms with *adaptive* stepsizes have been proposed for *nonconvex* composite optimization problems to date. The efficiency of our new algorithms is verified throughout extensive numerical experiments for applicative composite optimization problems with benchmark and synthetic datasets.

Keywords inexact proximal gradient algorithms · nonlinear programming · composite optimization problems · nonconvex optimization · adaptive proximal gradient

Mathematics Subject Classification (2010) 49J40 · 47H04 · 47H10

1 Introduction

In this paper, we focus on addressing nonsmooth and nonconvex optimization problems characterized by the following formulation:

$$\min\ F(x) := f(x) + g(x) \quad \text{subject to} \quad x \in \mathbb{E}, \tag{P}$$

✉ Pham Thi Hoai
hoai.phamthi@hust.edu.vn; phamhoai051087@gmail.com
Nguyen Dang Hao
haond4work@gmail.com
Jen-Chih Yao
yaojc@mail.cmu.edu.tw

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

[2] Research Center for Inter neural Computing, China Medical University Hospital, China Medical University, Taichung, Taiwan

where $\mathbb{E}$ is a finite-dimensional Euclidean space, f and g are functions satisfying *Assumption 1* below.

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

(A2) $f : \mathbb{E} \to (-\infty, +\infty]$ *is proper and closed such that* $\operatorname{dom} f$ *is convex,* $\operatorname{dom} g \subset \operatorname{int}(\operatorname{dom} f)$ *and* ∇f *is globally Lipschitz on* $\operatorname{dom} g$ *with Lipschitz constant* L_f.

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

The optimization model (P) serves as a mathematical backbone for various modern applications in science and engineering, including inverse problems, image processing, and high-dimensional statistical learning [27,17,14,15,33,36]. To solve Problem (P), one of the traditional approaches is proximal gradient method which was introduced by Fukushima and Mine [20]. Actually, this one is developed from forward backward splitting method in the work of Bruck [16] and Passty [31] for solving variational inequalities and inclusion problems (generalizations of Problem (P)). The detailed idea of this method for solving (P) is as follows (see e.g., [9]):

Algorithm 1.1 The Proximal Gradient Method

Initialization: Choose $x^0 \in \operatorname{dom} g$.
General step: for any $k = 0, 1, 2, ...$, execute the following steps:
(a) pick $t_k > 0$
(b) set $x^{k+1} = \operatorname{prox}_{t_k g}\left(x^k - t_k \nabla f(x^k)\right)$

The main step in proximal gradient method involves evaluating the proximal operator $\operatorname{prox}_{tg}(\cdot)$ defined by

$$\operatorname{prox}_{tg}(y) = \underset{x \in \mathbb{E}}{\operatorname{argmin}} \left\{ \frac{1}{2t} \|x - y\|^2 + g(x) \right\}. \tag{1}$$

Clearly, from the definition of proximal gradient operator in (1) we have to solve a convex optimization problem to find its value at each step. Fortunately, in some special cases, the optimal solution to (1) is available under a closed-form; for instance, when g is the ℓ_1 norm or the indicator function of a closed convex set $C \subset \mathbb{E}$ (where C has a special structure like a box, hyperplane, ball,...). One can find a list of such functions g in [9] where $\operatorname{prox}_{tg}(y)$ is analytically computable. However, when the non-smooth function g possesses a complex structure; for example, overlapping group lasso, elastic net or nuclear norm regularization,..., see e.g. [6,37,10] and the references therein), the exact solution to (1) rarely admits a closed-form expression. Consequently, solving it to high precision at every iteration becomes computationally prohibitive. To overcome this drawback, inexact computation techniques are utilized to find an optimal solution of (1). In recent years, various inexact proximal gradient methods have been proposed to resolve costly exact proximal evaluations, evolving from frameworks with absolute, summable error tolerances [32] to more adaptive, endogenous relative error rules [34,29]. In particular, [32,21, 28,26] approximate the proximal operator using an exogenous sequence of absolute error tolerances satisfying the summability of the given errors. Notably, in these inexact proximal gradient algorithms, the stepsizes are commonly chosen as fixed constants for proximal gradient steps. Although these methods are guaranteed convergent even for nonconvex case of (P), such diminishing error sequences must be chosen a priori, thereby completely disregarding the dynamic information available during the iterative process and often forcing unnecessarily high-precision computations in the early iterations.. Moreover, the fixed stepsize may require the knowledge of Lipschitz gradient constant (L_f) and the fixed stepsize may be small if L_f is large. Other research such as [35,5,25] applied the absolute errors for inexact computation but in accelerated versions. To overcome the limitation of absolute errors, one can compute an inexact optimal solution of (1) based on a relative inexactness criterion which utilizes the behavior of the algorithms during the computation and therefore completely liberating the algorithm from restrictive summability conditions of absolutely inexact error method. There are a lot of publications following this research direction such as [34,29] for monotone inclusion problems; [1,2,18,19,4] for ADMM or Augmented Lagrangian algorithms; [28,13,10] for proximal gradient scheme solving composite optimization problems.

Recently, motivated by [11,29], Bello-Cruz et al. [10] proposed a new relative error for computing inexact optimal solution of (1) with $t = 1$, i.e., determining $\tilde{x}^k = \mathrm{prox}_g(x^k - \nabla f(x^k))$ and then computing $x^{k+1} = x^k + \beta_k(\tilde{x}^k - x^k)$, where β_k is exogenous stepsize chosen by using an explicit line search procedures. The detailed algorithm proposed by Bello-Cruz et al. [10] is the following.

IPG-ELS (Bello-Cruz et al. [10]) - for solving the **convex** case of (P)

1. **Initialization.** Let $x^0 \in \mathrm{dom}\, g$, $\tau \in (0,1]$, $\theta \in (0,1)$, $\gamma_1 > 1$, $\gamma_2 \geq 1$, and $\alpha \in [0, 1-\tau]$. Set $k = 0$.
2. **Inexact Proximal Subproblem.** Compute $\tilde{x}^k \approx \mathrm{prox}_g(x^k - \nabla f(x^k))$ as an inexact proximal solution to (1) with $t = 1$ along with a residual pair (v^k, ε_k). Specifically, find a triple $(\tilde{x}^k, v^k, \varepsilon_k)$ such that
$$v^k \in \nabla f(x^k) + \partial_{\varepsilon_k} g(\tilde{x}^k) + \tilde{x}^k - x^k, \tag{2}$$
$$g(\tilde{x}^k - v^k) - g(\tilde{x}^k) - \langle \nabla f(x^k), v^k \rangle + \frac{(1+\gamma_1)}{2}\|v^k\|^2 + (1+\gamma_2)\varepsilon_k \quad \leq \frac{(1-\tau-\alpha)}{2}\|x^k - \tilde{x}^k\|^2; \tag{3}$$
3. **Stopping Criterion.** If $\tilde{x}^k = x^k$, then stop.
4. **Line search Procedure.** Set $\beta = 1$ and $y^k := \tilde{x}^k - v^k$. If
$$f\left(x^k + \beta(y^k - x^k)\right) \leq f(x^k) + \beta\langle \nabla f(x^k), y^k - x^k \rangle + \frac{\beta\tau}{2}\|x^k - \tilde{x}^k\|^2 + \frac{\gamma_1}{2}\beta\|v^k\|^2 + \beta\gamma_2\varepsilon_k, \tag{4}$$
then set $\beta_k = \beta$, $x^{k+1} = x^k + \beta_k(y^k - x^k)$ and $k := k+1$, and go to Step 2. Otherwise, set $\beta = \theta\beta$, and verify (4).

It is worth noting that IPG-ELS [10] integrates a relative inexactness mechanism in (2) and (3) with an explicit line search stepsizes strategy in (4), further reducing computational cost by performing the inexact proximal evaluation only once per iteration. IPG-ELS reduces PG-ELS [11] if the computation of proximal gradient operator is exact. The convergence of IPG-ELS is investigated for convex case of Problem (P). Experimental results given by [10] demonstrate its efficiency compared with the exact and inexact methods in the literature. However, the stepsize strategies using line search backtracking may suffer from the expensive cost during the computation.

In this paper, motivated by Bello-Cruz et al. [10] and Hoai and Thai [23], we introduce a new framework that simultaneously integrates two key components: an adaptive stepsize strategy inspired by [23], and an inexact calculation mechanism governed by a relative error criterion inspired by [10]. In particular, we propose two New Inexact adaptive Proximal Gradient algorithms (NIPG1 and NIPG2) which have the following features:

- By generalizing the inexact mechanism of IPG-ELS which is only applied for computing $\mathrm{prox}_g(x^k - \nabla f(x^k))$, we propose a new relative error to inexactly calculate a proximal solution with dynamical stepsize t_k ($t_k > 0$), i.e., $\tilde{x}^k \approx \mathrm{prox}_{t_k g}(x^k - t_k\nabla f(x^k))$. The value of $\tilde{x}^k$ will be utilized to determine the next iterate.
- The adaptive stepsize t_k of our method is computed by one of the two adaptive ways (corresponding to two versions of NIPG) which utilize the behavior of the differentiable term f. Neither line search backtracking procedure nor the knowledge of the Lipschitz constant L_f is required to compute t_k.

- The convergence of our new inexact proximal gradient algorithms (NIPG1, NIPG2) is obtained for a wide class of nonconvex objective functions. Specifically, any cluster point of the sequence generated by NIPG method is proved to be a stationary point of (P). For the convex case, we prove the sequence generated by our algorithms converge to a global optimal solution of (P).
- From a finite iteration, we establish the convergence rates of $O(\frac{1}{\sqrt{K}})$ and $O(\frac{1}{K})$ for a large class of nonconvex objectives and convex objectives, respectively. To the best of our knowledge, no inexact proximal gradient algorithms with *adaptive stepsizes* have been proposed for nonconvex composite optimization problems so far.
- The compactness of our algorithms speed up the processing time significantly. This is demonstrated by extensive numerical experiments for applicative convex and nonconvex composite optimization problems with benchmark and synthetic datasets.

The remainder of this paper is structured as follows. Section 2 presents preliminary materials, including basic definitions, notations, and auxiliary lemmas. Section 3 introduces the concept of an inexact proximal solution under relative error criteria, details our new adaptive inexact proximal gradient (NIPG) algorithms, and provides a methodological comparison with existing methods. Section 4 establishes the convergence properties and results for the nonconvex case. Section 5 analyzes the convergence behavior and rates of the proposed algorithms in the convex setting. Numerical experiments on various high-dimensional datasets are reported in Section 6 to demonstrate the computational efficiency and practical performance of our methods. Finally, concluding remarks and future research directions are provided in Section 7.

2 Preliminaries

In this section, we recall some preliminary materials and notations that will be used throughout this paper with the notation that f, g satisfy Assumption 1.

Definition 2.1 (ϵ-subdifferential) Let $g : \mathbb{E} \to (-\infty, +\infty]$ be a proper, lower semi-continuous, and convex function. For a given $\epsilon \geq 0$, the ϵ-subdifferential of g at $x \in \operatorname{dom} g := \{x \in \mathbb{E} \mid g(x) < +\infty\}$, denoted by $\partial_\epsilon g(x)$, is defined as

$$\partial_\epsilon g(x) := \{v \in \mathbb{E} \mid g(y) \geq g(x) + \langle v, y - x\rangle - \epsilon, \forall y \in \mathbb{E}\}. \tag{5}$$

When $x \notin \operatorname{dom} g$, we define $\partial_\epsilon g(x) = \emptyset$. Any element $v \in \partial_\epsilon g(x)$ is called an ϵ-subgradient. If $\epsilon = 0$, then $\partial_0 g(x)$ becomes the classical subdifferential of g at x, denoted by $\partial g(x)$.

Remark 2.1 It follows immediately from (5) that

$$v \in \partial_\epsilon g(y),\ u \in \partial g(x) \implies \langle v - u, y - x\rangle \geq -\epsilon. \tag{6}$$

Before proceeding, it is useful to recall two fundamental characteristics associated with $\partial_\epsilon g$. Specifically, the graph of $\partial_\epsilon g$ is closed, and it obeys the so-called transportation formula.

Lemma 2.1 (Closed graph [22, Proposition 4.1.1]) *Let $(x^k, v^k, \varepsilon_k)_{k\in\mathbb{N}} \subseteq \mathbb{E} \times \mathbb{E} \times \mathbb{R}_+$ be a sequence converging to $(x, v, \epsilon) \in \mathbb{E} \times \mathbb{E} \times \mathbb{R}_+$ with $v^k \in \partial_{\epsilon_k} g(x^k)$ for all $k \in \mathbb{N}$. Then, $v \in \partial_\epsilon g(x)$.*

Lemma 2.2 (Transportation Formula [22, Proposition 4.1.2]) *For $x, y \in \operatorname{dom}\ g$, let $v \in \partial g(y)$. Then $v \in \partial_\epsilon g(x)$ where $\epsilon = g(x) - g(y) - \langle v, x - y\rangle \geq 0$.*

Next, we present the concept of the approximate stationary point of Problem (P). We refer to [29, Eq. 1] for the original one when applying for the more general context (the monotone inclusion problem) of the convex case of Problem (P). However, we can use this definition for the nonconvex case of (P) provided that it matches Assumption 1. First, note that $\bar{x}$ is a stationary point to Problem (P) if and only if $0 \in \nabla f(\bar{x}) + \partial g(\bar{x})$. The concept of approximate stationary point relaxes this inclusion by introducing a residual v and enlarging ∂g using $\partial_\epsilon g$.

Definition 2.2 **(η-approximate stationary point [?, Eq 1])** Given $\eta \geq 0$, an element $\tilde{x} \in \operatorname{dom} g$ is called an η-approximate stationary point of Problem (P) with a residual pair $(w, \epsilon) \in \mathbb{E} \times \mathbb{R}_+$ if

$$v \in \nabla f(\tilde{x}) + \partial_\epsilon g(\tilde{x}), \quad \max\{\|v\|, \epsilon\} \leq \eta. \tag{7}$$

Obviously, when $\eta = 0$, $\tilde{x}$ is a stationary point of (P).

We now review the concept of quasi-Fejér convergence and its properties.

Definition 2.3 **(Quasi-Fejér Convergence [3, Definition 1])** Let S be a nonempty subset of $\mathbb{E}$. A sequence $\{x^k\}_{k\in\mathbb{N}} \subseteq \mathbb{E}$ is said to be quasi-Fejér convergent to S if and only if, for every $x \in S$, there exists a summable sequence $(\delta_k)_{k\in\mathbb{N}} \subseteq \mathbb{R}_+$ such that

$$\|x^{k+1} - x\|^2 \leq \|x^k - x\|^2 + \delta_k, \quad \forall k \in \mathbb{N}. \tag{8}$$

The following result presents the main properties of quasi-Fejér convergent sequences.

Lemma 2.3 **(Quasi-Fejér Convergence Properties [7, Theorem 2.6])** *If $\{x^k\}_{k\in\mathbb{N}}$ is quasi-Fejér convergent to S, then the following statements hold:*

(a) The sequence $\{x^k\}_{k\in\mathbb{N}}$ is bounded;

(b) If an accumulation point $\bar{x}$ of $\{x^k\}_{k\in\mathbb{N}}$ belongs to S, then $\{x^k\}_{k\in\mathbb{N}}$ is convergent to $\bar{x}$.

3 New inexact adaptive proximal gradient (NIPG) algorithms

In this section, we first define some basic concepts which contribute to our new method. We then introduce the details of our proposed algorithms and provide a methodological comparison with related approaches in the literature.

Let us start with a definition of auxiliary functions $A_\ell(x, y) : \operatorname{dom} g \times \operatorname{dom} g \to \mathbb{R},\ \ell = 1, 2,$ which play an important role in computing the adaptive stepsizes in our new method.

Definition 3.1 For any $x, y \in \operatorname{dom} g$, we construct a function

$$A_\ell(x, y) : \operatorname{dom} g \times \operatorname{dom} g \to \mathbb{R},\ \ell = 1, 2, \tag{9}$$

with $A_1(x, y) = |\langle \nabla f(x) - \nabla f(y), x - y\rangle|$ and $A_2(x, y) = \|\nabla f(x) - \nabla f(y)\|\,\|x - y\|$.

To design our new inexact proximal algorithms, we also need another definition of inexact proximal solution in the following.

Definition 3.2 **(Inexact Proximal Solution)** Given an iterate x^k and the stepsize $t_k > 0$, the next iterate x^{k+1} of proximal gradient scheme is defined as follows

$$x^{k+1} = \operatorname{prox}_{t_k g}\left(x^k - t_k \nabla f(x^k)\right) = \operatorname*{argmin}_{x\in\mathbb{E}} \left\{ g(x) + \frac{1}{2t_k} \left\| x - (x^k - t_k \nabla f(x^k)) \right\|^2 \right\} \tag{10}$$

corresponding to solve the following monotone inclusion problem:

$$0 \in t_k \nabla f(x^k) + t_k \partial g(x^{k+1}) + x^{k+1} - x^k. \tag{11}$$

Then, a triple $(\tilde{x}^k, v^k, \epsilon_k) \in \mathbb{E} \times \mathbb{E} \times \mathbb{R}_+$ is called an *inexact proximal solution* of (11) if it satisfies the following relative error criteria:

$$v^k \in t_k \nabla f(x^k) + t_k \partial_{\epsilon_k} g(\tilde{x}^k) + \tilde{x}^k - x^k, \tag{12}$$

and

$$g(\tilde{x}^k - v^k) - g(\tilde{x}^k) - \left\langle \nabla f(x^k), v^k \right\rangle + \frac{\theta_1}{t_k}\|v^k\|^2 + (1 + \theta_2)\epsilon_k \leq \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2, \tag{13}$$

where $\theta_1 > 1$, $\theta_2 \geq 1$, and $\tau \in [0, \frac{1}{2})$ are pre-defined parameters.

We emphasize some noticeable items in the following remark to get insights into Definition 3.2.

Remark 3.1 (i) *Components of the Inexactness*: The approximate solution is characterized by two distinct conditions. Specifically, inequality (13) dynamically couples these errors with the deviation $\|x^k - \tilde{x}^k\|^2$. The parameter $\tau \in \left[0, \frac{1}{2}\right)$ dictates the stringency of this coupling: a smaller τ forces the approximate solution to be closer to the exact one, while a larger τ allows for more computationally relaxed updates. Additionally, it is observed that, the relation (12) is indeed

$$w^k = \frac{v^k - \tilde{x}^k + x^k}{t_k} - \nabla f(x^k) + \nabla f(\tilde{x}^k) \in \nabla f(\tilde{x}^k) + \partial_{\varepsilon_k} g(\tilde{x}^k). \tag{14}$$

Therefore, if $\max\{\|w^k\|, \varepsilon_k\} \leq \eta$ (for $\eta \geq 0$) then $\tilde{x}^k$ is an η-approximation stationary point of Problem (P) according to Definition 2.2.

(ii) *Recovery of the Exact Proximal Step*: If the proximal subproblem is solved exactly, i.e., $\epsilon_k = 0$ and $v^k = 0$. In this scenario, the inclusion (12) reduces to:

$$0 \in t_k \nabla f(x^k) + t_k \partial g(\tilde{x}^k) + \tilde{x}^k - x^k \iff \tilde{x}^k = \mathrm{prox}_{t_k g}\left(x^k - t_k \nabla f(x^k)\right).$$

Furthermore, the left-hand side of inequality (13) trivially becomes zero, which is always less than or equal to $\frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2$. Thus, the exact proximal gradient step is a special case of our framework.

(iii) *Practical Implementation*: Finding a triple $(\tilde{x}^k, v^k, \epsilon_k)$ that satisfies Definition 3.2 is highly practical. In practice, the proximal subproblem can be solved by using inner iterative procedures, such as the Dykstra-like algorithm [8]; ADMM algorithm [1,2,4] or by Frank-Wolfe method. These inner algorithms can be terminated early based on a computable stopping criterion constructed directly from (13), thereby yielding significant computational savings. One can see the detailed description of these procedures in Sections 3.1 and 3.2 in [10].

Now, we are ready to propose new inexact proximal gradient algorithms with adaptive stepsize for solving Problem (P) as follows.

Algorithm 3.1 NIPGℓ, $\ell = 1, 2$

Preparation: From Definition 3.1, pick an option for the function $A_\ell(x, y)$, $\ell = 1, 2$. Then NIPGℓ corresponds to the option of $A_\ell(x, y)$.

Step 0. Select $t_0 > 0$, $\theta_1 > 1$, $\theta_2 \geq 1$, $\tau \in [0, 1/2)$, $0 < c_1 < c_0 \leq \frac{1}{2}$ and a positive real sequence $\{\gamma_k\}$ such that $\sum_{k=0}^{+\infty} \gamma_k < \infty$. Choose $x^0 \in \operatorname{dom} g$, $t_{-1} = t_0$ and set $k = 0$.

Step 1. Using Definition 3.2, compute an inexact proximal solution $\tilde{x}^k \approx \operatorname{prox}_{t_k g}\left(x^k - t_k \nabla f(x^k)\right)$, specifically, find a triple $(\tilde{x}^k, v^k, \varepsilon_k) \in \mathbb{E} \times \mathbb{E} \times \mathbb{R}_+$ such that

$$v^k \in t_k \nabla f(x^k) + t_k \partial_{\varepsilon_k} g(\tilde{x}^k) + \tilde{x}^k - x^k, \tag{15}$$

$$g(\tilde{x}^k - v^k) - g(\tilde{x}^k) - \left\langle \nabla f(x^k), v^k \right\rangle + \frac{\theta_1}{t_k}\|v^k\|^2 + (1 + \theta_2)\varepsilon_k \leq \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2. \tag{16}$$

Step 2. If $\tilde{x}^k = x^k$ **then** STOP
else compute $x^{k+1} = \tilde{x}^k - v^k$ and go to **Step 3**.

Step 3.

$$\textbf{If} \quad A_\ell(x^k, x^{k+1}) > \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 \tag{17}$$

$$\textbf{then} \quad t_{k+1} = c_1 \frac{\|x^{k+1} - x^k\|^2}{A_\ell(x^k, x^{k+1})} \tag{18}$$

else $\gamma'_k = \gamma_k$

if $\dfrac{t_k}{t_{k-1}} < 1$ then $\gamma'_k = \min\left\{\gamma_k, \sqrt{1 + \dfrac{t_k}{t_{k-1}}} - 1\right\}$

$$t_{k+1} = (1 + \gamma'_k)t_k. \tag{19}$$

Setting $k := k + 1$ and return to **Step 1**.

To understand our new method more thoroughly, we first outline the main differences of Algorithm 3.1 with the most related ones in the literature in the following remark.

Remark 3.2 (i) *The relation of our method and NPG method in [23]*: When $v^k = 0$ and $\varepsilon_k = 0$, following Remark 3.1 (ii) we have $x^{k+1} = \tilde{x}^k = \operatorname{prox}_{t_k g}\left(x^k - t_k \nabla f(x^k)\right)$ and with $\ell = 2$, NIPGℓ reduces NPG1 [23] with a little change in the range of c_0, c_1. In particular, $0 < c_1 < c_0 < \frac{1}{\sqrt{2}}$ for NPG1 but $0 < c_1 < c_0 \leq \frac{1}{2}$ for NIPG2.

(ii) *The relation of our method and IPG-ELS in [10]*: we highlight the three different points:

(a) For the definition of inexact proximal solution, Definition 3.1 in Bello-Cruz et al. [10] applies to finding the approximation of $\operatorname{prox}_g(x^k - \nabla f(x^k))$, i.e., fixing $t_k = 1$ in IPG-ELS (see (2) and (3)). Whereas, ours in Definition 3.2 uses for computing $\operatorname{prox}_{t_k g}(x^k - t_k \nabla f(x^k))$ inexactly with an arbitrary $t_k > 0$ (see (15) and (16)). In addition, for $t_k = 1$, Definition 3.2 reduces to Definition 3.1 in Bello-Cruz et al. [10]. It is worth noting that Bello-Cruz et al. [10] reviewed and compared their inexact definition with the related ones in the literature thoroughly. One can see Section 3 in [10] for more details.

(b) After computing proximal solution inexactly, if the stopping criterion does not satisfy, IPG-ELS runs line search backtracking procedure (4) to find the exogenous stepsize β_k to determine the next iterate. While, our algorithms (Algorithm 3.1-NIPGℓ) do not need using any backtracking line search procedure and the knowledge of L_f to figure out t_k. Our method easily computed t_k by a closed formula (18) or (19) after checking the condition (17) one time.

(c) In the next section we will show that NIPGℓ, $\ell = 1, 2$ converge for a wide class of nonconvex case of (P) while the convergence of IPG-ELS is obtained for the convex case of (P).

We end this section by giving a remark for finding a triple $(\tilde{x}^k, v^k, \varepsilon_k)$ satisfying (15) and (16). Clearly, if we choose $v^k = 0$ then we do not need to calculate the values of g in the inequality (16) since $g(\tilde{x}^k - v^k) - g(\tilde{x}^k) = 0$ in this case. And the inexact relations (15) and (16) become

$$0 \in t_k \nabla f(x^k) + t_k \partial_{\varepsilon_k} g(\tilde{x}^k) + \tilde{x}^k - x^k, \tag{20}$$

$$\varepsilon_k \leq \frac{\tau}{(1+\theta_2)t_k} \|x^k - \tilde{x}^k\|^2. \tag{21}$$

Then if we find a pair of $(\tilde{x}^k, \varepsilon_k)$ satisfying conditions (20) and (21) then $\tilde{x}^k$ is a desired inexact proximal solution. To make it more concretely, in the following, we will give an example illustrating the implementation of inexact steps (15) and (16) in Algorithm 3.1-NIPGℓ when we choose $v^k = 0$.

Example 3.1 Given $\mathbb{E} = \mathbb{R}^{n\times m}$, $\lambda_{\text{row}} > 0, \lambda_{\text{col}} > 0$; and $\mu \geq 0$; a matrix $X \in \mathbb{R}^{m\times n}$ and $X^{(i)}$ $(i = 1, \dots, n)$, $X_{(j)}$ $(j = 1, \dots, m)$ respectively denote the row i and the column j of X. Define the functions

$$g_\zeta : \mathbb{E} = \mathbb{R}^{n\times m} \to (-\infty, +\infty), \ \ \zeta \in \{1, 2, 3\}$$

by $g_1(X) = \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2$, $g_2(X) = \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2$ and $g_3(X) = \frac{\mu}{2}\|X\|_F^2$ ($\|\cdot\|_F$ stands for the Frobenius norm of a matrix and $\|\cdot\|_2$ denotes for the Euclidean norm of a vector). Obviously, for some $t > 0$ and $Z \in \mathbb{R}^{n\times m}$ we have the analytic expression of $\text{prox}_{tg_1}(Z)$, $\text{prox}_{tg_2}(Z)$ and $\text{prox}_{tg_3}(Z)$. Specifically, based on the separability of the proximal operator and the results in Section 6.5.1 of [30], we have

$$(\text{prox}_{tg_1}(Z))^{(i)} = \max\left\{1 - \frac{\lambda_{\text{row}}}{t\|Z^{(i)}\|_2}, 0\right\} Z^{(i)}, \quad \text{for} \quad i = 1, \dots, n \tag{22}$$

and

$$(\text{prox}_{tg_2}(Z))_{(j)} = \max\left\{1 - \frac{\lambda_{\text{col}}}{t\|Z_{(j)}\|_2}, 0\right\} Z_{(j)}, \quad \text{for} \quad j = 1, \dots, m. \tag{23}$$

Additional by easy computation we also get that $\text{prox}_{tg_3}(Z) = \frac{1}{1+t\mu} Z$. Now, setting

$$g(X) = \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2 + \frac{\mu}{2}\|X\|_F^2 = g_1(X) + g_2(X) + g_3(X),$$

we do not have the closed form of $\text{prox}_{tg}(Z)$ and how can we find it? Relying on Definition 3.2 and Algorithm 3.1 we can find $\text{prox}_{tg}(Z)$ inexactly, particularly, satisfying the conditions (15) and (16). It is worth noting that such type of function g usually occurs in the CUR-like factorization when g represents some regularization like group lasso or elastic net.

Suppose that we are at iteration k and we want to compute a triple $(\tilde{X}^k, v^k, \varepsilon_k)$ that satisfies conditions (15) and (16). First, for $Z = X^k - t_k \nabla f(X^k)$, the exact proximal solution $\text{prox}_{t_k g}(Z)$ is defined by

$$\text{prox}_{t_k g}(Z) = \underset{X\in\mathbb{R}^{n\times m}}{\text{argmin}} \left\{ \frac{1}{2t_k}\|X - Z\|_F^2 + \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2 + \frac{\mu}{2}\|X\|_F^2 \right\}. \tag{24}$$

Simplifying

$$\begin{aligned}
\frac{\mu}{2}\|X\|_F^2 + \frac{1}{2t_k}\|X - Z\|_F^2 &= \frac{\mu}{2}\|X\|_F^2 + \frac{1}{2t_k}\left(\|X\|_F^2 - 2\langle X, Z\rangle + \|Z\|_F^2\right) \\
&= \frac{1+\mu t_k}{2t_k}\|X\|_F^2 - \frac{1}{t_k}\langle X, Z\rangle + \frac{1}{2t_k}\|Z\|_F^2 \\
&= \frac{1+\mu t_k}{2t_k}\left(\|X\|_F^2 - 2\frac{1}{1+\mu t_k}\langle X, Z\rangle\right) + \frac{1}{2t_k}\|Z\|_F^2 \\
&= \frac{1+\mu t_k}{2t_k}\left\|X - \frac{1}{1+\mu t_k}Z\right\|_F^2 + C,
\end{aligned}$$

where C is a constant that depends on Z and t_k. Therefore, we can rewrite (24) as

$$\begin{aligned}
\operatorname{prox}_{t_k g}(Z) &= \operatorname*{argmin}_{X\in\mathbb{R}^{n\times m}} \left\{ \frac{1+\mu t_k}{2t_k}\left\|X - \frac{1}{1+\mu t_k}Z\right\|_F^2 + \lambda_{\text{row}}\sum_{i=1}^{n}\|X^{(i)}\|_2 + \lambda_{\text{col}}\sum_{j=1}^{m}\|X_{(j)}\|_2 \right\} \\
&= \operatorname*{argmin}_{X\in\mathbb{R}^{n\times m}} \left\{ \frac{1}{2}\|X - Z_{eff}\|_F^2 + \frac{\lambda_{\text{row}}}{L_{eff}}\sum_{i=1}^{n}\|X^{(i)}\|_2 + \frac{\lambda_{\text{col}}}{L_{eff}}\sum_{j=1}^{m}\|X_{(j)}\|_2 \right\} \qquad (25) \\
&= \operatorname{prox}_{\frac{1}{L_{eff}}\bar{g}}(Z_{eff}),
\end{aligned}$$

where $L_{eff} = \dfrac{1+\mu t_k}{t_k}$, $Z_{eff} = \dfrac{Z}{1+\mu t_k}$ and $\bar{g} = g_1 + g_2$. Following [10], we can use the Dykstra-like algorithm in [8] for solving (25) or finding $\operatorname{prox}_{\frac{1}{L_{eff}}\bar{g}}(Z_{eff}) = \operatorname{prox}_{t_k g}(Z)$ as follows.

The computation of an inexact proximal gradient solution in Step 1 of Algorithm 3.1
With the initial points $z_0 = Z_{eff}$, $p_0 = 0$, $q_0 = 0$, Dykstra-like algorithm in [8] generates the sequences

$$\begin{cases} y_r = \operatorname{prox}_{\frac{1}{L_{eff}}g_1}(z_r + p_r) \\ p_{r+1} = z_r + p_r - y_r \end{cases} \quad \text{and} \quad \begin{cases} z_{r+1} = \operatorname{prox}_{\frac{1}{L_{eff}}g_2}(y_r + q_r) \\ q_{r+1} = y_r + q_r - z_{r+1} \end{cases}, \quad r = 0, 1, 2, \ldots \qquad (26)$$

From the definition of y_r in (26), we obtain $(z_r + p_r - y_r)L_{eff} \in \partial g_1(y_r)$. Combining with Lemma 2.2, we have

$$(z_r + p_r - y_r)L_{eff} \in \partial_{\epsilon_{k,r}} g_1(z_{r+1}) \qquad (27)$$

where

$$\epsilon_{k,r} = g_1(z_{r+1}) - g_1(y_r) - \langle (z_r + p_r - y_r)L_{eff}, z_{r+1} - y_r\rangle \geq 0. \qquad (28)$$

Note that following Theorem 3.3 in [8], both of the sequences $\{y_r\}$ and $\{z_r\}$ converge to $\operatorname{prox}_{\bar{g}}(Z_{eff}) = \operatorname{prox}_{t_k g}(Z)$ then $z_{r+1} - y_r \to 0$. Moreover, $\{y_r\}$ is bounded $(z_r + p_r - y_r)L_{eff}$ in $\partial g_1(y_r)$ [1] is bounded for all r. These follows $\lim\limits_{r\to+\infty} \epsilon_{k,r} = 0$. Additionally, the definition of z_{r+1} in (26) implies

$$(y_r + q_r - z_{r+1})L_{eff} \in \partial g_2(z_{r+1}) \qquad (29)$$

From (27) and (29) we derive

$$(z_r + p_r + q_r - z_{r+1})L_{eff} \in \partial_{\epsilon_{k,r}} \bar{g}(z_{r+1}). \qquad (30)$$

[1] Using [9, Example 3.34], for $U \in \mathbb{R}^{n\times m}$, we have $\partial g_1(U) = \left\{V \in \mathbb{R}^{n\times m} : V^{(i)} \in \partial\left(\lambda_{\text{row}}\|U^{(i)}\|_2\right),\ \forall i = 1, 2, \ldots, n\right\}$, where $\partial\left(\lambda_{\text{row}}\|U^{(i)}\|_2\right) = \begin{cases} \left\{\lambda_{\text{row}}\frac{U^{(i)}}{\|U^{(i)}\|_2}\right\}, & \text{if } U^{(i)} \neq 0, \\ \{a \in \mathbb{R}^m : \|a\|_2 \leq \lambda_{\text{row}}\}, & \text{if } U^{(i)} = 0. \end{cases}$ then $\partial g_1(U)$ is bounded for all $U \in \mathbb{R}^{n\times m}$.

Using $Z_{eff} = \frac{1}{1+\mu t_k}(X^k - \nabla f(X^k))$, from (27), we get that

$$\begin{aligned} &(z_r + p_r + q_r - Z_{eff})L_{eff} \in \partial_{\epsilon_{k,r}}\bar{g}(z_{r+1}) - Z_{eff}L_{eff} + z_{r+1}L_{eff} \\ \iff &(z_r + p_r + q_r - Z_{eff})L_{eff} \in \partial_{\epsilon_{k,r}}\bar{g}(z_{r+1}) + \nabla f(X_k) + L_{eff}\left(z_{r+1} - \frac{1}{1+\mu t_k}X_k\right). \end{aligned} \tag{31}$$

On the other hand, by (26), $p_{r+1}+q_{r+1}+z_{r+1} = p_r+q_r+z_r$ for all $r = 0, 1, 2, ...$ Then, $p_r+q_r+z_r = p_0 + q_0 + z_0 = Z_{eff}$. Therefore we set $v^k = (z_r + p_r + q_r - Z_{eff})L_{eff} = 0$ and it follows from (31) that

$$\begin{aligned} 0 &\in \partial_{\epsilon_{k,r}}\bar{g}(z_{r+1}) + \nabla f(X^k) + L_{eff}\left(z_{r+1} - \frac{1}{1+\mu t_k}X^k\right) \\ &= \frac{1+\mu t_k}{t_k}\left(z_{r+1} - \frac{1}{1+\mu t_k}X^k\right) + \partial_{\epsilon_{k,r}}\bar{g}(z_{r+1}) + \nabla f(X^k) \\ &= \frac{z_{r+1} - X^k}{t_k} + \partial_{\epsilon_{k,r}}g(z_{r+1}) + \nabla f(X^k) \end{aligned} \tag{32}$$

Finally, to choose ε_k satisfying (16), particularly, (21), we find the iteration $\bar{r}$ of the Dykstra-like algorithm (26) so that

$$\epsilon_{k,\bar{r}} \leq \frac{\tau}{(1+\theta_2)t_k}\|X^k - z_{\bar{r}+1}\|_F^2. \tag{33}$$

This condition is well-defined provided that $\tau\|X^k - \text{prox}_{t_k g}(Z)\|_F^2 > 0$ since $\lim_{r\to+\infty} \epsilon_{k,r} = 0$ as previous arguments. Therefore, the triple $(\tilde{X}^k, v^k, \varepsilon_k) = (z_{\bar{r}+1}, 0, \epsilon_{k,\bar{r}})$ has been found successfully by Dykstra-like algorithm and satisfies (15)-(16).

4 The convergence of Algorithm 3.1 for the nonconvex case of (P)

In this section, we establish the convergence of Algorithm 3.1 - NIPGℓ under the nonconvex assumptions, particularly, besides *Assumption 1*, Problem (P) satisfies *Assumption 2* below.

Assumption 2 *f is a Quacavex function, i.e., for $u, v \in \operatorname{dom} g$, 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.

Remark 4.1 As presented in [24,23], the class of *Quacavex* functions includes convex, concave, indefinite quadratic and indefinite quadratic over a positive affine functions. Obviously, this class covers differentiable and convex functions.

Firstly, we prepare some auxiliary lemmas regarding the properties of t_k which play an important role deriving the convergence of NIPGℓ.

Lemma 4.1 *Let $\{t_k\}$ be a sequence of stepsizes generated by Algorithm 3.1. Then,*

(i) there exists $t_{\min} > 0$ such that $t_k \geq t_{\min}$ for all $k \geq 0$;

(ii) $\{t_k\}$ is convergent to $t^* > 0$.

Proof (i) At each iteration k, the stepsize t_{k+1} is updated via one of two branches. In the contraction branch, $t_{k+1} = \frac{c_1\|x^{k+1}-x^k\|^2}{A_\ell(x^k, x^{k+1})}$. Since $A_\ell(x^k, x^{k+1}) \leq L_f\|x^{k+1} - x^k\|^2$ due to the L_f-Lipschitz continuity of ∇f (Definition 3.1), we obtain $t_{k+1} \geq \frac{c_1}{L_f}$. In the expansion branch, $t_{k+1} = (1 + \gamma_k')t_k \geq t_k$. By mathematical induction, $t_k \geq t_{\min} := \min\left\{t_0, \frac{c_1}{L_f}\right\} > 0$ for all $k \geq 0$. Thus, $\inf_{k\geq 0} t_k > 0$.

(ii) Define $r_k = \ln t_{k+1} - \ln t_k = r_k^+ - r_k^-$, where $r_k^+ = \max\{0, r_k\}$ and $r_k^- = -\min\{0, r_k\}$. From the update rules in Algorithm 3.1, we have $r_k = \ln \frac{t_{k+1}}{t_k} \le \ln(1+\gamma_k') \le \gamma_k' \le \gamma_k,\quad$ for all $k \ge 0$. This implies $r_k^+ \le \gamma_k$. Since $\sum\limits_{k=0}^{\infty} \gamma_k < \infty$, the non-negative series $\sum\limits_{k=0}^{\infty} r_k^+$ also converges. By telescoping, we obtain

$$\ln t_{k+1} = \ln t_0 + \sum_{i=0}^{k} r_i^+ - \sum_{i=0}^{k} r_i^-. \tag{34}$$

If the non-negative series $\sum\limits_{k=0}^{\infty} r_k^-$ diverges, then $\lim\limits_{k\to\infty} \ln t_{k+1} = -\infty$, which implies $\lim\limits_{k\to\infty} t_k = 0$. This contradicts $\inf\limits_{k\ge 0} t_k > 0$ established in part (i). Therefore, $\sum\limits_{k=0}^{\infty} r_k^-$ must converge. Consequently, from (34), the sequence $\{\ln t_k\}$ converges, yielding $\lim\limits_{k\to\infty} t_k = t^* \in (0, +\infty)$.

Lemma 4.2 *For Algorithm 3.1 (NIPGℓ, $\ell = 1, 2$), there exists $k_\ell^* \in \mathbb{N}$ such that*

$$A_\ell(x^{k+1}, x^k) \le \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 \quad \forall k \ge k_\ell^*.$$

Consequently, the stepsize sequence is non-decreasing and bounded from above by t^*:

$$0 < \inf_{k\ge 0} t_k \le t_{k_\ell^*} \le t_k \le t_{k+1} \le t^* \quad \forall k \ge k_\ell^*.$$

Proof Suppose by contradiction that condition (17) holds infinitely often. Then, there exists a subsequence $\{k_i\}$ such that $A_\ell(x^{k_i}, x^{k_i+1}) > \frac{c_0}{t_{k_i}}\|x^{k_i+1} - x^{k_i}\|^2$ for all i. From the contraction update rule (18), we have:

$$t_{k_i+1} = \frac{c_1\|x^{k_i+1} - x^{k_i}\|^2}{A_\ell(x^{k_i}, x^{k_i+1})},$$

which implies $\frac{t_{k_i+1}}{t_{k_i}} < \frac{c_1}{c_0}$. By Lemma 4.1, $\lim\limits_{k\to\infty} t_k = t^* > 0$. Taking the limit as $i \to \infty$ yields $1 \le \frac{c_1}{c_0}$, which contradicts the parameter setting $c_1 < c_0$.

Thus, there exists an index k_ℓ^* such that for all $k \ge k_\ell^*$, condition (17) fails, i.e., $A_\ell(x^k, x^{k+1}) \le \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2$. Consequently, for $k \ge k_\ell^*$, the stepsize is updated via the expansion branch (19), yielding $t_{k+1} = (1+\gamma_k')t_k \ge t_k$.

Since the sequence $\{t_k\}_{k\ge k_\ell^*}$ is monotonically non-decreasing and converges to t^*, it is bounded above by its limit. Combining this with the strict lower bound from Lemma 4.1, we conclude:

$$0 < \inf_{k\ge 0} t_k \le t_{k_\ell^*} \le t_k \le t_{k+1} \le t^*, \quad \forall k \ge k_\ell^*.$$

We continue to employ the properties of inexact conditions (15) and (16) in the following lemmas.

Lemma 4.3 *Let $x \in \operatorname{dom} g$ and $k \in \mathbb{N}$ then we have*

$$g(x) \ge g(\tilde{x}^k) + \left\langle \frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k), x - \tilde{x}^k \right\rangle - \epsilon_k. \tag{35}$$

Proof In view of the inexact relative inclusion (15), we have

$$\frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k) \in \partial_{\epsilon_k} g(\tilde{x}^k),$$

and hence the definition of $\partial_\epsilon g$ in (5) implies that

$$g(x) \ge g(\tilde{x}^k) + \left\langle \frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k), x - \tilde{x}^k \right\rangle - \epsilon_k. \tag{36}$$

Lemma 4.4 *For Algorithm 3.1, we have the following assertion:*

$$\left\langle v^k, \tilde{x}^k - x^k \right\rangle + (\theta_1 - 1)\|v^k\|^2 + \theta_2 t_{\min}\varepsilon_k \le \tau\|\tilde{x}^k - x^k\|^2. \tag{37}$$

Proof From Lemma 4.3 (inequality (35)) we substitute x by $\tilde{x}^k - v^k$ to have that

$$g(\tilde{x}^k - v^k) \ge g(\tilde{x}^k) + \left\langle \frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k), -v^k \right\rangle - \varepsilon_k. \tag{38}$$

Next, plugging (38) into (16), we get that

$$\left\langle v^k, \frac{\tilde{x}^k - x^k}{t_k} \right\rangle + \frac{\theta_1 - 1}{t_k}\|v^k\|^2 + \theta_2\varepsilon_k \le \frac{\tau}{t_k}\|\tilde{x}^k - x^k\|^2$$

which is equivalent to

$$\left\langle v^k, \tilde{x}^k - x^k \right\rangle + (\theta_1 - 1)\|v^k\|^2 + \theta_2 t_k\varepsilon_k \le \tau\|\tilde{x}^k - x^k\|^2. \tag{39}$$

Now, remember that $t_k \ge t_{\min}$ for all $k \ge 0$ in Lemma 4.1 then it follows from inequality (39) that

$$\left\langle v^k, \tilde{x}^k - x^k \right\rangle + (\theta_1 - 1)\|v^k\|^2 + \theta_2 t_{\min}\varepsilon_k \le \tau\|\tilde{x}^k - x^k\|^2.$$

Lemma 4.5 *Algorithm 3.1 (NIPGℓ, $\ell = 1, 2$) stops at the iteration k if and only if x^k is a stationary point of Problem* (P).

Proof We divide the proof into two parts.
Part 1: For the necessary condition, we suppose that $x^k = \tilde{x}^k$ then from (37) we have

$$(\theta_1 - 1)\|v^k\|^2 + \theta_2 t_{\min}\varepsilon_k \le 0. \tag{40}$$

But in Algorithm 3.1, $\theta_1 > 1$, $\theta_2 \ge 1$ and from Lemma 4.1 we have $t_{\min} > 0$ then we obtain that $v^k = 0$ and $\varepsilon_k = 0$ which combines (15) to get $0 \in \nabla f(x^k) + \partial g(x^k)$, i.e., x^k is a stationary point of Problem (P).
Part 2: For the sufficient condition, we suppose that x^k is a stationary point of Problem (P), i.e., $-\nabla f(x^k) \in \partial g(x^k)$. Additionally, the inexact relative inclusion (15) follows

$$\frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k) \in \partial_{\epsilon_k} g(\tilde{x}^k).$$

Now, using the relation (6) we derive that

$$\left\langle \frac{v^k + x^k - \tilde{x}^k}{t_k}, \tilde{x}^k - x^k \right\rangle \ge -\varepsilon_k$$

which derives

$$\langle v^k, \tilde{x}^k - x^k \rangle \ge \|x^k - \tilde{x}^k\|^2 - \varepsilon_k t_k. \tag{41}$$

Next, combining (39) with (41) we get the following inequality

$$(1 - \tau)\|x^k - \tilde{x}^k\|^2 + (\theta_1 - 1)\|v^k\|^2 + (\theta_2 - 1)t_k\varepsilon_k \le 0. \tag{42}$$

The conditions on τ, θ_1, θ_2 in Algorithm 3.1 derive that $x^k = \tilde{x}^k$ and $v^k = 0$. Utilizing (16), we obtain $\varepsilon_k = 0$. Finally, we have the desired conclusion that Algorithm 3.1 (NIPGℓ) stops at the k^{th} iteration.

In the sequel, we intend to analyze the convergence of Algorithm 3.1 which strongly depend on the descent property of $\{F(x^k)\}_{k \ge k_\ell^*}$ presented in the following lemma.

Lemma 4.6 (Descent Lemma) *Assuming that Problem (P) satisfies Assumptions 1 and 2 then the sequence $\{x^k\}$ generated by NIPGℓ, $\ell = 1, 2$ (Algorithm 3.1) has the following property*

$$F(x^k) - F(x^{k+1}) \geq \frac{\left(\theta_1 - \frac{1}{2}\right)\|v^k\|^2 + \left(\frac{1}{2} - \tau\right)\|x^k - \tilde{x}^k\|^2 + \left(\frac{1}{2} - c_0\right)\|x^{k+1} - x^k\|^2}{t_k} + \theta_2\varepsilon_k, \ \forall k \geq k_\ell^*. \tag{43}$$

Proof By using the Fundamental Theorem of Calculus, we have

$$\begin{aligned} f(x^{k+1}) - f(x^k) &= \int_0^1 \left\langle \nabla f(x^k + t(x^{k+1} - x^k)), x^{k+1} - x^k \right\rangle dt \\ &= \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \int_0^1 u_k(t)dt, \ \forall k \geq 0, \end{aligned} \tag{44}$$

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 2, the quasiconvexity of $u_k(t)$ in $[0, 1]$ follows that for all $t \in [0, 1]$,

$$\begin{aligned} u_k(t) &\leq \max\{u_k(0), u_k(1)\} = \max\{0, u_k(1)\} \leq |u_k(1)| \\ &= |\langle \nabla f(x^{k+1}) - \nabla f(x^k), x^{k+1} - x^k \rangle| \\ &\leq A_\ell(x^k, x^{k+1}) \ \forall k \geq k_\ell^*, \ell = 1, 2. \end{aligned}$$

Thereafter, using Lemma 4.2, we derive that

$$\int_0^1 u_k(t)dt \leq \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2, \ \ \forall k \geq k_\ell^*, \ \ell = 1, 2. \tag{45}$$

Now, combining (44), (45) and Lemma 4.3 with $x = x^k$ we get that

$$\begin{aligned} F(x^k) - F(x^{k+1}) &= f(x^k) - f(x^{k+1}) + g(x^k) - g(x^{k+1}) \\ &\geq -\left\langle x^{k+1} - x^k, \nabla f(x^k) \right\rangle - \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + g(\tilde{x}^k) + \\ &+ \left\langle \frac{v^k + x^k - \tilde{x}^k}{t_k} - \nabla f(x^k), x^k - \tilde{x}^k \right\rangle - \epsilon_k - g(x^{k+1}) \quad \forall k \geq k_\ell^*, \ell = 1, 2. \end{aligned} \tag{46}$$

Moreover, from Algorithm 3.1, $v^k = \tilde{x}^k - x^{k+1}$ and relation (16) we rewrite (46) in the following form

$$\begin{aligned} F(x^k) - F(x^{k+1}) &\geq -g(\tilde{x}^k - v^k) + g(\tilde{x}^k) + \left\langle v^k, \nabla f(x^k) \right\rangle - \epsilon_k - \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \\ &+ \left\langle \frac{x^k - x^{k+1}}{t_k}, x^k - \tilde{x}^k \right\rangle \\ &\underset{\text{by(16)}}{\geq} \frac{\theta_1}{t_k}\|v^k\|^2 + \theta_2\varepsilon_k - \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2 - \frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \\ &\quad + \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^{k+1} - \tilde{x}^k\|^2 + \|x^{k+1} - x^k\|^2\right) \\ &\geq \frac{\left(\theta_1 - \frac{1}{2}\right)\|v^k\|^2 + \left(\frac{1}{2} - \tau\right)\|x^k - \tilde{x}^k\|^2 + \left(\frac{1}{2} - c_0\right)\|x^{k+1} - x^k\|^2}{t_k} + \theta_2\varepsilon_k, \ \forall k \geq k_\ell^*. \end{aligned} \tag{47}$$

The proof is finished.

The main convergence results of Algorithm 3.1 is stated in the following theorem for the nonconvex case of Problem (P).

Theorem 4.1 *Under Assumptions 1 and 2, the following assertions hold for Algorithm 3.1:*

(i) The sequence $\{F(x^k)\}_{k\geq k_\ell^*}$ is non-increasing and converges; and $F(x^{k+1}) < F(x^k)$ unless x^k is a stationary point of Problem (P).

(ii) $\sum_{k=1}^{+\infty}\|x^k-\tilde{x}^k\|^2$ and $\sum_{k=1}^{+\infty}\|x^k-x^{k+1}\|^2$ are convergent.

(iii) Any cluster point of $\{x^k\}$ is a stationary point of (P).

(iv) For $K > k_\ell^*$, there exists $i_K \in \{k_\ell^*, ..., K\}$ such that

$$\|w^{i_K}\| \leq \sqrt{\frac{F(x^{k_\ell^*})-F_*}{\Gamma(K-k_\ell^*+1)}} = O\left(\frac{1}{\sqrt{K}}\right), \quad \text{and } \varepsilon_{i_K} \leq \frac{F(x^{k_\ell^*})-F_*}{\theta_2(K-k_\ell^*+1)} = O\left(\frac{1}{K}\right), \tag{48}$$

where the definition of w^k is in Remark 3.1 (i). Consequently, for $\eta > 0$, we can obtain η- stationary point of (P) *after at most $k_\ell^* + O\left(\frac{1}{\eta^2}\right)$ iterations.*

Proof (i) From (43) and $\theta_1 > 1$, $\tau \in [0, \frac{1}{2})$ and $0 < c_0 \leq \frac{1}{2}$ we derive that the sequence $\{F(x^k)\}_{k\geq k_\ell^*}$ is non-increasing. However, by Assumption 1, $\{F(x^k)\}_{k\geq k_\ell^*}$ is lower bounded by F_* hence it converges to $\hat{F} \geq F_*$. Now, if $F(x^k) = F(x^{k+1})$ for some $k \geq k_\ell^*$ then from (47) we obtain $v^k = 0$, $x^k = \tilde{x}^k$, $\varepsilon_k = 0$ and hence from Lemma (4.5), x^k is a stationary point of (P).

(ii) Now, remember that by Lemma 4.2, $t_k \leq t^*$ for all $k \geq k_\ell^*$ then it follows from (43) that

$$\sum_{k=k_\ell^*}^{K}\|x^{k+1}-x^k\|^2 \leq \frac{t^*}{\frac{1}{2}-c_0}(F(x^{k_\ell^*})-F(x^{K+1})), \text{ for all } K \geq k_\ell^*+1 \tag{49}$$

and

$$\sum_{k=k_\ell^*}^{K}\|x^k-\tilde{x}^k\|^2 \leq \frac{t^*}{\frac{1}{2}-\tau}(F(x^{k_\ell^*})-F(x^{K+1})) \quad \text{for all } K \geq k_\ell^*+1. \tag{50}$$

Since $F(x^{K+1}) \geq \hat{F}$ for all $K \geq k_\ell^*+1$, tending K to infinity we obtain

$$\sum_{k=k_\ell^*}^{\infty}\|x^{k+1}-x^k\|^2 \leq \frac{t^*}{\frac{1}{2}-c_0}(F(x^{k_\ell^*})-\hat{F}) \tag{51}$$

and

$$\sum_{k=k_\ell^*}^{\infty}\|x^k-\tilde{x}^k\|^2 \leq \frac{t^*}{\frac{1}{2}-\tau}(F(x^{k_\ell^*})-\hat{F}). \tag{52}$$

Hence, both of the given series are convergent.

(iii) Next, suppose that $\bar{x}$ is a cluster point of $\{x^k\}$ then there exists a subsequence $\{x^{k_j}\}$ which converges to $\bar{x}$. Note that, the convergence of $\sum_{k=k_\ell^*}^{\infty}\|x^k-\tilde{x}^k\|^2$ follows $\lim_{k\to\infty}\|x^k-\tilde{x}^k\| = 0$. Moreover,

$$\|\tilde{x}^{k_j}-\bar{x}\| \leq \|\tilde{x}^{k_j}-x^{k_j}\| + \|x^{k_j}-\bar{x}\|$$

which derives that $\lim_{k_j\to\infty}\|\tilde{x}^{k_j}-\bar{x}\|=0$. On the other hand, from (i), the left hand side of (47) comes to zeros as k tending to infinity which means that $\lim_{k\to\infty}\|v^k\|=0$ and $\lim_{k\to\infty}\varepsilon_k=0$. In relation (36), substitute k by k_j we have

$$\frac{v^{k_j}+x^{k_j}-\tilde{x}^{k_j}}{t_{k_j}}-\nabla f(x^{k_j})\in\partial_{\epsilon_{k_j}}g(\tilde{x}^{k_j}). \tag{53}$$

Now, using the continuity of ∇f, $\lim_{k\to\infty}t_k=t^*$ (Lemma (4.1)), the closed graph property of $\partial_\varepsilon g$ in Lemma (2.1), we tend k_j in (53) to infinity and obtain that

$$-\nabla f(\bar{x})\in\partial g(\bar{x}), \tag{54}$$

i.e., $\bar{x}$ is a stationary point of (P).

(iv) From (14) and the global Lipschitzness of ∇f we have

$$\|w^k\|=\left\|\frac{v^k-\tilde{x}^k+x^k}{t_k}-\nabla f(x^k)+\nabla f(\tilde{x}^k)\right\| \tag{55}$$

$$\leq\frac{\|v^k\|}{t_k}+\left(\frac{1}{t_k}+L_f\right)\|x^k-\tilde{x}^k\|. \tag{56}$$

It follows that

$$\begin{aligned}\|w^k\|^2&\leq\left(\frac{1}{\sqrt{t_k(\theta_1-\frac{1}{2})}}\frac{\sqrt{\theta_1-\frac{1}{2}}\|v^k\|}{\sqrt{t_k}}+\frac{1+L_ft_k}{\sqrt{t_k(\frac{1}{2}-\tau)}}\frac{\sqrt{\frac{1}{2}-\tau}\|x^k-\tilde{x}^k\|}{\sqrt{t_k}}\right)^2\\&\leq\left(\frac{1}{t_k\left(\theta_1-\frac{1}{2}\right)}+\frac{(1+L_ft_k)^2}{t_k\left(\frac{1}{2}-\tau\right)}\right)\left(\frac{\left(\theta_1-\frac{1}{2}\right)\|v^k\|^2}{t_k}+\frac{\left(\frac{1}{2}-\tau\right)\|x^k-\tilde{x}^k\|^2}{t_k}\right).\end{aligned} \tag{57}$$

Again, by Lemma 4.2, $t_{k_\ell^*}\leq t_k\leq t^*$ for all $k\geq k_\ell^*$, therefore

$$\begin{aligned}\frac{\left(\theta_1-\frac{1}{2}\right)\|v^k\|^2\left(\frac{1}{2}-\tau\right)\|x^k-\tilde{x}^k\|^2}{t_k}&\geq\frac{t_k\left(\frac{1}{2}-\tau\right)\left(\theta_1-\frac{1}{2}\right)}{\frac{1}{2}-\tau+\left(\theta_1-\frac{1}{2}\right)(1+L_ft_k)^2}\|w^k\|^2\\&\geq\frac{t_{k_\ell^*}\left(\frac{1}{2}-\tau\right)\left(\theta_1-\frac{1}{2}\right)}{\frac{1}{2}-\tau+\left(\theta_1-\frac{1}{2}\right)(1+L_ft^*)^2}\|w^k\|^2\\&\geq\Gamma\|w^k\|^2,\end{aligned} \tag{58}$$

where

$$\Gamma=\frac{t_{k_\ell^*}\left(\frac{1}{2}-\tau\right)\left(\theta_1-\frac{1}{2}\right)}{\frac{1}{2}-\tau+\left(\theta_1-\frac{1}{2}\right)(1+Lt^*)^2}.$$

Combining with (43) we derive that

$$F(x^k)-F(x^{k+1})\geq\Gamma\|w^k\|^2+\frac{\frac{1}{2}-c_0}{t^*}\|x^{k+1}-x^k\|^2+\theta_2\varepsilon_k,\ \forall k\geq k_\ell^*. \tag{59}$$

Hence, taking some $K\geq k_\ell^*+1$ and summing (59) from $k=k_\ell^*$ to K we obtain that

$$F(x^{k_\ell^*})-F(x^{K+1})\geq\sum_{k=k_\ell^*}^{K}\left(\Gamma\|w^k\|^2+\frac{\frac{1}{2}-c_0}{t^*}\|x^{k+1}-x^k\|^2+\theta_2\varepsilon_k\right). \tag{60}$$

On the other hand, since $F(x^{K+1})\geq F_*$ hence we have that

$$(K-k_\ell^*+1)\min_{k=k_\ell^*,\dots,K}\left(\Gamma\|w^k\|^2+\frac{\frac{1}{2}-c_0}{t^*}\|x^{k+1}-x^k\|^2+\theta_2\varepsilon_k\right)\leq F(x^{k_\ell^*})-F_*. \tag{61}$$

Therefore, there exists $i_K \in \{k_\ell^*, ..., K\}$ such that

$$\|w^{i_K}\| \leq \sqrt{\frac{F(x^{k_\ell^*}) - F_*}{\Gamma(K - k_\ell^* + 1)}} O\left(\frac{1}{\sqrt{K}}\right), \text{ and } \varepsilon_{i_K} \leq \frac{F(x^{k_\ell^*}) - F_*}{\theta_2(K - k_\ell^* + 1)} = O\left(\frac{1}{K}\right), \quad K \geq k_\ell^* \quad (62)$$

and hence we can obtain η- stationary point after at most $k_\ell^* + O\left(\frac{1}{\eta^2}\right)$ iterations.

5 The convergence of of Algorithm 3.1 for the convex case of (P)

In this section we continue to provide stronger convergence results of Algorithm 3.1 when f is convex. We concrete this supposition in the following assumption.

Assumption 3 $f : \mathbb{E} \to (-\infty, +\infty]$ *is proper, closed and convex.*

Obviously, if f satisfies Assumptions 1 and 3 then it must match Assumptions 1 and 2, i.e., we inherit the convergence results of Algorithm 3.1 in Section 4 for the convex case of Problem (P) in this section.

To obtain the convergence of $\{x^k\}$ to some optimal solution $x^* \in S^*$ we need to establish a general inequality that can cover the inequality in Lemma 4.6. The details are in the following.

Lemma 5.1 *For every* $x \in \mathrm{dom}\ g$ *and* $k \in \mathbb{N}$*, we have*

$$\begin{aligned} F(x) - F(x^{k+1}) \geq &-\frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^k - x\|^2 + \|x^{k+1} - x\|^2\right) \\ &- \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2 + \left(\frac{\theta_1}{t_k} - \frac{1}{2t_k}\right)\|v^k\|^2 + \theta_2 \epsilon_k. \end{aligned} \quad (63)$$

Proof Since f is convex, we have

$$f(x) - f(x^k) \geq \left\langle \nabla f(x^k), x - x^k \right\rangle.$$

Combining with (35) and remember $x^{k+1} = \tilde{x}^k - v^k$, we obtain that

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

Hence,

$$\begin{aligned} F(x) - F(x^k) + F(x^k) - F(x^{k+1}) \geq &\ g(x^k) - g(x^{k+1}) + f(x^k) - f(x^{k+1}) + g(\tilde{x}^k) - g(x^k) \\ &+ \left\langle \nabla f(x^k), \tilde{x}^k - x^k \right\rangle + \left\langle \frac{x^k - x^{k+1}}{t_k}, x - \tilde{x}^k \right\rangle - \epsilon_k. \end{aligned} \quad (64)$$

Again, substituting the relation that $x^{k+1} = \tilde{x}^k - v^k$ into (64), it can be rewritten as

$$\begin{aligned} F(x) - F(x^{k+1}) \geq &\underbrace{g(\tilde{x}^k) - g(\tilde{x}^k - v^k) + \left\langle \nabla f(x^k), v^k \right\rangle - \epsilon_k}_{A} + \underbrace{\left\langle \nabla f(x^k), x^{k+1} - x^k \right\rangle + f(x^k) - f(x^{k+1})}_{B} + \\ &+ \left\langle \frac{x^k - x^{k+1}}{t_k}, x - \tilde{x}^k \right\rangle. \end{aligned} \quad (65)$$

On the other hand, from (16), we have

$$A = g(\tilde{x}^k) - g(\tilde{x}^k - v^k) + \left\langle \nabla f(x^k), v^k \right\rangle - \epsilon_k \geq \frac{\theta_1}{t_k}\|v^k\|^2 + \theta_2\epsilon_k - \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2. \tag{66}$$

Moreover, the convexity of f and the result of Lemma 4.2 implies that

$$\begin{aligned} B = f(x^k) - f(x^{k+1}) + \left\langle \nabla f(x^k), x^{k+1} - x^k \right\rangle &\geq \left\langle \nabla f(x^{k+1}), x^k - x^{k+1} \right\rangle + \left\langle \nabla f(x^k), x^{k+1} - x^k \right\rangle \\ &\geq -\left| \left\langle \nabla f(x^{k+1}) - \nabla f(x^k), x^k - x^{k+1} \right\rangle \right| = -|A_1(x^k, x^{k+1})| \\ &\geq -\|\nabla f(x^{k+1}) - \nabla f(x^k)\|\|x^k - x^{k+1}\| = -|A_2(x^k, x^{k+1})| \\ &\geq -\frac{c_0}{t_k}\|x^{k+1} - x^k\|^2, \ \forall k \geq k_\ell^*. \end{aligned} \tag{67}$$

Combine (65), (66), (67) with the identity

$$\begin{aligned} \left\langle \frac{x^k - x^{k+1}}{t_k}, x - \tilde{x}^k \right\rangle &= \frac{1}{t_k}\left(\left\langle x^k - \tilde{x}^k, x - \tilde{x}^k \right\rangle + \left\langle \tilde{x}^k - x^{k+1}, x - \tilde{x}^k \right\rangle \right) \\ &= \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 + \|x - \tilde{x}^k\|^2 - \|x^k - x\|^2 - \|\tilde{x}^k - x^{k+1}\|^2 - \|x - \tilde{x}^k\|^2 + \|x^{k+1} - x\|^2 \right) \\ &= \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^k - x\|^2 - \|\tilde{x}^k - x^{k+1}\|^2 + \|x^{k+1} - x\|^2 \right), \end{aligned}$$

we have

$$\begin{aligned} F(x) - F(x^{k+1}) &\geq -\frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^k - x\|^2 - \|\tilde{x}^k - x^{k+1}\|^2 + \|x^{k+1} - x\|^2 \right) \\ &\quad - \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2 + \frac{\theta_1}{t_k}\|v^k\|^2 + \theta_2\epsilon_k \\ &= -\frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^k - x\|^2 + \|x^{k+1} - x\|^2 \right) \\ &\quad - \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2 + \left(\frac{\theta_1}{t_k} - \frac{1}{2t_k} \right)\|v^k\|^2 + \theta_2\epsilon_k, \quad \forall k \geq k_\ell^*. \end{aligned}$$

The desired conclusion is obtained then.

Theorem 5.1 *The sequence $\{x^k\}_{k\in\mathbb{N}}$ generated by the NIPGℓ method converges to an optimal solution x^* of Problem* (P). *Moreover,*

$$F(x^K) - F_* \leq O\left(\frac{1}{K}\right), \ \forall K \geq k_\ell^*. \tag{68}$$

Proof Using Lemma 5.1 at $x = x^* \in S^* \subseteq \operatorname{dom} g$ we have

$$\begin{aligned} F(x^*) - F(x^{k+1}) \geq &-\frac{c_0}{t_k}\|x^{k+1} - x^k\|^2 + \frac{1}{2t_k}\left(\|x^k - \tilde{x}^k\|^2 - \|x^k - x^*\|^2 + \|x^{k+1} - x^*\|^2 \right) \\ &- \frac{\tau}{t_k}\|x^k - \tilde{x}^k\|^2 + \left(\frac{\theta_1}{t_k} - \frac{1}{2t_k} \right)\|v^k\|^2 + \theta_2\epsilon_k, \ \forall k \geq k_\ell^*, \end{aligned}$$

equivalently,

$$\begin{aligned} \|x^{k+1} - x^*\|^2 \leq &\|x^k - x^*\|^2 + 2c_0\|x^{k+1} - x^k\|^2 - (1 - 2\tau)\|x^k - \tilde{x}^k\|^2 - (2\theta_1 - 1)\|v^k\|^2 - \theta_2\epsilon_k t_k \\ &+ 2t_k(F(x^*) - F(x^{k+1})), \ \forall k \geq k_\ell^*. \end{aligned} \tag{69}$$

On the other hand, $F(x^*) - F(x^{k+1}) \leq 0$, $\tau < \frac{1}{2}$, $\theta_1 > 1$, $\theta_2 \geq 1$ then we have

$$\|x^{k+1} - x^*\|^2 \leq \|x^k - x^*\|^2 + 2c_0\|x^{k+1} - x^k\|^2, \ \forall k \geq k_\ell^*.$$

From Theorem 4.1, $\sum_{k=k_\ell^*}^{\infty} 2c_0\|x^{k+1}-x^k\|^2$ is convergent then $\{x^k\}_{k\geq k_\ell^*}$ is quasi-Fejér convergent to S^* (by Definition 2.3). Moreover, from Theorem 4.1 (iii), any cluster point of $\{x^k\}_{k\geq 1}$ is a stationary point of (P), i.e., is an optimal solution of (P). In other words, all accumulation of (P) belong to S^*. Hence, by Lemma 2.3, the sequence $\{x^k\}_{k\geq 1}$ converges to some $\bar{x} \in S^*$.

For the second part of the proof, from (69) we have that

$$F(x^{k+1}) - F(x^*) \leq \|x^k - x^*\|^2 - \|x^{k+1} - x^*\|^2 + 2c_0\|x^{k+1} - x^k\|^2, \ \forall k \geq k_\ell^*. \tag{70}$$

Summing up (70) from $k = k_\ell^*$ to $K \geq k_\ell^* + 1$, we get that

$$\begin{aligned}\sum_{k=k_\ell^*}^{K} \Big(F(x^{k+1}) - F(x^*)\Big) &\leq \|x^{k_\ell^*} - x^*\|^2 - \|x^{K+1} - x^*\|^2 + \sum_{k=k_\ell^*}^{K} 2c_0\|x^{k+1} - x^k\|^2 \\ &\leq \|x^{k_\ell^*} - x^*\|^2 + \sum_{k=k_\ell^*}^{\infty} 2c_0\|x^{k+1} - x^k\|^2. \end{aligned}\tag{71}$$

Next, using Theorem (4.1) (i), $F(x^k) \geq F(x^{k+1})$ for all $k \geq k_\ell^*$ then we have that

$$(K-k_\ell^*+1)(F(x^K)-F*) \leq \sum_{k=k_\ell^*}^{K} \Big(F(x^{k+1}) - F(x^*)\Big) \leq \|x^{k_\ell^*}-x^*\|^2 + \sum_{k=k_\ell^*}^{\infty} 2c_0\|x^{k+1}-x^k\|^2, \ \forall K \geq k_\ell^*+1. \tag{72}$$

Setting $D = \|x^{k_\ell^*} - x^*\|^2 + \sum_{k=k_\ell^*}^{\infty} 2c_0\|x^{k+1} - x^k\|^2 \in (0, +\infty)$, we thus have

$$F(x^K) - F_* \leq \frac{D}{K - k_\ell^* + 1} = O\left(\frac{1}{K}\right), \ \forall K \geq k_\ell^*. \tag{73}$$

6 Numerical experiments

In the numerical experiments, we evaluate the performance of our proposed NIPGℓ, $\ell = 1, 2$ by comparing them with three best algorithms in [10] including IPG-ELS, PG-ELS and IPG-FixStep. We use the same parameter setting in [10] for the these ones while for NIPG1 and NIPG2, the parameters are chosen as follows: $c_0 = 0.5$, $c_1 = 0.49$, $\tau = 0.2$, $\theta_1 = 1.1$, $\theta_2 = 1$; $\gamma_k = \frac{0.5(\ln(k+1))^{5.7}}{(k+1)^{1.1}}$. All algorithms were implemented in Python and executed on a personal computer equipped with an 11th Gen Intel Core i5-11400H processor (2.70 GHz, 6 cores, 12 threads) and 16 GB of RAM.To verify the performance, our code and datasets are available online at this link `https://github.com/hoaiphamthi/NIPG/`.

Tested instances: We use CUR-like factorization problems and indefinite quadratic programming problems with elastic net regularization for testing. In particular, these two following problems.

CUR-like factorization problem [6]: Consider $\mathbb{E} = \mathbb{R}^{n\times m}$, given a matrix $W \in \mathbb{R}^{m\times n}$, we want to solve the minimization

$$\min_{X\in\mathbb{R}^{n\times m}} \left\{ \frac{1}{2}\|W - WXW\|_F^2 + \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2 + \frac{\mu}{2}\|X\|_F^2 \right\}, \tag{74}$$

where $\|\cdot\|_F$ is the Frobenius norm, and where $X^{(i)}$ and $X_{(j)}$ respectively denote the i-th row and the j-th column of the matrix X. Problem (74) is a kind of (P) with $f(X) = \frac{1}{2}\|W - WXW\|_F^2$

and $g(X) = \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X^{(j)}\|_2 + \frac{\mu}{2}\|X\|_F^2$ and f and g are convex. The gradient of f is given by $\nabla f(X) = W^T(WXW - W)W^T$ and has a Lipschitz constant $L_f = \|W^T W\|_F^2$. It is worth noting that, Problem (74) is convex and satisfies Assumptions 1 and 3. This problem with $\mu = 0$ has already been used in [10] for illustrating convergence guarantees of a relative inexact proximal gradient method with an explicit line search. We utilize six datasets from various public repositories[2,3]. The input data matrix for the CUR-like factorization problem is preprocessed to systematically control the Lipschitz constant of the gradient. By using the similar way in [10], for each dataset, we scale the matrix W by a factor $\rho > 0$ to change the Lipschitz constant. This strategy allows us to intentionally generate multiple variants of the same dataset with drastically different Lipschitz constants therefore can rigorously evaluate the robustness and computational efficiency of the algorithms under varying degrees of objective curvature. We use $\lambda_{\text{row}} = \lambda_{\text{col}} = 0.01$ and $\mu \in \{0; 0.01\}$ for the experiments.

Indefinite quadratic programming problem with group lasso regularization (IQPP) has the following formulation:

$$\min_{X \in \mathbb{R}^{n\times m}} \left\{ \frac{1}{2}\text{tr}(X^T AX) + \text{tr}(B^T X) + \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2 \right\}, \tag{IQPP}$$

where $A \in \mathbb{R}^{n\times n}$ is a real symmetric and indefinite matrix, $X, B \in \mathbb{R}^{n\times m}$, and where $X^{(i)}$ and $X_{(j)}$ respectively denote the i-th row and the j-th column of the matrix X. Problem (IQPP) belongs to class of composite optimization problems (P) with $f(X) = \frac{1}{2}\text{tr}(X^T AX) + \text{tr}(B^T X)$ and $g(X) = \lambda_{\text{row}} \sum_{i=1}^{n} \|X^{(i)}\|_2 + \lambda_{\text{col}} \sum_{j=1}^{m} \|X_{(j)}\|_2$. We have $\nabla f(X) = AX + B$ is Lipschitz continuous with constant $L_f = \|A\|_2 = \max_{1\le i\le n} |\lambda_i(A)|$, where $\lambda_i(A)$, $i = 1, ..., n$ are eigenvalues of A. The indefinite quadratic programming problem uses randomly generated data. To ensure the indefinite property of the symmetric matrix $A \in \mathbb{R}^{n\times n}$, we construct this matrix via the spectral decomposition $A = U\text{diag}(\lambda_1, \ldots, \lambda_n)U^\top$, where the orthogonal matrix $U \in \mathbb{R}^{n\times n}$ is obtained from the QR decomposition of a standard normal matrix. The indefinite property is strictly guaranteed by fixing the proportion of negative eigenvalues at exactly 15%. Specifically, 15% of the eigenvalues are drawn from a uniform distribution $\mathcal{U}(-5, -0.1)$, while the remaining positive eigenvalues are drawn from $\mathcal{U}(0.1, L_{\max})$, with $L_{\max} \in \{30, 80\}$. With this type of data, Problem (IQPP) is *nonconvex* and satisfies Assumptions 1 and 2. We use $\lambda_{\text{row}} = \lambda_{\text{col}} = 0.01$ for IQPP experiment.

Inexact proximal computation for NIPG1 and NIPG2: We use the inner procedure based on Dykstra-like algorithm [8] to find an inexact proximal solution as presented in Example 3.1.

Stopping criteria for all algorithms: We use the same strategy presented in [10]. Particularly, we first execute IPG-ELS til 101 outer iterations to establish a baseline performance metric, defined as $F_* := F(X^{101})$. Then, the other four comparative algorithms terminate as soon as their objective values reach F_* or after reaching the limitation of 2001 iterations.

We present the numerical results in Tables 1, 2, 3, 4, 5 and Figures 1, 2. To compare the performance of the algorithms, we use the following metrics:

- $F(X^k)$: The value of F at the last iteration.
- Out-IT: The number of outer iterations.

---

[2] ELVIRA biomedical data set repository: `https://leo.ugr.es/elvira/DBCRepository/`

[3] CINA dataset: `http://www.causality.inf.ethz.ch/data/CINA.html`

- In-IT: The cumulative number of inner iterations of the Dykstra-like algorithms to compute an inexact proximal solution.
- ELS-IT: The total number of line search backtracking steps (marked as "–" for methods without line search).
- Time (s): The total CPU execution time in seconds.

Specifically, Tables 1 and 2 show the results for the convex CUR-like factorization with $\mu = 0$. Tables 3 and 4 present the results for the convex CUR-like factorization case with $\mu = 0.01$. The nonconvex results are summarized in Table 5. For each metric, we emphasize the best result by bold characters. Figures 1–2 illustrate the convergence curves over time and the trajectories of our adaptive stepsizes for a tested data of CUR-like factorization and IQPP problems, respectively. The experimental results in Tables 1, 2, 3, 4, 5 show that enforcing exact solutions in PG-ELS creates a huge bottleneck compared with inexact algorithms. Additionally, the proposed NIPG1 and NIPG2 algorithms demonstrate superior efficiency, achieving the best values of F, the fewest outer iterations, and, most notably, the shortest running times. For many datasets, our adaptive methods require substantially fewer outer iterations, leading to CPU times that are up to approximately 10 times shorter than those of the other inexact algorithms. Even for datasets where NIPG1 and NIPG2 require more iterations than IPG-ELS (e.g., Lung Cancer ($\rho = 4, 8$, $\mu = 0$) and Cina0 ($\rho = 8$, $\mu = 0$)), our methods still achieve shorter running times. This is because the adaptive stepsize strategies used in NIPG1 and NIPG2 incur only a low computational cost at each iteration.

Fixed-stepsize methods like IPG-FixStep are extremely slow when the Lipschitz constant L_f is large, because they are locked into a tiny stepsize $(1/L_f)$. For instance, on the Lung Cancer dataset with $\rho = 8$ (Table 2), IPG-FixStep hits the maximum limit of 2001 iterations. Meanwhile, NIPG1 and NIPG2 dynamically increase their stepsizes and successfully converge in 338 iterations (114.46 seconds).

It is worth noting that for both NIPG variants, the cumulative inner iterations (In-IT) remain nearly identical to the outer iterations (Out-IT) across several datasets of benchmark and synthetic. For instance, in Table 1 (where $\mu = 0$), on the Colon Tumor dataset with $\rho = 4$, NIPG1 requires exactly 57 outer and 57 inner iterations, while NIPG2 requires 123 outer and 123 inner iterations. Similarly, in Table 2, for the Secom dataset with $\rho = 4$, we see 67 Out-IT and 67 In-IT for NIPG1, and 123 Out-IT and 123 In-IT for NIPG2. This exact match also extends to the nonconvex setting in Table 5; for example, on the 250×2500 synthetic dataset with Lipschitz constant ≈ 80, NIPG1 needs 63 Out-IT and 63 In-IT, while NIPG2 needs 64 Out-IT and 64 In-IT. This phenomenon is mathematically attributed to the presence of the adaptive stepsize t_k in the denominator of the relative error threshold $\frac{\tau}{(1+\theta_2)t_k}\|X^k - z_{r+1}\|_F^2$. During the initial phases, the adaptive strategy often generates relatively small stepsizes t_k, which significantly increases the allowed error threshold on the right-hand side. Consequently, because the Dykstra-like subroutine converges very quickly and produces high-precision approximations even at the first inner iteration ($\bar{r} = 0$), the stopping criterion is immediately satisfied at each outer step, leading to In-IT $\approx$ Out-IT.

Figures 1–2 show that the objective errors of NIPG1 and NIPG2 drop almost vertically, reaching high precision much faster than other methods. Moreover, NIPG1 provides the bigger step lengths than NIPG2 (which can be explained partly from the notice that $A_1(x, y) \leq A_2(x, y)$ and the formulation of t_{k+1} in (18)). The effect of this phenomenon is that NIPG1 consistently outperforms NIPG2 in terms of both CPU time and the number of iterations on most of tested instances.

Table 1: Numerical comparison on benchmark datasets (Colon tumor (62×2000), Heart Disease (303×14), Central Nervous System (CNS) (60×7129)) for the CUR-like factorization problem (74) with $\mu = 0$

Problem	Lips	Method	$F(X^k)$	Out-IT	In-IT	ELS-IT	Time (s)
Colon Tumor ($\rho = 2$)	8.2139	IPG-ELS	0.6474	101	259	164	15.31
		PG-ELS	0.6474	102	4418	160	59.62
		IPG-FixStep	0.6474	347	348	–	14.77
		NIPG1	**0.6469**	**53**	75	–	**2.37**
		NIPG2	0.6473	72	122	–	3.68
Colon Tumor ($\rho = 4$)	131.4225	IPG-ELS	1.6166	101	108	582	28.18
		PG-ELS	1.6161	108	579	605	33.06
		IPG-FixStep	1.6164	453	454	–	28.42
		NIPG1	**1.6027**	**57**	57	–	**2.85**
		NIPG2	1.6152	123	123	–	6.59
Colon Tumor ($\rho = 8$)	2102.7607	IPG-ELS	3.4234	101	101	990	20.68
		PG-ELS	**3.4173**	**78**	271	685	17.24
		IPG-FixStep	3.4232	1314	1315	–	59.93
		NIPG1	3.4221	81	81	–	**3.10**
		NIPG2	3.4223	340	340	–	13.07
Heart Disease ($\rho = 2$)	15.2116	IPG-ELS	0.1476	101	229	254	1.26
		PG-ELS	0.1476	**98**	6475	229	19.58
		IPG-FixStep	0.1476	559	560	–	2.51
		NIPG1	0.1476	127	177	–	**0.90**
		NIPG2	0.1476	132	202	–	1.01
Heart Disease ($\rho = 4$)	243.3851	IPG-ELS	0.2377	101	103	707	0.73
		PG-ELS	0.2377	91	1370	584	4.81
		IPG-FixStep	0.2377	478	479	–	2.33
		NIPG1	**0.2367**	**64**	64	–	**0.37**
		NIPG2	0.2376	138	138	–	0.81
Heart Disease ($\rho = 8$)	3894.1611	IPG-ELS	0.6840	101	101	1074	0.73
		PG-ELS	0.6839	87	294	888	1.24
		IPG-FixStep	0.6839	604	605	–	2.73
		NIPG1	0.6837	**54**	54	–	**0.29**
		NIPG2	**0.6759**	113	113	–	0.59
CNS ($\rho = 2$)	8.2861	IPG-ELS	0.5912	101	192	169	131.22
		PG-ELS	0.5912	105	635	168	180.64
		IPG-FixStep	0.5912	372	373	–	180.44
		NIPG1	0.5912	**62**	78	–	18.98
		NIPG2	0.5912	57	80	–	**17.96**
CNS ($\rho = 4$)	132.5772	IPG-ELS	1.4485	101	101	573	355.83
		PG-ELS	1.4469	107	659	599	430.16
		IPG-FixStep	1.4483	428	429	–	350.29
		NIPG1	**1.4359**	**51**	51	–	**25.74**
		NIPG2	1.4484	94	94	–	43.71
CNS ($\rho = 8$)	2121.2350	IPG-ELS	3.9263	101	101	984	282.32
		PG-ELS	3.9232	78	338	704	231.55
		IPG-FixStep	3.9257	924	925	–	423.56
		NIPG1	**3.9174**	**80**	80	–	**23.00**
		NIPG2	3.9244	163	163	–	46.68

Table 2: Numerical comparison on benchmark datasets (Lung cancer-Michigan (96×7129), Secom (1567×590), and Cina0 (132×16033)) for the CUR-like factorization problem (74) with $\mu = 0$

Problem	Lips	Method	$F(X^k)$	Out-IT	In-IT	ELS-IT	Time (s)
Lung Cancer ($\rho = 2$)	10.3855	IPG-ELS	0.5075	101	296	166	157.61
		PG-ELS	0.5075	81	499	108	153.56
		IPG-FixStep	0.5075	599	600	–	363.13
		NIPG1	0.5075	**74**	134	–	**35.86**
		NIPG2	0.5075	78	147	–	39.11
Lung Cancer ($\rho = 4$)	166.1685	IPG-ELS	1.3428	101	128	600	398.40
		PG-ELS	1.3428	**89**	710	490	398.27
		IPG-FixStep	1.3428	1423	1424	–	1262.98
		NIPG1	1.3424	159	159	–	**92.15**
		NIPG2	**1.3397**	178	179	–	99.83
Lung Cancer ($\rho = 8$)	2658.6957	IPG-ELS	3.6391	**101**	101	992	305.89
		PG-ELS	3.6384	165	758	1607	562.66
		IPG-FixStep	3.8819	2001	2001	–	1043.98
		NIPG1	**3.6390**	338	338	–	**114.46**
		NIPG2	**3.6390**	495	495	–	163.31
Secom ($\rho = 2$)	9.0433	IPG-ELS	0.5832	101	294	163	73.65
		PG-ELS	0.5831	97	9228	149	810.76
		IPG-FixStep	0.5832	412	413	–	103.30
		NIPG1	0.5830	67	80	–	**17.53**
		NIPG2	**0.5827**	**52**	104	–	20.32
Secom ($\rho = 4$)	144.6929	IPG-ELS	0.7357	101	103	583	106.81
		PG-ELS	**0.7351**	94	8613	524	874.48
		IPG-FixStep	0.7356	470	471	–	146.13
		NIPG1	0.7356	**67**	67	–	**20.70**
		NIPG2	0.7355	123	123	–	38.64
Secom ($\rho = 8$)	2315.0869	IPG-ELS	1.2157	101	101	967	83.97
		PG-ELS	1.2140	76	1852	699	187.12
		IPG-FixStep	1.2157	885	886	–	169.99
		NIPG1	**1.2120**	**73**	73	–	**13.84**
		NIPG2	1.2146	222	222	–	46.86
Cina0 ($\rho = 2$)	13.5091	IPG-ELS	0.7602	**101**	284	207	1187.46
		PG-ELS	0.7602	108	1398	210	2267.36
		IPG-FixStep	0.7602	767	768	–	2843.81
		NIPG1	0.7602	252	496	–	1313.71
		NIPG2	0.7602	108	228	–	**590.34**
Cina0 ($\rho = 4$)	216.1458	IPG-ELS	0.9854	101	216	686	2877.48
		PG-ELS	0.9854	104	1949	689	5548.03
		IPG-FixStep	0.9854	225	226	–	1440.66
		NIPG1	**0.9850**	**46**	51	–	**277.23**
		NIPG2	0.9853	53	60	–	319.32
Cina0 ($\rho = 8$)	3458.3324	IPG-ELS	2.6661	101	144	1020	1828.50
		PG-ELS	**2.6348**	**41**	758	391	1436.24
		IPG-FixStep	2.6665	2001	2001	–	7310.17
		NIPG1	2.6661	168	168	–	**561.32**
		NIPG2	2.6661	342	342	–	1174.05

Table 3: Numerical comparison on benchmark datasets (Colon tumor (62×2000), Heart Disease (303×14), Central Nervous System (CNS) (60×7129)) for the CUR-like factorization problem (74) with $\mu = 0.01$

Problem $\mu = 0.01$	**Lips**	**Method**	$F(X^k)$	**Out-IT**	**In-IT**	**ELS-IT**	**Time (s)**
Colon Tumor ($\rho = 2$)	8.2139	IPG-ELS	0.6536	101	288	164	13.62
		PG-ELS	**0.6535**	102	3584	162	48.51
		IPG-FixStep	**0.6535**	332	333	–	14.24
		NIPG1	**0.6535**	**48**	71	–	**2.16**
		NIPG2	**0.6535**	73	127	–	3.58
Colon Tumor ($\rho = 4$)	131.4225	IPG-ELS	1.6314	101	109	587	15.39
		PG-ELS	1.6305	109	580	609	22.58
		IPG-FixStep	1.6314	451	452	–	20.65
		NIPG1	**1.6277**	**56**	56	–	**2.08**
		NIPG2	1.6281	123	123	–	4.57
Colon Tumor ($\rho = 8$)	2102.761	IPG-ELS	3.5226	101	101	992	19.87
		PG-ELS	3.5208	79	272	696	18.36
		IPG-FixStep	3.5222	1136	1137	–	58.34
		NIPG1	**3.487**	**73**	73	–	**2.54**
		NIPG2	3.5223	292	292	–	9.81
Heart Disease ($\rho = 2$)	15.2116	IPG-ELS	0.149	101	281	228	1.47
		PG-ELS	0.149	**99**	4375	206	13.61
		IPG-FixStep	0.149	763	764	–	3.43
		NIPG1	0.149	134	203	–	**1**
		NIPG2	0.149	221	396	–	1.94
Heart Disease ($\rho = 4$)	243.3851	IPG-ELS	0.2385	101	102	705	0.71
		PG-ELS	0.2385	90	1369	575	4.34
		IPG-FixStep	0.2385	448	449	–	14.24
		NIPG1	**0.2373**	**58**	58	–	**0.3**
		NIPG2	0.2384	130	130	–	0.66
Heart Disease ($\rho = 8$)	3894.161	IPG-ELS	0.6991	101	101	1060	0.73
		PG-ELS	0.6989	91	312	936	1.46
		IPG-FixStep	0.6989	525	526	–	2.5
		NIPG1	**0.6839**	**54**	54	–	**0.27**
		NIPG2	0.6911	112	112	–	0.57
CNS ($\rho = 2$)	8.2861	IPG-ELS	0.6013	101	212	170	124.88
		PG-ELS	0.6013	103	623	167	158.08
		IPG-FixStep	0.6013	369	370	–	169.12
		NIPG1	**0.6011**	**58**	76	–	**17.61**
		NIPG2	0.6013	85	139	–	27.82
CNS ($\rho = 4$)	132.5772	IPG-ELS	1.4568	101	101	573	197.02
		PG-ELS	1.4563	124	760	693	295.96
		IPG-FixStep	1.4567	434	435	–	199.03
		NIPG1	**1.4509**	**51**	51	–	**14.55**
		NIPG2	1.4555	97	97	–	27.81
CNS ($\rho = 8$)	2121.235	IPG-ELS	3.9644	101	101	981	275.62
		PG-ELS	3.9591	**79**	355	709	231.77
		IPG-FixStep	3.9642	885	886	–	406.63
		NIPG1	**3.8649**	98	98	–	**27.32**
		NIPG2	3.9586	154	154	–	42.75

Table 4: Numerical comparison on benchmark datasets (Lung cancer-Michigan (96×7129), Secom (1567×590), and Cina0 (132×16033)) for the CUR-like factorization problem (74) with $\mu = 0.01$

Problem $\mu = 0.01$	**Lips**	**Method**	$F(X^k)$	**Out-IT**	**In-IT**	**ELS-IT**	**Time (s)**
Lung Cancer ($\rho = 2$)	10.3855	IPG-ELS	0.51	101	314	166	154.24
		PG-ELS	0.51	103	631	167	175.89
		IPG-FixStep	0.51	2001	2001	–	1191.42
		NIPG1	0.51	84	169	–	37.45
		NIPG2	0.51	**80**	153	–	**33.48**
Lung Cancer ($\rho = 4$)	166.1685	IPG-ELS	1.3597	101	132	603	224.74
		PG-ELS	**1.3589**	**92**	728	509	249.71
		IPG-FixStep	1.3597	1275	1276	–	659.56
		NIPG1	1.3597	141	141	–	**47.39**
		NIPG2	1.3593	194	195	–	65.35
Lung Cancer ($\rho = 8$)	2658.696	IPG-ELS	3.823	**101**	101	996	308.09
		PG-ELS	3.8159	105	472	1040	349.7
		IPG-FixStep	3.8914	2001	2001	–	1061.11
		NIPG1	**3.7989**	177	177	–	**57.66**
		NIPG2	3.8218	326	326	–	118
Secom ($\rho = 2$)	9.0433	IPG-ELS	0.5946	101	358	166	63.24
		PG-ELS	0.5946	109	10282	176	745.86
		IPG-FixStep	0.5946	411	412	–	77.91
		NIPG1	**0.5944**	**65**	110	–	**17.4**
		NIPG2	0.5946	101	204	–	31.21
Secom ($\rho = 4$)	144.6929	IPG-ELS	0.7444	101	102	573	57.72
		PG-ELS	0.7442	94	8424	525	625.09
		IPG-FixStep	0.7443	458	459	–	86.45
		NIPG1	0.744	**66**	66	–	**11.86**
		NIPG2	**0.7328**	123	123	–	21.16
Secom ($\rho = 8$)	2315.087	IPG-ELS	1.2243	101	101	974	78.26
		PG-ELS	1.2237	**72**	1883	661	174.98
		IPG-FixStep	1.2242	863	864	–	154.88
		NIPG1	1.**2217**	**72**	72	–	**15.18**
		NIPG2	1.222	219	219	–	45.35
Cina0 ($\rho = 2$)	13.5091	IPG-ELS	0.7626	101	290	213	1226.1
		PG-ELS	0.7626	96	1223	189	1961.86
		IPG-FixStep	0.7626	2001	2001	–	7251.74
		NIPG1	0.7626	**68**	110	–	**320.29**
		NIPG2	0.7626	92	187	–	493.16
Cina0 ($\rho = 4$)	216.1458	IPG-ELS	0.9857	101	215	690	1598.93
		PG-ELS	0.9857	109	1960	723	3429.5
		IPG-FixStep	0.9857	228	229	–	934.59
		NIPG1	0.9857	**46**	52	–	**161.88**
		NIPG2	0.9857	55	62	–	192.58
Cina0 ($\rho = 8$)	3458.3324	IPG-ELS	2.6356	101	177	980	**1776.78**
		PG-ELS	**2.6355**	**48**	970	455	1868.56
		IPG-FixStep	2.6670	2001	2001	–	7537.68
		NIPG1	2.6356	658	658	–	2241.38
		NIPG2	2.6356	627	627	–	2143.95

Table 5: Numerical comparison on synthetic datasets for indefinite quadratic programming problem (IQPP)

Dims	Lips	Method	$F(X^k)$	Out-IT	In-IT	ELS-IT	Time (s)
3000x150	29.9987	IPG-ELS	-5.6146×10^7	101	109	359	44.33
		PG-ELS	-5.6153×10^7	102	9518	361	325.62
		IPG-FixStep	-5.6153×10^7	212	213	–	48.71
		NIPG1	$\mathbf{-5.6201 \times 10^7}$	**63**	63	–	**10.35**
		NIPG2	-5.6155×10^7	101	101	–	18.23
3000x150	79.9307	IPG-ELS	-4.8087×10^7	101	101	494	54.55
		PG-ELS	$\mathbf{-4.8145 \times 10^7}$	108	9356	508	222.56
		IPG-FixStep	-4.8088×10^7	397	398	–	56.72
		NIPG1	-4.8118×10^7	**95**	95	–	**11.26**
		NIPG2	-4.8131×10^7	98	98	–	11.89
500x500	29.9836	IPG-ELS	-3.0158×10^7	101	101	363	9.41
		PG-ELS	-3.0161×10^7	112	10629	402	147.14
		IPG-FixStep	-3.0162×10^7	209	210	–	10.85
		NIPG1	$\mathbf{-3.0196 \times 10^7}$	**62**	62	–	**3.73**
		NIPG2	-3.0159×10^7	103	103	–	6.55
500x500	79.9710	IPG-ELS	-2.6007×10^7	101	101	489	10.81
		PG-ELS	-2.6016×10^7	102	8169	483	135.50
		IPG-FixStep	-2.6007×10^7	239	240	–	13.14
		NIPG1	-2.6009×10^7	68	68	–	3.48
		NIPG2	$\mathbf{-2.6088 \times 10^7}$	**65**	65	–	**3.24**
250x2500	29.9886	IPG-ELS	-7.3545×10^7	101	101	367	53.76
		PG-ELS	-7.3547×10^7	116	10936	424	621.59
		IPG-FixStep	-7.3549×10^7	228	229	–	59.77
		NIPG1	-7.3553×10^7	**70**	70	–	**17.12**
		NIPG2	$\mathbf{-7.3557 \times 10^7}$	118	118	–	28.73
250x2500	79.4391	IPG-ELS	-6.4067×10^7	101	101	490	57.63
		PG-ELS	-6.4344×10^7	82	6721	390	379.09
		IPG-FixStep	-6.4090×10^7	238	239	–	57.26
		NIPG1	$\mathbf{-6.4348 \times 10^7}$	**63**	63	–	**15.31**
		NIPG2	-6.4225×10^7	64	64	–	15.64

7 Conclusions

In this paper, we have proposed New adaptive Inexact Proximal Gradient algorithms, termed NIPGℓ ($\ell = 1, 2$), designed for solving nonsmooth composite optimization problems in both convex and nonconvex settings. By integrating a local curvature-based adaptive stepsize rules with a robust relative error tolerance criterion for solving proximal subproblems, the proposed framework successfully eliminates the need for expensive backtracking line search and prior estimation of global Lipschitz constants. On the theoretical front, under nonconvex (Quacavex) assumption, we proved that the sequence of errors converges to zero and every cluster point of the generated sequence is a stationary point of the problem. For the convex setting, full convergence of the sequence to an optimal solution with a sublinear rate of $O(1/K)$ was successfully guaranteed, from a finite iteration. Numerical evaluations on high-dimensional CUR-like matrix factorization and indefinite quadratic programming problems with elastic net and group lasso regularization have confirmed the remarkable computational efficiency of our new method. As a direction for future research, we can extend our adaptive inexact proximal gradient (NIPG) framework from Euclidean spaces to manifolds inspired by the proximal gradient method developed by Bento and Santiago [12] recently.

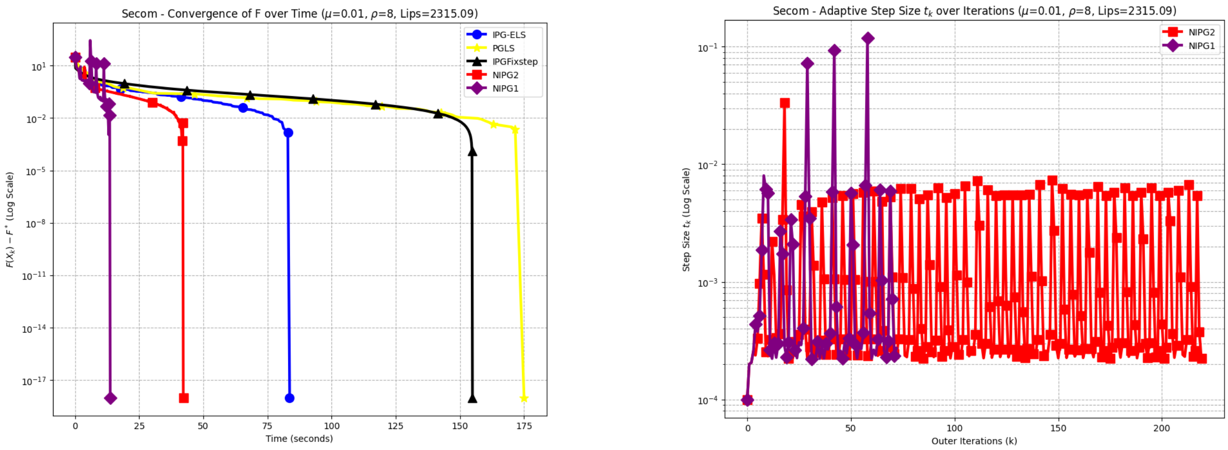

Fig. 1: Comparison of performance among different algorithms on the Secom with $\rho = 8$ dataset for the CUR-like factorization problem (74) with $\mu = 0.01$

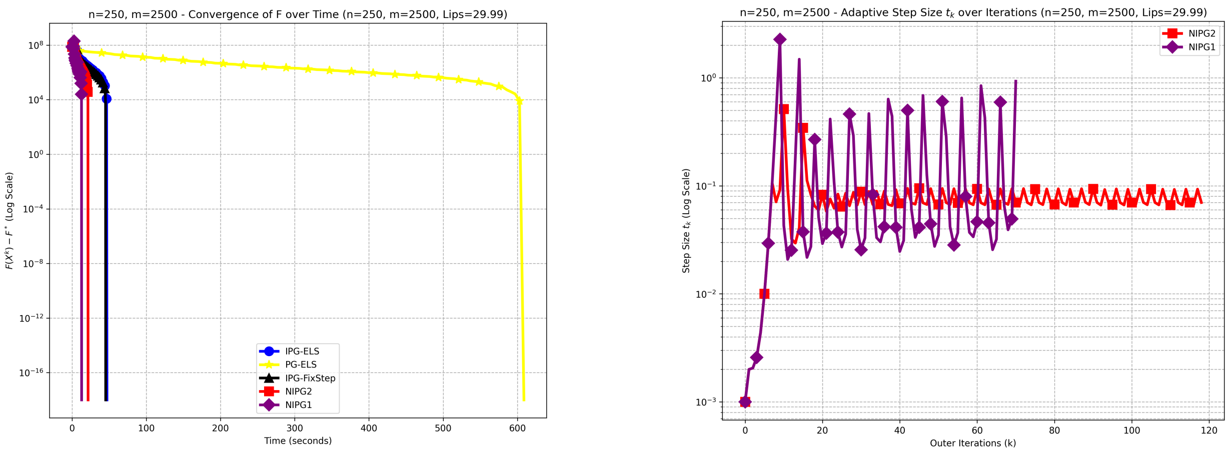

Fig. 2: Comparison of performance among different algorithms on the dims 250×2500 dataset for the indefinite quadratic programming problem (IQPP) with $Lf = 29.9986$

Acknowledgements

The author (Pham Thi Hoai) would like to express her deep gratitude to Professor Pham Duy Khanh for motivating her to pursue research on inexact algorithms at the annual workshop on optimization and scientific computing, Ba Vi, Hanoi, April 2025.

References

1. V.A. Adona, M.LN. Gonçalves, and J.G. Melo. A partially inexact proximal alternating direction method of multipliers and its iteration-complexity analysis. *Journal of Optimization Theory and Applications*, 182(2):640–666, 2019.

2. V.A. Adona, M.LN. Gonçalves, and J.G. Melo. An inexact proximal generalized alternating direction method of multipliers. *Computational Optimization and Applications*, 76(3):621–647, 2020.

3. Y.I. Alber, A.N. Iusem, and M. V Solodov. On the projected subgradient method for nonsmooth convex optimization in a Hilbert space. *Mathematical Programming*, 81(1):23–35, 1998.

4. M.M. Alves, J. Eckstein, M. Geremia, and J.G. Melo. Relative-error inertial-relaxed inexact versions of Douglas-Rachford and ADMM splitting algorithms: Mm alves et al. *Computational Optimization and Applications*, 75(2):389–422, 2020.

5. J-F. Aujol and C. Dossal. Stability of over-relaxations for the forward-backward algorithm, application to FISTA. *SIAM Journal on Optimization*, 25(4):2408–2433, 2015.
6. M. Barré, A. Taylor, and F. Bach. A note on approximate accelerated forward-backward methods with absolute and relative errors, and possibly strongly convex objectives. *Open Journal of Mathematical Optimization*, 3:1–15, 2022.
7. H.H. Bauschke and J.M. Borwein. On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3):367–426, 1996.
8. H.H. Bauschke and P.L. Combettes. A Dykstra-like algorithm for two monotone operators. *Pacific Journal of Optimization*, 4(3):383–391, 2008.
9. A. Beck. *First Order Methods in Optimization*. Society for Industrial and Applied Mathematics, USA, 2017.
10. J.Y. Bello-Cruz, M.LN. Gonçalves, J.G. Melo, and C. Mohr. A relative inexact proximal gradient method with an explicit linesearch. *Journal of Optimization Theory and Applications*, 206(1):9, 2025.
11. J.Y. Bello-Cruz and T.T.A. Nghia. On the convergence of the forward–backward splitting method with linesearches. *Optim. Methods Softw.*, 31(6):1209–1238, 2016.
12. G.C. Bento and C.R. Santiago. On the convergence and properties of a proximal-gradient method on Hadamard manifolds. *Journal of Nonlinear and Variational Analysis*, 10(3):617–633, 2026.
13. S. Bonettini, I. Loris, F. Porta, and M. Prato. Variable metric inexact line-search-based methods for nonsmooth optimization. *SIAM Journal on Optimization*, 26(2):891–921, 2016.
14. S. Bonettini and V. Ruggiero. On the convergence of primal–dual hybrid gradient algorithms for total variation image restoration. *Journal of Mathematical Imaging and Vision*, 44(3):236–253, 2012.
15. K. Bredies. A forward–backward splitting algorithm for the minimization of non-smooth convex functionals in Banach space. *Inverse Problems*, 25(1):015005, 2009.
16. R.E. 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.*, 61:159–164, 1977.
17. P.L. Combettes and V.R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4):1168–1200, 2005.
18. J. Eckstein and P.J.S. Silva. A practical relative error criterion for augmented Lagrangians. *Mathematical Programming*, 141(1):319–348, 2013.
19. J. Eckstein and W. Yao. Relative-error approximate versions of Douglas-Rachford splitting and special cases of the admm. *Mathematical Programming*, 170(2):417–444, 2018.
20. M. Fukushima and H. Mine. A generalized proximal point algorithm for certain non-convex minimization problems. *International Journal of Systems Science*, 12(8):989–1000, 1981.
21. B. Gu, D. Wang, Z. Huo, and H. Huang. Inexact proximal gradient methods for non-convex and non-smooth optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.
22. U. Hiriart, B. Jean, and C. Lemaréchal. Grundlehren der mathematischen wissenschaften. *Convex Analysis and Minimization Algorithms II*, 306, 1993.
23. P.T. Hoai and N.P.D. Thai. Composite optimization models via proximal gradient method with a novel enhanced adaptive stepsize. *Optimization Online, preprint, https://optimization-online.org/?p=26741*, 2024.
24. P.T. Hoai, N.T. Vinh, and N.P.H. Chung. A novel stepsize for gradient descent method. *Oper. Res. Lett.*, page 107072, 2024.
25. K. Jiang, D. Sun, and K-C. Toh. An inexact accelerated proximal gradient method for large scale linearly constrained convex sdp. *SIAM Journal on Optimization*, 22(3):1042–1064, 2012.
26. P.D. Khanh, B.S. Mordukhovich, V.T. Phat, and D.B. Tran. Inexact proximal methods for weakly convex functions. *Journal of Global Optimization*, 91(3):611–646, 2025.
27. J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Convex and network flow optimization for structured sparsity. *Journal of Machine Learning Research*, 12(9), 2011.

28. R.D. Millán and M.P. Machado. Inexact proximal ϵ-subgradient methods for composite convex optimization problems. *Journal of Global Optimization*, 75(4):1029–1060, 2019.

29. R.D.C. Monteiro and B. Svaiter. On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. *SIAM Journal on Optimization*, 20(6):2755–2787, 2010.

30. N. Parikh and S. Boyd. Proximal algorithms. *Found. Trends Optim.*, 1(3):127–239, 2014.

31. G.B. Passty. Ergodic convergence to a zero of the sum of monotone operators in Hilbert space. *J. Math. Anal. Appl.*, 72:383–390, 1979.

32. M. Schmidt, N. Roux, and F. Bach. Convergence rates of inexact proximal-gradient methods for convex optimization. *Advances in Neural Information Processing Systems*, 24, 2011.

33. T. Schuster, B. Kaltenbacher, B. Hofmann, and K.S. Kazimierski. *Regularization methods in Banach spaces*, volume 10. Walter de Gruyter, 2012.

34. M.V. Solodov and B.F. Svaiter. A hybrid approximate extragradient–proximal point algorithm using the enlargement of a maximal monotone operator. *Set-Valued Analysis*, 7(4):323–345, 1999.

35. S. Villa, S. Salzo, L. Baldassarre, and A. Verri. Accelerated and inexact forward-backward algorithms. *SIAM Journal on Optimization*, 23(3):1607–1633, 2013.

36. J. Yu, Z. Wang, Q. Wang, and H. He. Inertial proximal extragradient algorithms for nonsmooth composite optimization. *Optimization Eruditorum*, 2(2):97–115, 2025.

37. L.W. Zhong and J.T. Kwok. Efficient sparse modeling with automatic feature grouping. *IEEE Transactions on Neural Networks and Learning Systems*, 23(9):1436–1447, 2012.