École polytechnique fédérale de Lausanne

Master thesis spring 2023

Master in Applied Mathematics

Analysis of PDLP A First Order Method For Solving Linear Programming Problems

Author:
Corentin Tissier

Supervisor:
Nicolas Boumal

Contents

Abstract

In this work we study PDLP, an algorithm introduced in [1] designed to solve linear programming (LP) problems. As PDLP currently lacks formal theoretical guarantees, we review the convergence theory of its baseline algorithm, PDHG. We motivate the various enhancements that PDLP provides over PDHG through both theory and numerical experiments, validating the advantages and benefits offered by the enhancements present in PDLP.

1 Introduction

In recent years, first order methods have become popular and effective optimization techniques that can be applied to a wide range of problems in different fields. In particular, PDHG (Primal Dual Hybrid Gradient) also known as Chambolle Pock method gained significant attention and was studied in [4, 5, 6].

The fundamental idea behind PDHG lies in the reformulation of an optimization problem as a saddle point problem involving both primal and dual variables. By exploiting the duality gap and utilizing a subgradient-based iterative scheme, PDHG algorithms are able to minimize the primal and maximize the dual simultaneously, thus reaching optimal solutions efficiently in various applications.

One such area is Linear Programming (LP), and the main class of problem that will be studied here. LP is a fundamental class of optimization problems in applied mathematics characterised by linear objective functions and constraints. In 2021, an enhanced version of PDHG specified in the resolution of LP problems named PDLP (Primal Dual LP) was released [1] claiming to obtain results comparable to standard commercial LP solver.

This paper aims to study the theory underlying PDLP and in particular PDHG theoretical foundations by providing a comprehensive analysis of its convergence properties, but also to test the performance of PDLP to understand the benefits it brings over the simpler PDHG.

We begin by showing how to derive a saddle point formulation from a LP problem. We then study the convergence property of the basic PDHG and adaptive stepsize versions of it. Subsequently, we motivate the need for a restart scheme and study the underlying theory. This leads to the investigation of PDLP, an enhanced version of PDHG that combines adaptive stepsizes and restart strategies by conducting a numerical study to evaluate its performance.

2 Preliminaries

In this section, we introduce the notation we use throughout the paper and summarize the LP formulations.

2.1 Notations and preliminary definitions

Let $\mathbb{R}^n_+$ denote the set of real valued vector in $\mathbb{R}^n$ with non negative entries, let $\|.\|_2$ denote the spectral norm for a matrix. For a vector $v \in \mathbb{R}^n$ we use v^+ and v^- for their positive and negative parts, i.e., $v_i^+ = \max\{0, v_i\}$ and $v_i^- = -\min\{0, v_i\}$. The symbol $v_{1:m}$ denotes the vector with the first m components of v. For two sets X and Y, we will often denote $Z = X \times Y$ and $z = (x, y)$ for any $x \in X$ and $y \in Y$. If not specified otherwise $\|x\| = \|x\|_2$ is the Euclidean norm for a vector x. Similarly, in general for a matrix A, $\|A\|$ will denote the operator norm of A.

Given a convex set $X \subset \mathbb{R}^n$, we define $\mathbf{proj}_X$ the projection on X by

$$\mathbf{proj}_X(z) \coloneqq \underset{x \in X}{\operatorname{argmin}} \frac{1}{2}\|x - z\|_2^2.$$

For a lower semi-continuous convex function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ we define the subgradient of f at $x \in \mathbf{dom} f \coloneqq \{x \in \mathbb{R}^n : f(x) < +\infty\}$ by

$$\partial f(x) \coloneqq \{g : (y - x)^\intercal g + f(x) \leq f(y) \quad \forall y \in \mathbf{dom} f\}.$$

The proximal/proximity operator (see [3] for more information on this object) of f is then defined by

$$\mathrm{Prox}_f(x) \coloneqq \arg\min_{x \in \mathcal{X}} \left(f(x) + \frac{1}{2}\|x - v\|_2^2 \right).$$

Finally, we will use $\mathcal{X}_C$ to denote the characteristic function of a convex set C, which is defined as follows:

$$\mathcal{X}_C = \begin{cases} 0 & \text{if} \quad x \in C \\ \infty & \text{otherwise.} \end{cases}$$

2.2 Primal Problem

Our goal is to solve LP problems of the form:

$$\begin{aligned} \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad & c^\intercal x \\ \text{subject to} \quad & Gx \geq h \\ & Ax = b \\ & l \leq x \leq u \end{aligned} \tag{1}$$

where $G \in \mathbb{R}^{m_1 \times n}$, $A \in \mathbb{R}^{m_2 \times n}$, $c \in \mathbb{R}^n$, $h \in \mathbb{R}^{m_1}$, $b \in \mathbb{R}^{m_2}$, $l \in (\mathbb{R} \cup \{-\infty\})^n$ and $u \in (\mathbb{R} \cup \{+\infty\})^n$.

2.3 Dual Problem

As we will soon prove, the dual formulation of the primal problem (1) is:

$$\begin{aligned} \underset{y\in\mathbb{R}^{m_1+m_2},\lambda\in\mathbb{R}^n}{\text{maximize}} \quad & q^\intercal y + l^\intercal \lambda^+ - u^\intercal \lambda^- \\ \text{subject to} \quad & c - K^\intercal y = \lambda \\ & y_{1:m1} \geq 0 \\ & \lambda \in \Lambda \end{aligned} \tag{2}$$

where we have defined $K^\intercal := (G^\intercal, A^\intercal)$, $q^\intercal := (h^\intercal, b^\intercal)$ and

$$\Lambda = \Lambda_1 \times ... \times \Lambda_n \quad \Lambda_i := \begin{cases} 0 & \text{if} \quad l_i = -\infty, u_i = \infty \\ \mathbb{R}^- & \text{if} \quad l_i = -\infty, u_i \in \mathbb{R} \\ \mathbb{R}^+ & \text{if} \quad l_i \in \mathbb{R}, u_i = +\infty \\ \mathbb{R} & \text{otherwise.} \end{cases} \tag{3}$$

This dual interpretation of the LP problem will be used later to build a termination criteria for the algorithm.

Lemma 2.1 (Dual formulation of LP). *The dual formulation of (1) is (2).*

Proof. Let's consider the Lagrangian for the optimisation problem (1), it is given by the function

$$\begin{aligned} L &: \mathbb{R}^n \times \mathbb{R}^{m_2} \times \mathbb{R}_+^{m_1} \times \mathbb{R}_+^n \times \mathbb{R}_+^n \\ &: (x, \mu, \lambda_1, \lambda_2, \lambda_3) \mapsto c^\intercal x + \mu^\intercal (b - Ax) + \lambda_1^\intercal (h - Gx) + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u) \end{aligned}$$

The dual problem is by definition:

$$\underset{(\mu,\lambda_1,\lambda_2,\lambda_3)\in\mathbb{R}^{m_2}\times\mathbb{R}_+^{m_1}\times\mathbb{R}_+^n\times\mathbb{R}_+^n}{\text{maximize}} \quad L_D(\mu, \lambda_1, \lambda_2, \lambda_3)$$

where the dual function L_D is :

$$\begin{aligned} L_D(\mu, \lambda_1, \lambda_2, \lambda_3) &= \inf_{x\in\mathbb{R}^n} L(x, \mu, \lambda_1, \lambda_2, \lambda_3) \\ &= \inf_{x\in\mathbb{R}^n} c^\intercal x + \mu^\intercal (b - Ax) + \lambda_1^\intercal (h - Gx) + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u) \\ &= \inf_{x\in\mathbb{R}^n} (c - A^\intercal \mu - G^\intercal \lambda_1 - \lambda_2 + \lambda_3)^\intercal x + b^\intercal \mu + h^\intercal \lambda_1 + l^\intercal \lambda_2 - u^\intercal \lambda_3 \\ &= \inf_{x\in\mathbb{R}^n} \begin{cases} b^\intercal \mu + h^\intercal \lambda_1 + l^\intercal \lambda_2 - u^\intercal \lambda_3 & \text{if} \quad c - A^\intercal \mu - G^\intercal \lambda_1 - \lambda_2 + \lambda_3 = 0 \\ -\infty & \text{otherwise} \end{cases} \\ &= \inf_{x\in\mathbb{R}^n} \begin{cases} q^\intercal y + l^\intercal \lambda_2 - u^\intercal \lambda_3 & \text{if} \quad c - K^\intercal y = \lambda_2 - \lambda_3 \\ -\infty & \text{otherwise} \end{cases} \end{aligned}$$

with $y^\intercal = (\lambda_1^\intercal, \mu^\intercal) \in \mathbb{R}_+^{m_1} \times \mathbb{R}^{m_2}$.

Thus by noticing that when $l_i = -\infty$, x_i is not constrained by a lower bound and $\lambda_2 = 0$ (similarly $u_i = +\infty \Rightarrow \lambda_3 = 0$), we have that $\lambda_2 - \lambda_3 \in \Lambda$ and the dual problem becomes:

$$\begin{aligned} \underset{y\in\mathbb{R}^{m_1+m_2},\lambda\in\mathbb{R}^n}{\text{maximize}} \quad & q^\intercal y + l^\intercal \lambda_2 - u^\intercal \lambda_3 \\ \text{subject to} \quad & c - K^\intercal y = \lambda \\ & \lambda_2 - \lambda_3 = \lambda \\ & y_{1:m1}, \lambda_2, \lambda_3 \geq 0 \\ & \lambda \in \Lambda \end{aligned}$$

We are now very close to (2) but it remains to show that the best possible candidate λ_2, λ_3 are in fact λ^+, λ^-.
Since $\lambda_2 - \lambda_3 = \lambda = \lambda^+ - \lambda^-$, we can define $a \coloneqq \lambda_2 - \lambda^+ = \lambda_3 - \lambda^-$. Moreover, by definition, λ^+ is the smallest positive (entry wise) vector such that there exists an other positive (entry wise) vector λ_3 with $\lambda^+ - \lambda_3 = \lambda$. Thus, $a \coloneqq \lambda_2 - \lambda^+ \geq 0$. But the objective function is

$$\begin{aligned} q^\intercal y + l^\intercal \lambda_2 - u^\intercal \lambda_3 &= q^\intercal y + l^\intercal \left(\lambda^+ + a\right) - u^\intercal \left(\lambda^- + a\right) \\ &= q^\intercal y + l^\intercal \lambda^+ - u^\intercal \lambda^- + (l-u)^\intercal a \end{aligned}$$

and as $(l-u) \leq 0$ and $a \geq 0$, to maximize the objective we need to take $a = 0$, i.e. $\lambda_2 = \lambda^+$ and $\lambda_3 = \lambda^-$. □

2.4 Saddle point problem

Now, to solve the LP problem using PDHG (the baseline of PDLP), we need to express it as a saddle point problem.

Lemma 2.2 (Saddle point formulation of LP)**.** *The pair of primal (1), dual (2) problems is equivalent to:*

$$\min_{x \in X} \max_{y \in Y} \mathcal{L}(x,y) = \min_{x \in X} \max_{y \in Y} qc^\intercal x - y^\intercal K x + q^\intercal y \tag{4}$$

with $X \coloneqq \{x \in \mathbb{R}^n : l \leq x \leq u\}$ *and* $Y \coloneqq \{y \in \mathbb{R}^{m_1+m_2} : y_{1:m_1} \geq 0\}$.

Proof. Once again, let's consider the Lagrangian for this LP problem:

$$\begin{aligned} &L : \mathbb{R}^n \times \mathbb{R}^{m_2} \times \mathbb{R}_+^{m_1} \times \mathbb{R}_+^n \times \mathbb{R}_+^n \\ &\quad : (x, \mu, \lambda_1, \lambda_2, \lambda_3) \mapsto c^\intercal x + \mu^\intercal (b - Ax) + \lambda_1^\intercal (h - Gx) + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u)\,. \end{aligned}$$

We first rewrite the Lagrangian:

$$\begin{aligned} L(x, \mu, \lambda_1, \lambda_2, \lambda_3) &= c^\intercal x + \mu^\intercal (b - Ax) + \lambda_1^\intercal (h - Gx) + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u) \\ &= c^\intercal x - (\mu^\intercal A x + \lambda_1^\intercal G x) + \lambda_2^\intercal (l - x) + (\mu^\intercal b + \lambda_1^\intercal) + \lambda_3^\intercal (x - u) \\ &= c^\intercal x - y^\intercal K x + q^\intercal y + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u) \end{aligned}$$

with $y^\intercal = (\lambda_1^\intercal, \mu^\intercal) \in \mathbb{R}_+^{m_1} \times \mathbb{R}^{m_2}$. Let L_P be the primal function:

$$\begin{aligned} L_P(x) &= \sup_{(y,\lambda_2,\lambda_3) \in Y \times \mathbb{R}_+^n \times \mathbb{R}_+^n} L(x, y, \lambda_2, \lambda_3) \\ &= \sup_{(y,\lambda_2,\lambda_3) \in Y \times \mathbb{R}_+^n \times \mathbb{R}_+^n} c^\intercal x - y^\intercal K x + q^\intercal y + \lambda_2^\intercal (l - x) + \lambda_3^\intercal (x - u) \\ &= \begin{cases} \sup\limits_{y \in Y} \quad c^\intercal x - y^\intercal K x + q^\intercal y & \text{if} \quad l \leq x \leq u \\ +\infty & \text{otherwise} \end{cases} \end{aligned}$$

By standard duality theory, we know that the primal problem (1) is equivalent to:

$$\min_{x \in \mathbb{R}^n} L_P(x) = \min_{x \in X} \max_{y \in Y} c^\intercal x - y^\intercal K x + q^\intercal y$$

which is the desired result □

Remark 1. *One should note that $\mathcal{L}(x,y) \coloneqq c^\intercal x - y^\intercal Kx + q^\intercal y$ is not the Lagrangian of the LP problem (1) but rather the Lagrangian of the problem without the lower and upper bounds on x. We use this saddle point formulation (4) by convenience, as PDHG solves problem of the form (see for instance [4] or [6])*

$$\min_{x\in X}\max_{y\in Y} \mathcal{L}(x,y) = \min_{x\in X}\max_{y\in Y} f(x) + y^\intercal Ax - g(y) \tag{5}$$

where f and g are convex functions, $A \in \mathbb{R}^{M\times N}$ is a matrix, $X \subset \mathbb{R}^N$ and $Y \subset \mathbb{R}^M$ are convex sets.

We finish this preliminary section with a useful characterisation of a saddle point.

Lemma 2.3 (Strong duality of LP)**.** *Consider the saddle point formulation of LP (4). For any $z = (x,y) \in X \times Y$ we have*

$$\max_{\hat{z}\in Z} \{\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y)\} \geq 0.$$

Furthermore, $z^* = (x^*,y^*) \in X \times Y$ is a solution to (4) if and only if

$$\max_{\hat{z}\in Z} \{\mathcal{L}(x^*,\hat{y}) - \mathcal{L}(\hat{x},y^*)\} = 0.$$

Proof. Let $(x,y) \in Z$. We have then:

$$\begin{aligned} &\max_{\hat{y}\in Y} \mathcal{L}(x,\hat{y}) \geq \mathcal{L}(x,y) \geq \min_{\hat{x}\in X} \mathcal{L}(\hat{x},y) \\ \Rightarrow &\max_{\hat{y}\in Y} \mathcal{L}(x,\hat{y}) - \min_{\hat{x}\in X} \mathcal{L}(\hat{x},y) \geq 0 \\ \Rightarrow &\max_{\hat{z}\in Z} \{\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y)\} \geq 0, \end{aligned}$$

which proves the first assertion.

To prove the second assertion, we first note that:

$$\begin{aligned} \mathcal{L}(x,y) &= c^\intercal x - y^\intercal Kx + q^\intercal y \\ &= c^\intercal x - y^\intercal (Kx - q) \\ &= c^\intercal x - y^\intercal \begin{pmatrix} Gx - h \\ Ax - b \end{pmatrix}. \end{aligned}$$

But then,

$$\max_{\hat{y}\in Y} \mathcal{L}(x,\hat{y}) = \begin{cases} c^\intercal x & \text{if } Gx - h \geq 0, Ax - b = 0 \\ +\infty & \text{else.} \end{cases}$$

Thus we have:

$$\begin{aligned} &\max_{\hat{y}\in Y} \mathcal{L}(x,\hat{y}) \geq \mathcal{L}(x,y) \geq \min_{\hat{x}\in X} \mathcal{L}(\hat{x},y) \\ \Leftrightarrow &\begin{cases} c^\intercal x & \text{if } Gx \geq h, Ax = b \\ +\infty & \text{else.} \end{cases} \geq \mathcal{L}(x,y) \geq \min_{\hat{x}\in X} \mathcal{L}(\hat{x},y). \end{aligned} \tag{6}$$

This means that given any $y \in Y$, for any $x \in X$ such that $Gx \geq h, Ax = b$ we have $c^\intercal x \geq \min_{\hat{x}} \mathcal{L}(\hat{x}, y)$. In particular, if we find (x^*, y^*) such that this last inequality is an equality, we then know that x^* minimizes $c^\intercal x$ and thus minimizes $\max_{\hat{y} \in Y} \mathcal{L}(x, \hat{y})$. In other terms, such x^*, y^* satisfy:

$$\min_{x \in X} \max_{y \in Y} \mathcal{L}(x, y) = \mathcal{L}(x^*, y^*).$$

But $(x^*, y^*) \in Z$ is such that

$$\max_{\hat{z} \in Z} \{\mathcal{L}(x^*, \hat{y}) - \mathcal{L}(\hat{x}, y^*)\} = 0$$

if and only if $Gx^* \geq h, Ax^* = b$ and

$$c^\intercal x^* = \mathcal{L}(x^*, y^*) = \min_{\hat{x} \in X} \mathcal{L}(\hat{x}, y^*),$$

and by our previous argument, it follows that this is equivalent to (x^*, y^*) being a saddle point of (4). □

3 PDHG

The PDHG algorithm was popularised by Chambolle and Pock in 2011 [4]. It is a first order method that solves problem of the form:

$$\min_{x \in X} \max_{y \in Y} \mathcal{L}(x, y) = \min_{x \in X} \max_{y \in Y} f(x) + y^\intercal A x - g(y) \tag{7}$$

where $f : \mathbb{R}^N \to \mathbb{R} \cup \{\pm\infty\}$ and $g : \mathbb{R}^M \to \mathbb{R} \cup \{\pm\infty\}$ are convex functions, $A \in \mathbb{R}^{M \times N}$ is a matrix, $X \subset \mathbb{R}^N$ and $Y \subset \mathbb{R}^M$ are convex sets. By convenience, we define $F(x) := (f + \mathcal{X}_X)(x)$ and $G(y) := (g + \mathcal{X}_Y)(y)$.

The PDHG algorithm is then:

Algorithm 1 PDHG

```
Require: $x_0 \in \mathbb{R}^N, y_0 \in \mathbb{R}^M, \sigma_k, \tau_k > 0$
while Not Converged do
    $\hat{x}_{k+1} = x_k - \tau_k A^\intercal y_k$;
    $x_{k+1} = \mathrm{Prox}_{\tau_k F}(\hat{x}_{k+1}) = \arg\min_{x \in X} f(x) + \frac{1}{2\tau_k}\|x - \hat{x}_{k+1}\|_2^2$;
    $\tilde{x}_{k+1} = x_{k+1} + (x_{k+1} - x_k)$;
    $\hat{y}_{k+1} = y_k + \sigma_k A \tilde{x}_{k+1}$;
    $y_{k+1} = \mathrm{Prox}_{\sigma_k G}(\hat{y}_{k+1}) = \arg\min_{y \in Y} g(y) + \frac{1}{2\sigma_k}\|y - \hat{y}_{k+1}\|_2^2$;
end while
```

We will show in the convergence analysis section 3.2 that PDHG converges for constant stepsize $\tau_k = \tau$ and $\sigma_k = \sigma$ as long as $\sigma\tau < \frac{1}{\|A\|^2}$.

3.1 Heuristic and first analysis

We begin this section by a useful characterization of the proximal operator.

Lemma 3.1. *Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a lower semi-continuous convex function. The proximal operator of f is characterized by*

$$p = \operatorname{Prox}_f(x) \Leftrightarrow x - p \in \partial f(p).$$

Proof. Let $y \in \mathbb{R}^n$. First, suppose that $p = \operatorname{Prox}_f x$ and set $(\forall \alpha \in]0,1[)$ $p_\alpha = \alpha y + (1-\alpha)p$. Then, for every $\alpha \in]\,0,1[$, using the convexity of f yields

$$\begin{aligned} f(p) + \frac{1}{2}\|x-p\|_2^2 \leq f(p_\alpha) + \frac{1}{2}\|x - p_\alpha\|_2^2 & \\ \Rightarrow f(p) \leqslant f(p_\alpha) + \frac{1}{2}\|x-p_\alpha\|^2 - \frac{1}{2}\|x-p\|^2 & \\ \Rightarrow f(p) \leqslant \alpha f(y) + (1-\alpha) f(p) - \alpha(x-p)^\intercal(y-p) + \frac{\alpha^2}{2}\|y-p\|^2 & \end{aligned}$$

and hence $(y-p)^\intercal(x-p) + f(p) \leqslant f(y) + \frac{\alpha}{2}\|y-p\|^2$. Letting $\alpha \downarrow 0$, we obtain the desired inequality.

Conversely, suppose that $x - p \in \partial f(x)$. Then for any $y \in \mathbf{dom}_f$

$$\begin{aligned} &(y-p)^\intercal(x-p) + f(p) \leq f(y) \\ &\quad \Rightarrow f(p) + \frac{1}{2}\|x-p\|^2 \leq f(y) + \frac{1}{2}\|x-p\|^2 + (x-p)^\intercal(p-y) \\ &\qquad\qquad\qquad\qquad\quad \leq f(y) + \frac{1}{2}\|x-p\|^2 + (x-p)^\intercal(p-y) + \frac{1}{2}\|p-y\|^2 \\ &\qquad\qquad\qquad\qquad\quad = f(y) + \frac{1}{2}\|x-y\|^2 \end{aligned}$$

and we conclude that $p = \operatorname{Prox}_f x$. □

The main idea behind PDHG is to transform a constrained optimization problem into an unconstrained one by introducing a dual variable. With the saddle point formulation (7) we still have some constraints materialised by the sets $X \subset \mathbb{R}^N$ and $Y \subset \mathbb{R}^M$. We thus rewrite it in an unconstrained form using the characteristics function:

$$\begin{aligned} &\min_{x\in\mathbb{R}^N} \max_{y\in\mathbb{R}^M} f(x) + \mathcal{X}_X(x) + y^\intercal A x - g(y) - \mathcal{X}_Y(y) \\ &= \min_{x\in\mathbb{R}^N} \max_{y\in\mathbb{R}^M} L(x,y) := F(x) + y^\intercal A x - G(y) \end{aligned} \tag{8}$$

Then, by definition of a subgradient, (x^*, y^*) is a solution to the saddle point problem (8) if and only if it satisfies:

$$\begin{aligned} &\begin{cases} 0 \in \partial F(x^*) + A^\intercal y^* \\ 0 \in \partial G(y^*) - Ax^* \end{cases} \\ \Leftrightarrow \quad &\begin{cases} -A^\intercal y^* \in \partial F(x^*) \\ Ax^* \in \partial G(y^*) \end{cases} \\ \Leftrightarrow \quad &\begin{cases} (x^* - A^\intercal y^*) - x^* \in \partial F(x^*) \\ (y^* + Ax^*) - y^* \in \partial G(y^*) \end{cases} \\ \Leftrightarrow \quad &\begin{cases} x^* = \mathrm{Prox_F}(x^* - A^\intercal y^*) \\ y^* = \mathrm{Prox_G}(y^* + Ax^*) \end{cases} \end{aligned} \tag{9}$$

where we used Lemma 3.1 in the last step. This motivates the PDHG updates of x_k and y_k described in Algorithm 1 as the convergence of the iterate (x_k, y_k) to a point (x^*, y^*) would mean that it is a fixed point of the algorithm i.e.

$$\begin{cases} x^* = \mathrm{Prox_F}(x^* - A^\intercal y^*) \\ y^* = \mathrm{Prox_G}(y^* + Ax^*). \end{cases}$$

As the PDHG updates cannot be done both independently and simultaneously, instead of computing y_{k+1} using x_{k+1}, we use $\tilde{x}_{k+1} = x_{k+1} + (x_{k+1} - x_k)$. This choice of $\tilde{x}_{k+1}$ will be important in the convergence analysis, in particular in the proof of Theorem 3.5.

Lemma 3.2 (PDHG for LP)**.** *For a saddle point problem of the form:*

$$\min_{x \in X} \max_{y \in Y} c^\intercal x + y^\intercal Ax - q^\intercal y$$

with $c \in \mathbb{R}^N, A \in \mathbb{R}^{M \times N}, q \in \mathbb{R}^M$ and $X \subset \mathbb{R}^N, Y \subset R^M$ two convex sets, the PDHG updates steps are:

$$\begin{aligned} x_{k+1} &= \mathbf{proj}_X \left(x_k - \tau_k (c + A^\intercal y_k) \right) \\ y_{k+1} &= \mathbf{proj}_Y \left(y_k + \sigma_k (-q + A(2x_{k+1} - x_k)) \right). \end{aligned} \tag{10}$$

Proof. Let's start with x_{k+1}. In Algorithm 1 the corresponding updates is with

$\hat{x}_{k+1} = x_k - \tau_k A^\intercal y_k$:

$$\begin{aligned}
x_{k+1} &= \mathrm{Prox}_{\tau_k F}(\hat{x}_{k+1}) \\
&= \arg\min_{x \in \mathbb{R}^N} \left\{ \tau_k \left(c^\intercal x + \mathcal{X}_X(x) \right) + \frac{1}{2} \|x - \hat{x}_{k+1}\|_2^2 \right\} \\
&= \arg\min_{x \in X} \left\{ c^\intercal x + \frac{1}{2\tau_k} \|x - \hat{x}_{k+1}\|_2^2 \right\} \\
&= \arg\min_{x \in X} \left\{ \frac{1}{2\tau_k} \left(2\tau_k c^\intercal x + x^\intercal x - 2x^\intercal \hat{x}_{k+1} + \hat{x}_{k+1}^\intercal \hat{x}_{k+1} \right) \right\} \\
&= \arg\min_{x \in X} \left\{ x^\intercal x - 2x^\intercal (\hat{x}_{k+1} - \tau_k c) + \hat{x}_{k+1}^\intercal \hat{x}_{k+1} \right\} \\
&= \arg\min_{x \in X} \left\{ \|x - (\hat{x}_{k+1} - \tau_k c)\|_2^2 + \|\hat{x}_{k+1}\|_2^2 - \|\hat{x}_{k+1} - \tau_k c\|_2^2 \right\} \\
&= \arg\min_{x \in X} \|x - (\hat{x}_{k+1} - \tau_k c)\|_2^2 \\
&= \mathbf{proj}_X(\hat{x}_{k+1} - \tau_k c) \\
&= \mathbf{proj}_X\left(x_k - \tau_k (c + A^\intercal y_k) \right)
\end{aligned}$$

Similarly for y_{k+1} with $\hat{y}_{k+1} = y_k + \sigma_k A(2x_{k+1} - x_k)$:

$$\begin{aligned}
y_{k+1} &= \mathrm{Prox}_{\sigma_k G}(\hat{y}_{k+1}) \\
&= \arg\min_{y \in \mathbb{R}^M} \left\{ \sigma_k \left(q^\intercal y + \mathcal{X}_Y(y) \right) + \frac{1}{2} \|y - \hat{y}_{k+1}\|_2^2 \right\} \\
&= \arg\min_{y \in Y} \left\{ q^\intercal y + \frac{1}{2\sigma_k} \|y - \hat{y}_{k+1}\|_2^2 \right\} \\
&= \arg\min_{y \in Y} \left\{ y^\intercal y - 2y^\intercal (\hat{y}_{k+1} - \sigma_k q) + \hat{y}_{k+1}^\intercal \hat{y}_{k+1} \right\} \\
&= \arg\min_{y \in Y} \|y - (\hat{y}_{k+1} - \sigma_k q)\|_2^2 \\
&= \mathbf{proj}_Y(\hat{y}_{k+1} - \sigma_k q) \\
&= \mathbf{proj}_Y\left(y_k + \sigma_k (-q + A(2x_{k+1} - x_k)) \right)
\end{aligned}$$

□

Remark 2. *In our LP case (7) with $A \leftarrow -K, q \leftarrow -q, X = \{x \in \mathbb{R}^n : l \le x \le u\}$ and $Y = \{y \in \mathbb{R}^{m_1+m_2} : y_{1:m_1} \ge 0\}$ the PDHG steps in (10) can be simply written as:*

$$\begin{aligned}
x_{k+1} &= \min\left\{ \max\left\{ x_k - \tau_k (c - K^\intercal y_k), l \right\}, u \right\} \\
y_{k+1} &= \max\left\{ y_k + \sigma_k (q - K(2x_{k+1} - x_k)), 0 \right\}
\end{aligned} \tag{11}$$

where the min and max are taken entry wise.

3.2 Convergence analysis

We first introduce a useful semi norm:

Definition 3.3. *Let $z = (x, y) \in \mathbb{R}^N \times \mathbb{R}^M, A \in \mathbb{R}^{M \times N}$ and $\tau, \sigma \in \mathbb{R}$. We define the matrix $W_{\tau,\sigma} := \begin{pmatrix} \frac{1}{\tau}\mathbf{I} & -A^\intercal \\ -A & \frac{1}{\sigma}\mathbf{I} \end{pmatrix}$ and the semi norm $\|z\|_{W_{\tau,\sigma}}^2 := z^\intercal W_{\tau,\sigma} z = \frac{\|x\|^2}{\tau} - 2y^\intercal A x + \frac{\|y\|^2}{\sigma}$.*

Remark 3. *To avoid heavy notations, we choose to write*

$$\|z\|_{W_{\tau,\sigma}}^2 = \frac{\|x\|^2}{\tau} - 2y^\intercal Ax + \frac{\|y\|^2}{\sigma}$$

even when the right hand side can be negative, i.e. even when $W_{\tau,\sigma}$ is not positive semi-definite.

Lemma 3.4. *If $\tau\sigma\|A\|^2 < 1$, then $W_{\tau,\sigma}$ is positive semi-definite and thus $\|z\|_{W_{\tau,\sigma}}$ is a norm. In particular we have for any $z = (x, y) \in \mathbb{R}^M \times \mathbb{R}^N$:*

$$\begin{aligned}(1 - \sqrt{\tau\sigma}\|A\|)\left(\frac{\|x\|^2}{\tau} + \frac{\|y\|^2}{\sigma}\right) &\leq \|z\|_{W_{\tau,\sigma}}^2 \\ &\leq (1 + \sqrt{\tau\sigma}\|A\|)\left(\frac{\|x\|^2}{\tau} + \frac{\|y\|^2}{\sigma}\right)\end{aligned} \tag{12}$$

Proof. Let $z = (x, y) \in \mathbb{R}^M \times \mathbb{R}^N$.

$$\begin{aligned}\|z\|_{W_{\tau,\sigma}}^2 &= \frac{\|x\|^2}{\tau} - 2y^\intercal Ax + \frac{\|y\|^2}{\sigma} \\ &\leq \frac{\|x\|^2}{\tau} + 2\|A\|\sqrt{\sigma/\tau}\|x\|^2\frac{\|y\|^2}{\sqrt{\sigma/\tau}} + \frac{\|y\|^2}{\sigma} \\ &\leq \frac{\|x\|^2}{\tau} + \frac{\|A\|\sqrt{\sigma\tau}}{2\tau}\|x\|^2 + \frac{\|A\|\sqrt{\sigma\tau}}{2\sigma}\|y\|^2 + \frac{\|y\|^2}{\sigma} \\ &= (1 + \sqrt{\tau\sigma}\|A\|^2)\left(\frac{\|x\|^2}{\tau} + \frac{\|y\|^2}{\sigma}\right)\end{aligned}$$

where we used the identity $2ab \leq a^+b^2$. Similarly, we have:

$$\begin{aligned}\|z\|_{W_{\tau,\sigma}}^2 &= \frac{\|x\|^2}{\tau} - 2y^\intercal Ax + \frac{\|y\|^2}{\sigma} \\ &\geq \frac{\|x\|^2}{\tau} - 2\|A\|\sqrt{\sigma/\tau}\|x\|^2\frac{\|y\|^2}{\sqrt{\sigma/\tau}} + \frac{\|y\|^2}{\sigma} \\ &\geq \frac{\|x\|^2}{\tau} - \frac{\|A\|\sqrt{\sigma\tau}}{2\tau}\|x\|^2\frac{\|A\|\sqrt{\sigma\tau}}{2\sigma}\|y\|^2 + \frac{\|y\|^2}{\sigma} \\ &= (1 - \sqrt{\tau\sigma}\|A\|^2)\left(\frac{\|x\|^2}{\tau} + \frac{\|y\|^2}{\sigma}\right)\end{aligned}$$

□

This lemma justifies the choice of L and the relevance of the obtained bound in the following theorem:

Theorem 3.5. *Consider problem (4) and suppose that there exists a saddle-point $\hat{z} = (\hat{x}, \hat{y})$. Let $L = \|A\|$ and choose constant step size $\tau\sigma L^2 < 1$. Then if $(z_n = (x_n, y_n), \tilde{x}_n, \tilde{y}_n)$ follow the PDHG updates:*

$$\begin{cases}\tilde{y}_n = y_n \\ x_{n+1} = \mathrm{Prox}_{\tau F}(x_n - \tau A^\intercal \tilde{y}_n) \\ \tilde{x}_{n+1} = 2x_{n+1} - x_n \\ y_{k+1} = \mathrm{Prox}_{\sigma G}(y_n + \sigma A\tilde{x}_{n+1})\end{cases} \tag{13}$$

we have:

(a) For any n,

$$\frac{\|y_n - \hat{y}\|^2}{2\sigma} + \frac{\|x_n - \hat{x}\|^2}{2\tau} \leq \frac{\|z_0 - \hat{z}\|^2_{W_{\tau,\sigma}}}{2(1-\sqrt{\tau\sigma}L)};$$

(b) If we let $\bar{x}_N = \left(\sum_{n=1}^N x^n\right)/N$ and $\bar{y}_N = \left(\sum_{n=1}^N y^n\right)/N$, then for any $(x, y) \in X \times Y$

$$\mathcal{L}(\bar{x}_N, y) - \mathcal{L}(x, \bar{y}_N) \leq \frac{\|z_0 - z\|^2_{W_{\tau,\sigma}}}{2N(1-\sqrt{\tau\sigma}L)}. \tag{14}$$

Moreover, the accumulation points of $(\hat{x}_N, \hat{y}_N)$ are saddle points of (8);

(c) There exists a saddle point (x^*, y^*) such that $x_n \to x^*$ and $y_n \to y^*$.

Proof. As seen earlier with (28) the PDHG updates can be written:

$$\begin{aligned} -A^\intercal \tilde{y}_n + \frac{(x_n - x_{n+1})}{\tau} &\in \partial F(x_{n+1}) \\ A\tilde{x}_{n+1} + \frac{(y_n - y_{n+1})}{\sigma} &\in \partial G(y_{n+1}) \end{aligned}$$

thus, by definition of a subgradient we have for any $(x, y) \in \mathbb{R}^N \times \mathbb{R}^M$

$$\begin{aligned} F(x) &\geq F(x_{n+1}) + \left(\frac{x_n - x_{n+1}}{\tau}\right)^\intercal (x - x_{n+1}) - \tilde{y}_n^\intercal A(x - x_{n+1}) \\ G(y) &\geq G(y_{n+1}) + \left(\frac{y_n - y_{n+1}}{\sigma}\right)^\intercal (y - y_{n+1}) + (y - y_{n+1})^\intercal A\tilde{x}_{n+1}. \end{aligned}$$

By summing both inequalities, we get using the identity $2a^\intercal b = \|a\|^2 + \|b\|^2 - \|a - b\|^2$:

$$\begin{aligned} &\frac{\|x - x_n\|^2}{2\tau} + \frac{\|y - y_n\|^2}{2\sigma} \geq \\ &\quad [F(x_{n+1}) + y^\intercal A x_{n+1} - G(y)] - [F(x) + y_{n+1}{}^\intercal A x - G(y_{n+1})] \\ &\quad + \frac{\|x - x_{n+1}\|^2}{2\tau} + \frac{\|y - y_{n+1}\|^2}{2\sigma} + \frac{\|x_n - x_{n+1}\|^2}{2\tau} + \frac{\|y_n - y_{n+1}\|^2}{2\sigma} \\ &+ (y_{n+1} - y)^\intercal A(x_{n+1} - \tilde{x}_{n+1}) - (y_{n+1} - \tilde{y}_n)^\intercal A(x_{n+1} - x) \end{aligned} \tag{15}$$

From the last line of (15), we see that the choice of $\tilde{x}_{n+1}$ and $\tilde{y}_n$ is important for the convergence analysis. As we first update x_{n+1} using $\tilde{y}_n$, it is logical to consider $\tilde{y}_n = y_n$. But then the question is how to choose $\tilde{x}_{n+1}$ for the y_{n+1} update. The standard approach over the years has been to consider updates of the form $\tilde{x}_{n+1} = x_{n+1} + \theta(x_{n+1} - x_n)$ [4]. PDHG corresponds to the case $\theta = 1$.

Thus, with $\tilde{x}_{n+1} = 2x_{n+1} - x_n$ and $\tilde{y}_n = y_n$ the last line of (15) becomes:

$$\begin{aligned}
&(y_{n+1} - y)^\intercal A(x_{n+1} - (2x_{n+1} - x_n)) - (y_{n+1} - y_n)^\intercal A(x^{n+1} - x) \\
&= (y_{n+1} - y)^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y_n)^\intercal A(x_{n+1} - x) \\
&= (y_n - y)^\intercal A(x_n - x_{n+1}) - (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y_n)^\intercal A(x_{n+1} - x) \\
&= (y_n - y)^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y)^\intercal A(x_{n+1} - x) - (y - y_n)^\intercal A(x_{n+1} - x) \\
&\quad - (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) \\
&= (y_n - y)^\intercal A(x_n - x) - (y_{n+1} - y)^\intercal A(x_{n+1} - x) - (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) \\
&\geq (y_n - y)^\intercal A(x_n - x) - (y_{n+1} - y)^\intercal A(x_{n+1} - x) \\
&\quad - \frac{L\sqrt{\tau\sigma}}{2\tau} \|x_n - x_{n+1}\|^2 - \frac{L\sqrt{\tau\sigma}}{2\sigma} \|y_n - y_{n+1}\|^2
\end{aligned} \tag{16}$$

where we have used in the last inequality that since $2ab \leq a^2 + b^2$:

$$\begin{aligned}
(y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) &\leq L\sqrt{\sigma/\tau} \|x_n - x_{n+1}\| \frac{\|y_n - y_{n+1}\|}{\sqrt{\sigma/\tau}} \\
&\leq \frac{L\sqrt{\sigma\tau}}{2\tau} \|x_n - x_{n+1}\|^2 + \frac{L\sqrt{\sigma\tau}}{2\sigma} \|y_n - y_{n+1}\|^2 .
\end{aligned}$$

Injecting (16) in (15) then yields:

$$\begin{aligned}
\frac{\|x - x_n\|^2}{2\tau} + \frac{\|y - y_n\|^2}{2\sigma} \geq [L(x_{n+1}, y) - L(x, y_{n+1})] \quad &+ \frac{\|x - x_{n+1}\|^2}{2\tau} + \frac{\|y - y_{n+1}\|^2}{2\sigma} \\
&+ (1 - L\sqrt{\tau\sigma})\frac{\|x_n - x_{n+1}\|^2}{2\tau} + (1 - L\sqrt{\tau\sigma})\frac{\|y_n - y_{n+1}\|^2}{2\sigma} \\
&+ (y_n - y)^\intercal A(x_n - x) - (y_{n+1} - y)^\intercal A(x_{n+1} - x)
\end{aligned} \tag{17}$$

where $L(x, y) = F(x) + y^\intercal Ax - G(y)$. (In particular $L(x, y) = \mathcal{L}(x, y) = f(x) + y^\intercal Ax - g(y)$ for all $(x, y) \in X \times Y$.)
By summing on both side of (17) from $n = 0$ to $N - 1$, we get:

$$
\begin{aligned}
\frac{1}{2}\|z - z_0\|^2_{W_{\tau,\sigma}} &= \frac{\|x - x_0\|^2}{2\tau} + \frac{\|y - y_0\|^2}{2\sigma} - (y_0 - y)^\intercal A(x_0 - x) \\
&\geq \sum_{n=1}^{N} [L(x_n, y) - L(x, y_n)] \\
&\quad + \frac{\|x - x_N\|^2}{2\tau} + \frac{\|y - y_N\|^2}{2\sigma} \\
&\quad + (1 - L\sqrt{\tau\sigma}) \left[\sum_{n=1}^{N} \frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|y_n - y_{n-1}\|^2}{2\sigma}\right] \\
&\quad - (y_N - y)^\intercal A(x_N - x) \\
&\geq \sum_{n=1}^{N} [L(x_n, y) - L(x, y_n)] \\
&\quad + (1 - L\sqrt{\tau\sigma}) \left[\frac{\|x - x_N\|^2}{2\tau} + \frac{\|y - y_N\|^2}{2\sigma}\right] \\
&\quad + (1 - L\sqrt{\tau\sigma}) \left[\sum_{n=1}^{N} \frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|y_n - y_{n-1}\|^2}{2\sigma}\right]
\end{aligned}
\tag{18}
$$

Now, take $(x, y) = (\hat{x}, \hat{y}) = \hat{z}$ a saddle point of (8). Then, by definition we have $L(x, \hat{y}) \geq L(x, y) \geq L(\hat{x}, y)$ for any $(x, y) \in \mathbb{R}^N \times \mathbb{R}^M$ and thus both $\sum_{n=1}^{N} [L(x_n, y) - L(x, y_n)]$ and $(1 - L\sqrt{\tau\sigma}) \left[\sum_{n=1}^{N} \frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|y_n - y_{n-1}\|^2}{2\sigma}\right]$ are non negative, which implies:

$$
(1 - L\sqrt{\tau\sigma}) \left[\frac{\|\hat{x} - x_N\|^2}{2\tau} + \frac{\|\hat{y} - y_N\|^2}{2\sigma}\right] \leq \frac{1}{2}\|\hat{z} - z_0\|^2_{W_{\tau,\sigma}}
$$

yielding point (a).

For point (b), let $\bar{x}_N = (\sum_{n=1}^{N} x_n)/N$ and $\bar{y}_N = (\sum_{n=1}^{N} y_n)/N$. Then using the convexity of L in x, the concavity of L in y, $1 - L\sqrt{\tau\sigma} > 0$ and (18) we have:

$$
L(\bar{x}_N, y) - L(x, \bar{y}_N) \leq \frac{2}{N}\|z \,\hat{-}\, z_0\|^2_{W_{\tau,\sigma}} \tag{19}
$$

From point (a), we know that x_n, y_n is bounded and thus so is $(\hat{x}_N, \hat{y}_N)$. Let (x^*, y^*) be an accumulation point of $(\hat{x}_N, \hat{y}_N)$. By lower semi continuity and convexity of F and G and (19) we then have:

$$
L(x^*, y) - L(x, y^*) \leq 0
$$

which shows that (x^*, y^*) satisfy (8) and therefore is a saddle point.

Finally, let (x_{n_k}, y_{n_k}) be a convergent subsequence of (x_n, y_n) (which is bounded by (a)) and let (x^*, y^*) be its limit. It is then a fixed point of (13) and by (9) it is a saddle point of 8. To prove the convergence of the whole sequence, take

$(x, y) = (x^*, y^*)$ in (17) and as before sum both side but this time from $n = n_k$ to $N - 1$, $N > n_k$:

$$\frac{1}{2}\|z^* - z_{n_k}\|^2_{W_{\tau,\sigma}} \geq (1 - L\sqrt{\tau\sigma})\left[\frac{\|x^* - x_N\|^2}{2\tau} + \frac{\|y^* - y_N\|^2}{2\sigma}\right] + \sum_{n=n_k+1}^{N}[L(x_n, y^*) - L(x^*, y_n)] + (1 - L\sqrt{\tau\sigma})\left[\sum_{n=n_k+1}^{N}\frac{\|x_n - x_{n-1}\|^2}{2\tau} + \frac{\|y_n - y_{n-1}\|^2}{2\sigma}\right].$$

In particular using similar arguments as before we obtain:

$$\frac{\|x^* - x_N\|^2}{2\tau} + \frac{\|y^* - y_N\|^2}{2\sigma} \leq \frac{1 - L\sqrt{\tau\sigma}}{2}\|z^* - z_{n_k}\|^2_{W_{\tau,\sigma}}$$

Letting k and thus N go to $+\infty$, we then easily deduce that $(x_N, y_N) \to (x^*, y^*)$. □

Remark 4. *This proof is heavily inspired by the one given for Theorem 1 in [4]. However, the update scheme here is slightly different and we find different upper bounds as well that correspond to remark 2 of [5] by the same authors.*

Remark 5. *At first glance it looks like the proof of theorem 3.5 could be slightly modified to prove the convergence with non constant step size $\tau_k\sigma_k$ such that $\tau_k\sigma_k L^2 < 1$ for all k by replacing $(1 - L\sqrt{\tau\sigma})$ with $(1 - \max_n \sqrt{\tau_n\sigma_n}L)$, but it is not that simple. Following the same steps as in Theorem 3.5 with τ_n, σ_n instead of τ, σ (17) would lead to:*

$$\sum_{n=0}^{N-1}\left[\frac{1}{2}\|z - z_n\|^2_{W_{\tau_n,\sigma_n}} - \frac{1}{2}\|z - z_{n+1}\|^2_{W_{\tau_n,\sigma_n}}\right] \geq \sum_{n=0}^{N-1}[L(x_n, y) - L(x, y_n)] + (1 - \max_n \sqrt{\tau_n\sigma_n}L)\sum_{n=0}^{N-1}\left[\frac{\|x_n - x_{n-1}\|^2}{2\tau_n} + \frac{\|y_n - y_{n-1}\|^2}{2\sigma_n}\right]$$

But this time, the term $\frac{1}{2}\|z - z_n\|^2_{W_{\tau_n,\sigma_n}}$ and $\frac{1}{2}\|z - z_{n+1}\|^2_{W_{\tau_n,\sigma_n}}$ will not cancel out:

$$
\begin{aligned}
\sum_{n=0}^{N-1} &\left[\frac{1}{2}\|z - z_n\|^2_{W_{\tau_n,\sigma_n}} - \frac{1}{2}\|z - z_{n+1}\|^2_{W_{\tau_n,\sigma_n}}\right] \\
&= \frac{1}{2}\|z - z_0\|^2_{W_{\tau_0,\sigma_0}} - \frac{1}{2}\|z - z_N\|^2_{W_{\tau_{N-1},\sigma_{N-1}}} \\
&\quad + \sum_{n=1}^{N} \frac{1}{2}\left[\frac{\|x - x_n\|^2}{\tau_{n-1}} + \frac{\|y - y_n\|^2}{\sigma_{n-1}} - \frac{\|x - x_n\|^2}{\tau_n} + \frac{\|y - y_n\|^2}{\sigma_n}\right] \\
&\leq \frac{1}{2}\|z - z_0\|^2_{W_{\tau_0,\sigma_0}} - \frac{1}{2}\|z - z_N\|^2_{W_{\tau_{N-1},\sigma_{N-1}}} \\
&\quad + \sum_{n=1}^{N} \frac{1}{2}\left[\max\left\{\frac{\tau_n - \tau_{n-1}}{\tau_{n-1}}, \frac{\sigma_n - \sigma_{n-1}}{\sigma_{n-1}}\right\}\left(\frac{\|x - x_n\|^2}{\tau_n} + \frac{\|y - y_n\|^2}{\sigma_n}\right)\right]
\end{aligned}
$$

To continue the proof as before, we see that we thus need bounds on the oscillations of the stepsizes τ_n and σ_n. This will be addressed in Section 3.3.

Theorem 3.5 proves convergence in a very straight forward way but requires previous estimation of $\|A\|$ for the stepsize choice. With large scale applications this can become a problem, but we can relax the assumptions on W in the following manner:

Theorem 3.6. *Consider problem (4) and suppose that there exists a saddle-point $\hat{z} = (\hat{x}, \hat{y})$. Let $(z_n = (x_n, y_n), \tau, \sigma)$ follow the PDHG updates:*

$$
\begin{cases}
x_{n+1} = \operatorname{Prox}_{\tau F}(x_n - \tau A^\intercal \tilde{y}_n) \\
y_{k+1} = \operatorname{Prox}_{\sigma G}(y_n + \sigma A(2x_{n+1} - x_n)).
\end{cases} \tag{20}
$$

If additionally for all iterations:

$$
\frac{1}{2}\|z_n - z_{n+1}\|^2_{W_{\tau,\sigma}} := \frac{\|x_{n+1} - x_n\|^2}{2\tau} + \frac{\|y_{n+1} - y_n\|^2}{2\sigma} - (y_{n+1} - y_n)^\intercal A(x_{n+1} - y_n) \geq 0 \tag{21}
$$

then we have:

(a) For any n,

$$
\frac{1}{2}\|\hat{z} - z_N\|^2_{W_{\tau,\sigma}} \leq \frac{1}{2}\|\hat{z} - z_0\|^2_{W_{\tau,\sigma}};
$$

(b) If we let $\bar{x}_N = \left(\sum_{n=1}^{N} x^n\right)/N$ and $\bar{y}_N = \left(\sum_{n=1}^{N} y^n\right)/N$, then for any $(x, y) \in X \times Y$

$$
\mathcal{L}(\bar{x}_N, y) - \mathcal{L}(x, \bar{y}_N) \leq \frac{1}{2N}\left(\|z \mathbin{\hat{-}} z_0\|^2_{W_{\tau,\sigma}} - \|z - z_N\|^2_{W_{\tau,\sigma}}\right). \tag{22}
$$

Proof. We first proceed as in the proof of Theorem 3.5.As before, from (28) we can write the PDHG update:

$$
\begin{aligned}
-A^\intercal y_n + \frac{(x_n - x_{n+1})}{\tau} &\in \partial F(x_{n+1}) \\
A(2x_{n+1} - x_n) + \frac{(y_n - y_{n+1})}{\sigma} &\in \partial G(y_{n+1})
\end{aligned}
$$

and by definition of a subgradient we have then for any $(x, y) \in \mathbb{R}^N \times \mathbb{R}^M$

$$\begin{aligned} F(x) &\geq F(x_{n+1}) + \left(\frac{x_n - x_{n+1}}{\tau} \right)^\intercal (x - x_{n+1}) - {y_n}^\intercal A(x - x_{n+1}) \\ G(y) &\geq G(y_{n+1}) + \left(\frac{y_n - y_{n+1}}{\sigma} \right)^\intercal (y - y_{n+1}) + (y - y_{n+1})^\intercal A(2x_{n+1} - x_n), \end{aligned}$$

which yields by summing both inequalities:

$$\begin{aligned} &\frac{\|x - x_n\|^2}{2\tau} + \frac{\|y - y_n\|^2}{2\sigma} \geq \\ &\quad [F(x_{n+1}) + y^\intercal A x_{n+1} - G(y)] - [F(x) + {y_{n+1}}^\intercal A x - G(y_{n+1})] \\ &\quad + \frac{\|x - x_{n+1}\|^2}{2\tau} + \frac{\|y - y_{n+1}\|^2}{2\sigma} + \frac{\|x_n - x_{n+1}\|^2}{2\tau} + \frac{\|y_n - y_{n+1}\|^2}{2\sigma} \\ &+ (y_{n+1} - y)^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y_n)^\intercal A(x_{n+1} - x). \end{aligned} \tag{23}$$

Now, we don't have estimation on $\|A\|$ and we have to proceed a bit differently. The last line of (23) is:

$$\begin{aligned} &(y_{n+1} - y)^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y_n)^\intercal A(x_{n+1} - x) \\ = &(y_n - y)^\intercal A(x_n - x_{n+1}) - (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y_n)^\intercal A(x_{n+1} - x) \\ = &(y_n - y)^\intercal A(x_n - x_{n+1}) - (y_{n+1} - y)^\intercal A(x_{n+1} - x) - (y - y_n)^\intercal A(x_{n+1} - x) \\ &- (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) \\ = &(y_n - y)^\intercal A(x_n - x) - (y_{n+1} - y)^\intercal A(x_{n+1} - x) - (y_n - y_{n+1})^\intercal A(x_n - x_{n+1}) \end{aligned} \tag{24}$$

Injecting (24) in (23) yields:

$$\frac{1}{2}\|z - z_n\|^2_{W_{\tau,\sigma}} \geq [L(x_{n+1}, y) - L(x, y_{n+1})] + \frac{1}{2}\|z - z_{n+1}\|^2_{W_{\tau,\sigma}} + \frac{1}{2}\|z_n - z_{n+1}\|^2_{W_{\tau,\sigma}} \tag{25}$$

By summing on both side of (25) from $n = 0$ to $N - 1$, we get:

$$\frac{1}{2}\|z - z_0\|^2_{W_{\tau,\sigma}} \geq \quad \sum_{n=1}^{N} [L(x_n, y) - L(x, y_n)] + \frac{1}{2}\|z - z_N\|^2_{W_{\tau,\sigma}} + \sum_{n=1}^{N} \frac{1}{2}\|z_{n-1} - z_n\|^2_{W_{\tau,\sigma}} \tag{26}$$

Now, take $(x, y) = (\hat{x}, \hat{y}) = \hat{z}$ a saddle point of (8). Then, $\sum_{n=1}^{N} [L(x_n, y) - L(x, y_n)]$ is non negative, and by the step size choice (21) so is $\sum_{n=1}^{N} \frac{1}{2}\|z_n - z_{n+1}\|^2_{W_{\tau,\sigma}}$. This implies that:

$$\frac{1}{2}\|z - z_N\|^2_{W_{\tau,\sigma}} \leq \frac{1}{2}\|z - z_0\|^2_{W_{\tau,\sigma}}$$

yielding point (a).

Let $\bar{x}_N = (\sum_{n=1}^{N} x_n)/N$ and $\bar{y}_N = (\sum_{n=1}^{N} y_n)/N$. Then using the convexity of L in x, the concavity of L in y, the step size choice (21) and (26) we have:

$$L(\bar{x}_N, y) - L(x, \bar{y}_N) \leq \frac{2}{N} \left(\|z \hat{-} z_0\|^2_{W_{\tau,\sigma}} - \|z - z_N\|^2_{W_{\tau,\sigma}} \right), \tag{27}$$

yielding point (b). □

Lemma 3.7. *Consider a sequence $(z_n) \subset X \times Y$ following the PDHG update (20). Suppose that X or Y is bounded. Then for any $z^* \in X \times Y$, $\|z_n - z^*\|^2_{W_{\tau,\sigma}}$ is lower bounded.*

Proof. Assume without loss of generality that X is bounded, i.e. there exists a constant $C_X \geq 0$ such that $\|x\| \leq C_X$ for all x in X. Then

$$\|z_n - z^*\|^2_{W_{\tau,\sigma}} \geq \frac{\|y_n - y^*\|^2}{\sigma} - 2C_X\|A\|\|y_n - y^*\|$$

As the right hand side is quadratic in $\|y_n - y^*\|$ with a positive second order coefficient, it is lower bounded which proves the lemma. □

Corollary 3.8. *Let x_n, y_n, τ, σ follow the assumptions of Theorem 3.6, $\bar{x}_N = \left(\sum_{n=1}^N x^n\right)/N$ and $\bar{y}_N = \left(\sum_{n=1}^N y^n\right)/N$. If either X or Y is bounded, then any accumulation point x^*, y^* of $\bar{x}_N, \bar{y}_N$ is a saddle point of (8).*

Proof. By Theorem 3.6 we have:

$$\mathcal{L}(\bar{x}_N, y) - \mathcal{L}(x, \bar{y}_N) \leq \frac{1}{2N}\left(\|z \hat{-} z_0\|^2_{W_{\tau,\sigma}} - \|z - z_N\|^2_{W_{\tau,\sigma}}\right).$$

Using Lemma 3.7 we know that there exists a constant C such that

$$\|z \hat{-} z_0\|^2_{W_{\tau,\sigma}} - \|z - z_N\|^2_{W_{\tau,\sigma}} \leq C \quad \forall N \in \mathbb{N}.$$

By lower semi continuity and convexity of F and G and we then have for any $(x, y) \in X \times Y$:

$$L(x^*, y) - L(x, y^*) \leq 0$$

which shows that (x^*, y^*) satisfy (8) and therefore is a saddle point. □

3.3 PDHG with non constant step sizes

We present here the residual balancing PDHG algorithm introduced in [6], but first we have to introduce the residuals as it is a key component of the algorithm. Remember that the cost for expressing the LP problem (1) in an unconstrained form was to use the non differentiable functions $F : \mathbb{R}^N \mapsto \mathbb{R} \cup \{\infty\}$ and $G : \mathbb{R}^M \mapsto \mathbb{R} \cup \{\infty\}$ and that we have derived the optimality condition:

$$\begin{aligned} 0 &\in \partial F(x^*) + A^\intercal y^* \\ 0 &\in \partial G(y^*) - Ax^*. \end{aligned}$$

However this formulation, it is not easy to keep track of the multi valued primal $\partial F(x^*) + A^\intercal y^*$ and dual $\partial G(y^*) - Ax^*$ residuals, but using Lemma 3.1, we have

$$\begin{aligned}
&\begin{cases} x_{k+1} = \mathrm{Prox}_{\tau_k F}(\hat{x}_k + 1) \\ y_{k+1} = \mathrm{Prox}_{\sigma_k G}(\hat{y}_k + 1) \end{cases} \\
\Leftrightarrow &\begin{cases} \hat{x}_{k+1} - x_{k+1} \in \partial(\tau_k F)(x_{k+1}) \\ \hat{y}_{k+1} - y_{k+1} \in \partial(\sigma_k G)(y_{k+1}) \end{cases} \\
\Leftrightarrow &\begin{cases} 0 \in \partial F(x_{k+1}) + A^\intercal y_k + \frac{1}{\tau_k}(x_{k+1} - x_k) \\ 0 \in \partial G(y_{k+1}) - A(2x_{k+1} - x_k) + \frac{1}{\sigma_k}(y_{k+1} - y_k) \end{cases} \\
\Leftrightarrow &\begin{cases} \frac{1}{\tau_k}(x_k - x_{k+1}) - A^\intercal(y_k - y_{k+1}) \in \partial F(x_{k+1}) + A^\intercal y_{k+1} \\ \frac{1}{\sigma_k}(y_k - y_{k+1}) - A(x_k - x_{k+1}) \in \partial G(y_{k+1}) - Ax_{k+1}. \end{cases}
\end{aligned} \tag{28}$$

Thus, by defining the sequences of residual

$$\begin{aligned}
P_{k+1} &\coloneqq \frac{1}{\tau_k}(x_k - x_{k+1}) - A^\intercal(y_k - y_{k+1}) \\
D_{k+1} &\coloneqq \frac{1}{\sigma_k}(y_k - y_{k+1}) - A(x_k - x_{k+1}),
\end{aligned} \tag{29}$$

we see that (x_k, y_k) converge to a solution when:

$$\lim_{k\to\infty} \|P_k\|_2 + \|D_k\|_2 = 0 \tag{30}$$

This heuristic will be used in Algorithm 2 in order to increase the convergence speed.

As we will see in the next theorem, even when $\tau_n\sigma_n\|A\|^2 < 1$, we need some conditions on the stepsizes for PDHG to converge.

Theorem 3.9. *Assume that problem (8) admits a solution $z^* = (x^*, y^*)$. Let $(z_n = (x_n, y_n), \tau_n, \sigma_n)$ follow the PDHG updates:*

$$\begin{cases} x_{n+1} = \mathrm{Prox}_{\tau_n F}(x_n - \tau A^\intercal \tilde{y}_n) \\ y_{k+1} = \mathrm{Prox}_{\sigma_n G}(y_n + \sigma A(2x_{n+1} - x_n)). \end{cases}$$

If we furthermore assume that τ_n, σ_n satisfy the following conditions:

- ***A*** *The sequences $\{\tau_n\}$ and $\{\sigma_n\}$ are bounded.*
- ***B*** *There exists a constant $0 < C_\phi < \infty$ such that*

$$\sum_{n=0} \phi_n \le C_\phi,$$

where $\phi_n \coloneqq \max\left\{\frac{\tau_n - \tau_{n+1}}{\tau_n}, \frac{\sigma_n - \sigma_{n+1}}{\sigma_n}, 0\right\}$.

- ***C*** *One of the following two conditions is met:*
  - ***C1*** *There is a constant L such that for all $n > 0$*

$$\tau_n\sigma_n < L < \frac{1}{\|A\|^2}.$$

C2 *Either X or Y is bounded, and there is a constant $\gamma \in (0,1)$ such that for all $n > 0$*

$$\frac{2(y_{n+1}-y_n)^\intercal A(x_{n+1}-x_n)}{\|x_{n+1}-x_n\|^2/\tau_n + \|y_{n+1}-y_n\|^2/\sigma_n} \le \gamma \tag{31}$$

Then:

(a) The algorithm converges in the residuals, i.e.

$$\lim_{n\to\infty} \|P_n\|^2 + \|D_n\|^2 = 0 \tag{32}$$

(b) If we define $x_N = \left(\sum_{n=1}^N x_n\right)/N$ and $y_N = \left(\sum_{n=1}^N y_n\right)/N$, for any $z \in X \times Y$ there exists some non negative constants C_1 and C_2 s.t.

$$\mathcal{L}(\bar{x}_N, y) - \mathcal{L}(x, \bar{y}_N) \le \frac{\|z-z_0\|^2_{W_{\tau_0,\sigma_0}} - \|z-z_N\|^2_{W_{\tau_N,\sigma_N}} + C1 + C2\|z-z^*\|^2}{2N}$$

and the algorithm converges with rate $\mathcal{O}(1/n)$.

The proof is given in [6]. Note that one can always find small enough stepsizes for (31) to hold using backtracking methods.

Here is the adaptive stepsize algorithm proposed in [6]:

Algorithm 2 Residual Balancing PDHG

```
Require: $x_0 \in \mathbb{R}^N, y_0 \in \mathbb{R}^M, \sigma_0\tau_0 < 1/\|A\|^2, (\alpha_0, \eta) \in (0,1)^2\Delta > 1, s > 0$
while $p_k$,$d_k$ > tolerance do
    $\hat{x}_{k+1} = x_k - \tau_k A^\intercal y_k$;
    $x_{k+1} = \text{Prox}_{\tau_k F}(\hat{x}_{k+1}) = \arg\min_{x\in X} f(x) + \frac{1}{2\tau_k}\|x - \hat{x}_{k+1}\|^2$;
    $\tilde{x}_{k+1} = x_{k+1} + (x_{k+1} - x_k)$;
    $\hat{y}_{k+1} = y_k + \sigma_k A\tilde{x}_{k+1}$;
    $y_{k+1} = \text{Prox}_{\sigma_k G}(\hat{y}_{k+1}) = \arg\min_{y\in Y} g(y) + \frac{1}{2\sigma_k}\|y - \hat{y}_{k+1}\|^2$;
    $p_{k+1} = \|P_{k+1}\|_2 = \|(x_k - x_{k+1}/\tau_k - A^\intercal(y_k - y_{k+1})\|_1$ ;
    $d_{k+1} = \|D_{k+1}\|_1 = \|(y_k - y_{k+1}/\sigma_k - A(x_k - x_{k+1})\|_1$;
    if $p_{k+1} > sd_{k+1}\Delta$ then
        $\tau_{k+1} = \tau_k/(1-\alpha_k)$;
        $\sigma_{k+1} = \sigma_k(1-\alpha_k)$;
        $\alpha_{k+1} = \alpha_k\eta$;
    end if
    if $p_{k+1} < sd_{k+1}/\Delta$ then
        $\tau_{k+1} = \tau_k(1-\alpha_k)$;
        $\sigma_{k+1} = \sigma_k/(1-\alpha_k)$;
        $\alpha_{k+1} = \alpha_k\eta$;
    end if
    if $sd_{k+1}/\Delta < p_{k+1} < sd_{k+1}\Delta$ then
        $\tau_{k+1} = \tau_k$;
        $\sigma_{k+1} = \sigma_k$;
        $\alpha_{k+1} = \alpha_k$;
    end if
end while
```

In line 7 and 8 we compute the primal and dual residual, and then we adapt the stepsizes form line 9 to 23 accordingly following the heuristic developed earlier.

Lemma 3.10. *Let* $(x_n, y_n, \tau_n, \sigma_n)$ *follow Algorithm 2. Then, they also satisfy conditions* $\mathbf{A}$, $\mathbf{B}$ *and* $\mathbf{C_1}$.

Proof. By construction, $\tau_n\sigma_n = \tau_0\sigma_0 < \frac{1}{\|A\|^2}$ for every iteration n and thus both $\mathbf{A}$ and $\mathbf{C_1}$ hold. For condition $\mathbf{B}$, note that

$$\phi_n = \begin{cases} \alpha_n & \text{if the stepsizes are updated} \\ 0 & \text{else} \end{cases}$$

and as $\alpha_{n+1} = \eta\alpha_n$ with $\eta < 1$ when the step size change, the sum is geometric and thus the series converges. □

4 First numerical experiments and restart motivation

4.1 Experimental set up

In order to numerically investigate the convergence of the different method we have studied until now, and in particular the number of iterations needed for the them to converge, we need to establish a termination criteria. By strong duality of Linear Programming, we know that $(x^*, y^*) \in X \times Y$ is a solution to problem (4) if and only if the primal objective $c^\intercal x^*$ is the same as the dual objective $q^\intercal y^* + l^\intercal\lambda^+ - u^\intercal\lambda^-$ and they satisfy the feasibility conditions. Thus, following Section 4.1 of [1], we say that our algorithm terminates with termination tolerance $\epsilon \in (0, \infty)$ when the iterates $x \in X, y \in Y, \lambda \in \Lambda$ satisfy:

$$\begin{aligned} |q^\intercal y + l^\intercal\lambda^+ - u^\intercal\lambda^- - c^\intercal x| &\leq \epsilon(1 + |q^\intercal y + l^\intercal\lambda^+ - u^\intercal\lambda^-| + |c^\intercal x|) \\ \left\| \begin{pmatrix} Ax - b \\ (h - Gx)^+ \end{pmatrix} \right\|_2 &\leq \epsilon(1 + \|q\|_2) \\ \|c - K^\intercal y - \lambda\|_2 &\leq (1 + \|c\|_2) \end{aligned} \tag{33}$$

where the first condition is the duality gap going to zero, the second is the primal feasibility and the third is the dual feasibility. Note that as our algorithm doesn't explicitly include the Lagrange variable λ (it is implicitly expressed by the restriction to primal solution in X via projection), and that is why we compute $\lambda = \mathbf{proj}_\Lambda(c - K^\intercal y)$. For all our numerical experiments we will initialize z_0 with all 0 entries and our stepsizes η and σ will be initialised at $\frac{1}{\sqrt{\|K\|}}$. For the residual balancing (non constant stepsize) PDHG, as recommended in [6] we use $a_0 = 0.5, \Delta = 1.5, \eta = 0.95$ and $s = \max(K)$. Finally, we use a termination tolerance of $\epsilon = 10^{-4}$.

4.2 Restart motivations

Consider the very simple saddle point problem:

$$\min_{x\in\mathbb{R}}\max_{y\in\mathbb{R}}\mathcal{L}(x,y)=\min_{x\in\mathbb{R}}\max_{y\in\mathbb{R}} ax+xy+by. \tag{34}$$

It admits the unique solution $(x^*,y^*) = (b,a)$. As we have seen previously (for example consider Theorem 3.5), we have convergence result for the current PDHG iterate z_n, but also for the average iterate $\bar{z}_n$. As we can see on Figure 1, when we use PDHG to solve problem (34) the last iterate z_n loops around the solution making steady but slow progress whereas the average iterates $\bar{z}_n$ approach very fast the solution at first but then have slow progress.

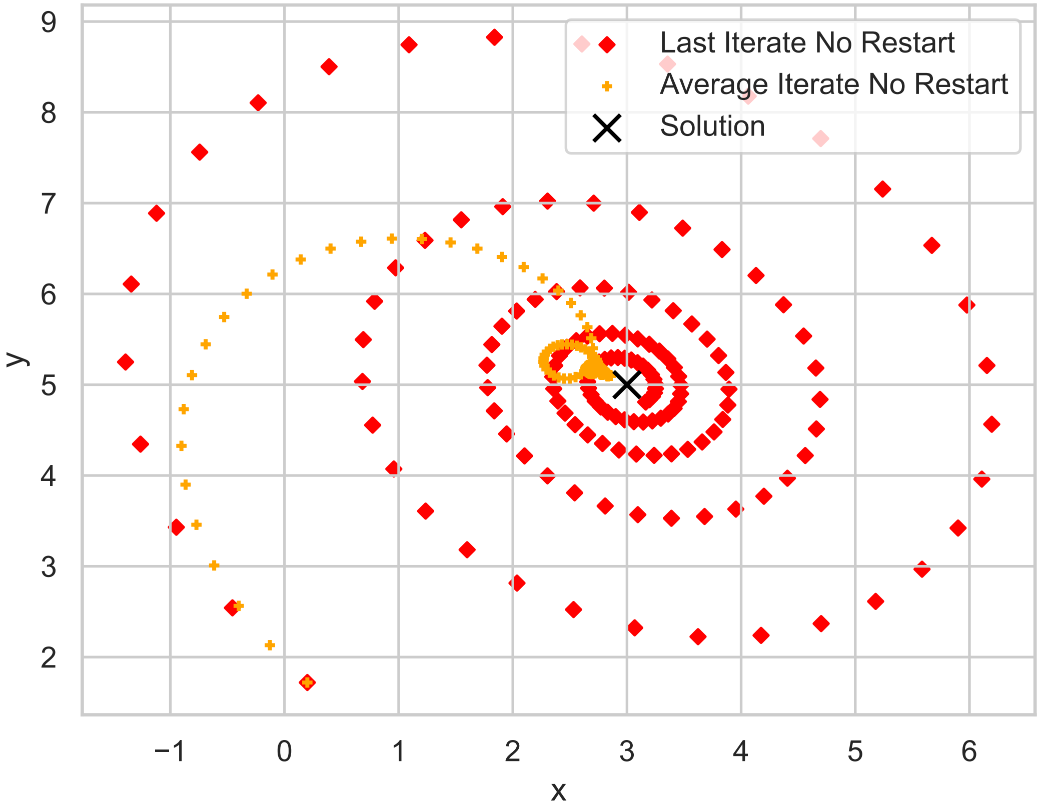

Figure 1: Plot of the first 150 iterates of the last and average PDHG iterate on problem (34) with $a=5, b=3$ and step sizes $\sigma=\tau=0.2$.

This is why in order to boost the steady pace of the last iterates, we introduce a restart method: whenever a prespecified restart condition is met, the algorithm restarts with the average of the iterates. This is illustrated in Figure 2 where the algorithm restart every 25 iterations. We can clearly see that both the last iterate and the average iterate converge much faster with the restart.

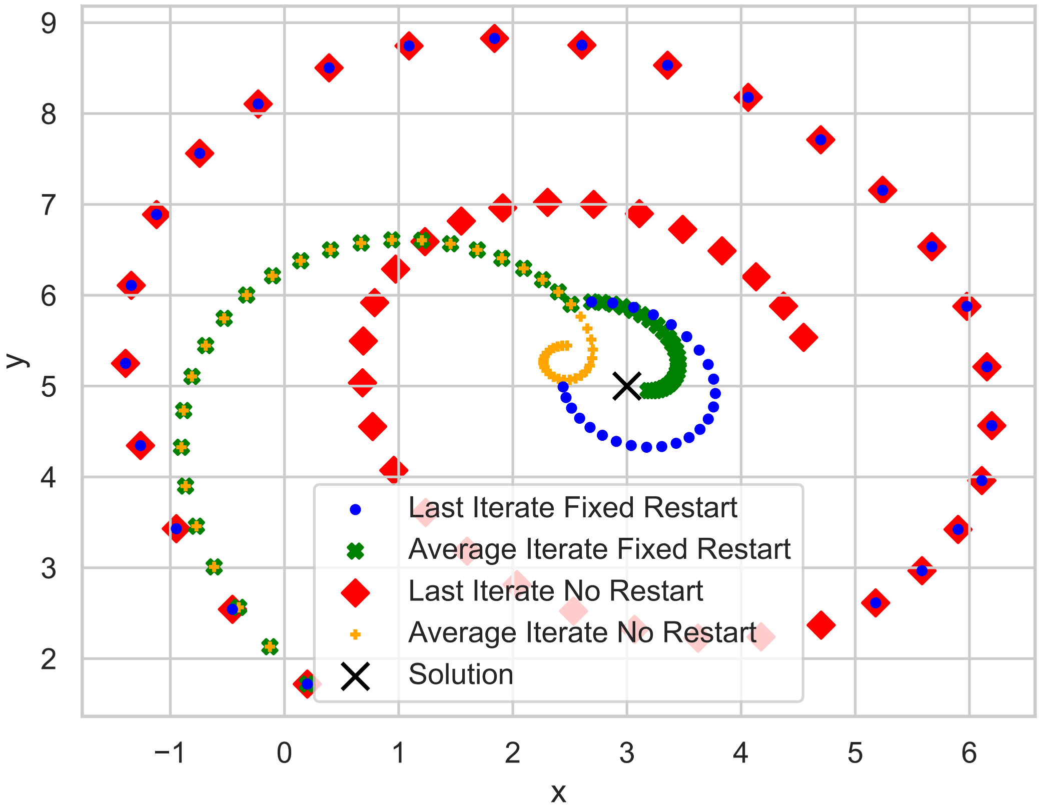

Figure 2: Plot of the first 50 iterates of non-restarted and restarted PDHG on problem (34) with $a = 5, b = 3$, step sizes $\sigma = \tau = 0.2$ and restart length 25 (if restart)
.

4.3 Comparison with adaptive PDHG and restart length investigation

To further investigate the effects of restart but also the usefulness of the adaptive stepsize PDHG described in Algorithm 2, we consider the relatively small LP problem "bk4x3"[1] ($n = 24$, $m_1 = 12$ and $m_2 = 7$).

Table 1 describes the number of iterations needed for each PDHG version to converge. We can first see that even for a quite small LP instance, PDHG takes a lot of iterations before meeting the termination criteria, and the residual balancing version where the step sizes are adjusted doesn't perform too well either. This can be explained by the potentially poor choice of hyper parameters $(s, \eta, a_0, \Delta ...)$ but we have found in practice that contrary to what was found for image processing problems in [6] there are no general good choice of them for LP problems. We can also note that the adaptive stepsize method doesn't combine well with fixed frequency restart in general. Finally, the restart length

[1] This LP instance can be found on http://plato.asu.edu/ftp/lptestset/fctp/.

Restart length	Constant stepsizes	Adaptive stepsizes
∞ (no restart)	15723	14650
50	7077	9377
100	5046	14673
500	6097	15440
1000	7047	16723
3000	9396	18678
5000	11167	20921

Table 1: Comparison of the number of iterations needed before convergence for different PDHG enhancements on problem bk4x3

seem to be quite important for the algorithm performance with the best fixed restart length being 100 for this problem.

Let's compare now the behaviour of the iterates of the restarted and non restarted PDHG with restart length 1000. From Table 1, we know that the non restarted version requires 15723 iterations, but we can see in Figure 3 that it approaches the solution quite fast compared to the last iterate with restart. However, as we can see in Figure 4, the regularising effect of the restart makes the convergence of the restarted PDHG iterates faster.

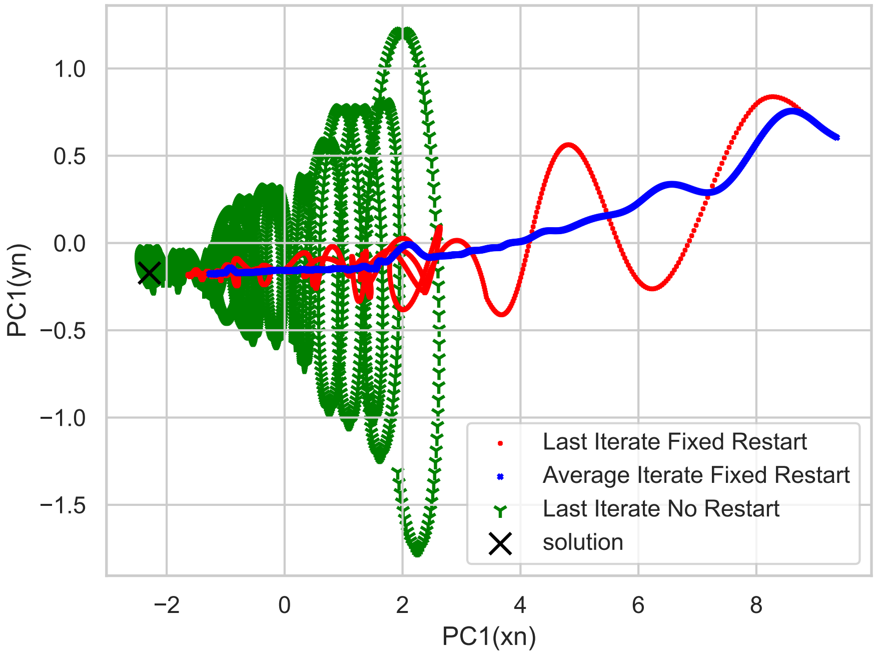

Figure 3: Scatter plot comparing the first principal component of the dual part versus the first principal component of the primal part for iterations 1000 to 4500 of non-restarted and restarted PDHG with a restart length of 1000 on problem bk4x3.

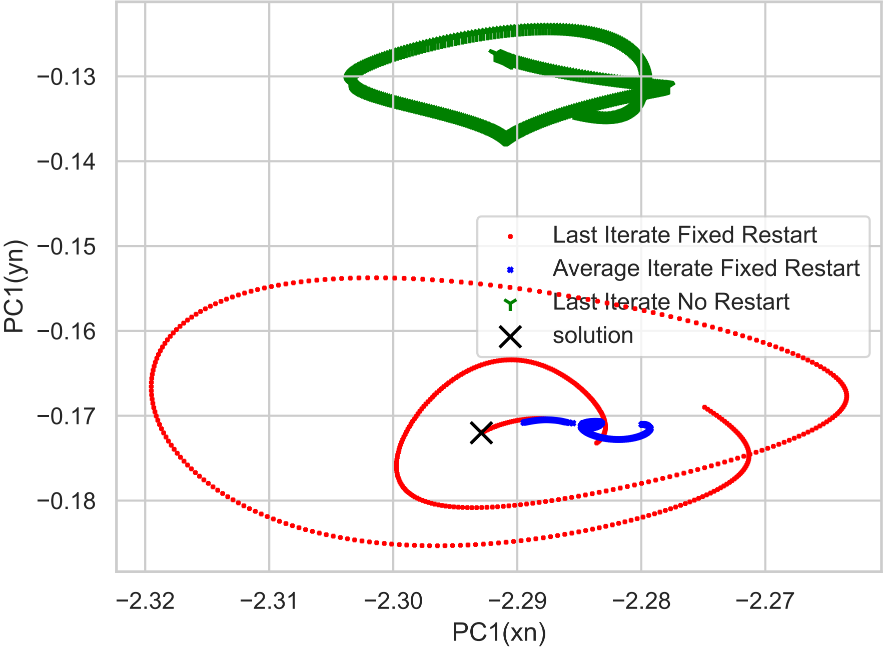

Figure 4: Scatter plot comparing the first principal component of the dual part versus the first principal component of the primal part for iterations 7000 to 8000 of non-restarted and restarted PDHG with a restart length of 1000 on problem bk4x3.

Recall the definition of the residuals $p_n = \|P_n\|_1$ and $d_n = \|D_n\|_1$ where P_n and D_n were introduced earlier (29). We have seen that convergence of PDHG is equivalent to convergence of the residuals to 0, but as we can see on Figure 5 and Figure 6, they oscillate a lot making it a poor metric to track the progression of the algorithm. As the algorithm attempts to approach both the primal and dual solution at each iterations this behaviour is not surprising. We have experimented some restart condition based on the comparison of the residuals of the average $\bar{z}_n$ and last iterate z_n but without much success due to this oscillation phenomena. One can also note on Figure 6 that at each restart (every 1000 iterations), the residual spike toward zero showing the efficiency of the restart.

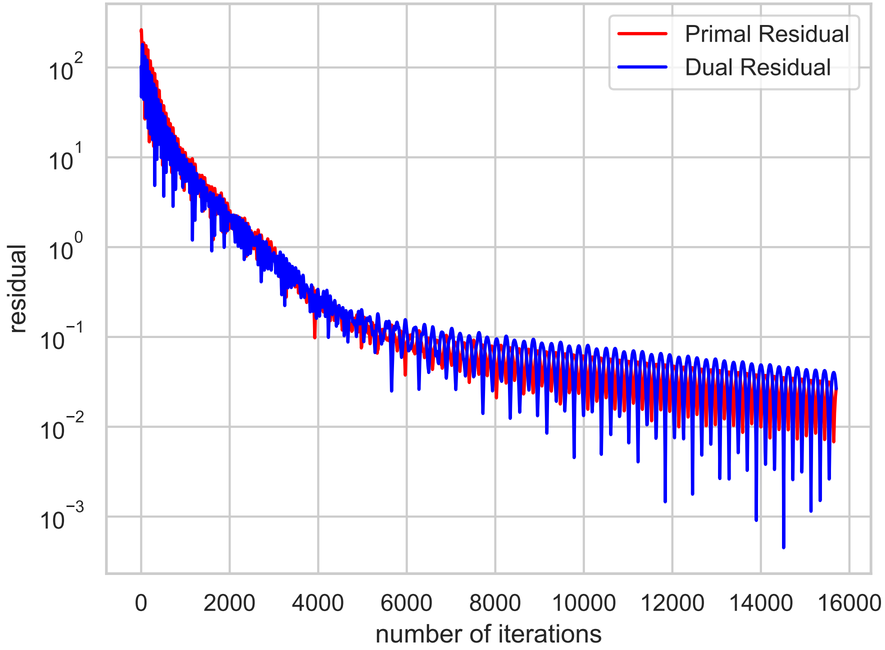

Figure 5: Primal and dual residual by number of iterations for simple PDHG on problem b4kx3.

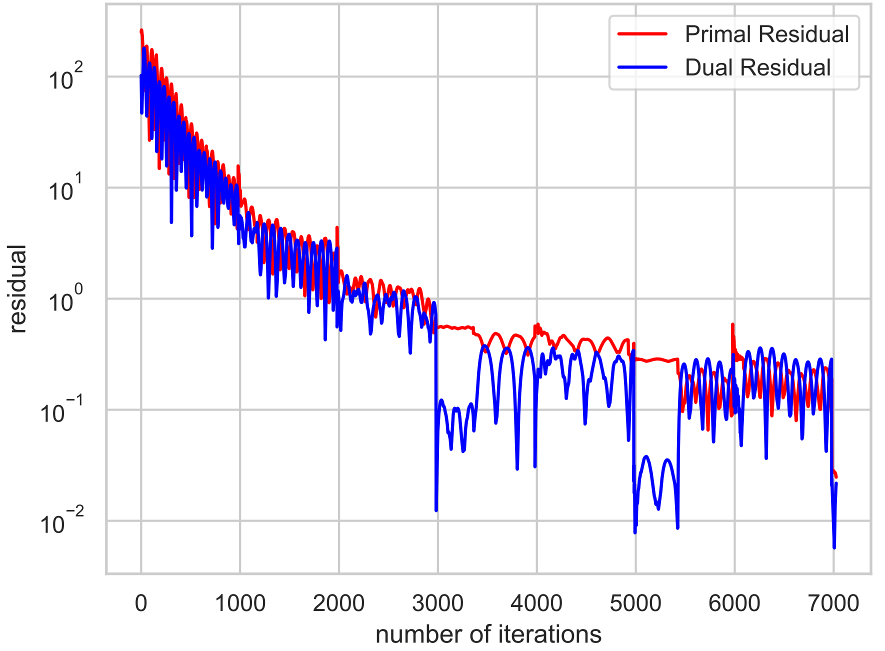

Figure 6: Primal and dual residual by number of iterations for simple PDHG on problem b4kx3.

As we have just seen, to fully exploit the restart potential one need to find a good restart condition. This is the motivation for introducing the normalised duality gap in the next section.

5 Adaptive restart PDHG for LP

From now on, we will focus only on PDHG for LP, i.e. PDHG on the saddle point problem (4). In this section, we review the theory on PDHG with adaptive restart from [2]. We start this section by introducing the key element of this algorithm, the normalised duality gap.

5.1 Normalised duality gap theory

By lemma 2.3, we know that $(x^*, y^*) \in Z$ is a saddle point if and only if the primal dual gap is 0:

$$\max_{\hat{z} \in Z} \{\mathcal{L}(x, \hat{y}) - \mathcal{L}(\hat{x}, y)\} = 0.$$

It might be tempting to track the progress of the PDHG algorithm using the primal dual gap, but we know that Y is unbounded by definition and this quantity may be infinite.

This motivates the use of the normalized duality gap:

$$\rho_r(z) := \frac{\max_{\hat{z} \in W_r(z)} \{\mathcal{L}(x, \hat{y}) - \mathcal{L}(\hat{x}, y)\}}{r},$$

where $W_r(z) := \{\hat{z} \in Z | \|z - \hat{z}\|\}$ is the ball of radius $r \in (0, \infty)$ and center z and $\|\cdot\|$ is a semi norm. In [2] for PDHG, and in the following, this semi norm is the norm $\|z\|_{M_{\tau,\sigma}} = \sqrt{z^\intercal M_{\tau,\sigma}}$ where

$$M_{\tau,\sigma} = \begin{pmatrix} \mathbf{I} & -\tau K^\intercal \\ -\sigma K & \mathbf{I} \end{pmatrix}$$

and the step sizes τ, σ are chosen such that $\tau\sigma\|K\|^2 \leq 1$. In this setting, the normalised duality gap is always defined and as expressed in the following lemmas the normalised duality gap is a good measure of optimality.

Lemma 5.1. *For any fixed $z \in Z$, $\rho_r(z)$ is non increasing for $r \in (0, \infty)$. Consequently, one can define $\rho_0(z) := \limsup_{r \to 0^+} \rho_r(z)$.*

Proof. Consider any $z \in Z$ and $0 < r_1 < r_2$. For any $z_2 \in \operatorname{argmax}_{\hat{z} \in W_{r_2}(z)}\{\mathcal{L}(x, \hat{y}) - \mathcal{L}(\hat{x}, y)\}$ we have by definition of $W_r(z)$ that $z_1 := z + \frac{r_1}{r_2}(z_2 - z) \in W_{r_1}(z)$. Thus:

$$\begin{aligned}
\rho_{r_2}(z) &= \frac{\mathcal{L}(x, y_2) - \mathcal{L}(x_2, y)}{r_2} \\
&= \frac{(\mathcal{L}(x, y_2) - \mathcal{L}(x, y)) + (\mathcal{L}(x, y) - \mathcal{L}(x_2, y))}{r_2} \\
&\leq \frac{(\mathcal{L}(x, y_1) - \mathcal{L}(x, y)) + (\mathcal{L}(x, y) - \mathcal{L}(x_1, y))}{r_1} \\
&\leq \rho_{r_1}(z).
\end{aligned}$$

□

where we have used for the first inequality the convexity of $\mathcal{L}(x,y)$ in x and the concavity of $\mathcal{L}(x,y)$ in y.

Lemma 5.2. *For any $r \in [0,\infty]$, the primal dual gap at z is 0 if and only if $\rho_r(z) = 0$.*

Proof. Suppose that the primal-dual gap at z is nonzero. Then there exists $\hat{z} \in Z$ such that $\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y) > 0$. Let $r \in (0,\infty)$, then there exists $\lambda > 0$ such that the convex combination $(1-\lambda)z + \lambda\hat{z} \in W_r(z)$ (take for instance $\lambda = \min\left\{1, \frac{r}{\|\hat{z}-z\|}\right\}$). But then by convexity of $\mathcal{L}(x,y)$ in x and concavity of $\mathcal{L}(x,y)$ in y we get:

$$\begin{aligned} 0 &< \frac{\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y)}{\|\hat{z}-z\|} \\ &\leq \frac{\mathcal{L}(x,y+\lambda(\hat{y}-y)) - \mathcal{L}(y+\lambda(\hat{y}-y),y)}{\lambda\|\hat{z}-z\|} \\ &\leq \frac{r}{\lambda\|\hat{z}-z\|}\rho_r(z). \end{aligned}$$

But as r was chosen arbitrarily, this yields that the first implication.

For the other direction, suppose that the primal-dual gap at z is zero. Then by definition of ρ_r we get for all $r \in (0,\infty)$ that

$$0 \leq \rho_r(z) \leq \frac{1}{r}\max_{\hat{z}\in Z}\{\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y)\} = 0$$

which ends the proof. □

5.2 The algorithm

The restarted PDHG algorithm presented in [2] takes the following form (when the primal dual step are computed with PDHG):

Algorithm 3 Restarted PDHG for LP

```
Require: $z_{0,0} \in Z, \sigma\tau < 1/\|K\|^2$
$n \leftarrow 0$;                                                        ▷ Number of restarts
while Not converged do
    $t \leftarrow 0$;                                                    ▷ Inner loop count
    repeat
        $x_{n,t+1} \leftarrow \mathbf{proj}_X \left(x_{n,t} - \tau(c - K^\intercal y_{n,t})\right)$ ;   ▷ PDHG steps
        $y_{n,t+1} \leftarrow \mathbf{proj}_Y \left(y_{n,t} + \sigma(q - K(2x_{n,t+1} - x_{n,t}))\right)$;
        $z_{n,t+1} \leftarrow (x_{n,t+1}, y_{n,t+1})$;
        $\bar{z}_{n,t+1} \leftarrow \frac{1}{t+1} \sum_{i=1}^{t+1} z_{n,i}$;   ▷ Compute the average iterate
        $t \leftarrow t + 1$;
    until Restart Condition
    $\tau_n \leftarrow t$ ;
    $z_{n+1,0} \leftarrow \bar{z}_{n,\tau_n}$ ;                          ▷ Restart with the average iterate
    $n \leftarrow n + 1$ ;
end while
Output $z_{n,0}$,
```

where the restart condition corresponds to restarting when:

$$\begin{cases} \rho_{\|\bar{z}_{n,t} - z_{n,0}\|}(\bar{z}_{n,t}) \leq \beta \rho_{\|z_{n,0} - z_{n-1,0}\|}(z_{n,0}) & \text{if } n \geq 1 \\ t \geq \tau_0 & \text{if } n = 0 \end{cases} \tag{35}$$

with hyper parameters $\tau_0 \geq 1$ which is the number of iteration before the first restart and $0 < \beta < 1$. In other terms, we restart when the average iterate is a better candidate in term of normalised duality gap then the current iterate by a factor β. Note that we have directly replaced the PDHG steps written with proximal operator by a simpler form with projections according to lemma 3.2.

Theorem 5.3. *Let Z^* be the set of solutions to the saddle point formulation of LP (4). Assume $Z^* \neq \emptyset$ and consider the sequence $\{z_{n,0}\}_{n=0}^{\infty}$ generated by Algorithm 3. Then there exists a constant $C > 0$ such that:*

$$\mathbf{dist}(z_{n,0}, Z^*) \leq \beta^n C \mathbf{dist}(z_{0,0}, Z^*). \tag{36}$$

In other terms, the distance between the iterate $z_{n,0}$ and a primal dual solution decays linearly.

Proof. See [2] □

6 PDLP

We are now ready to study PDLP, the recent LP problem solver algorithm presented in [1]. As PDLP is more motivated by heuristic and numerical experiments rather than formal theory, in a first time we will comment the implementation choices in light of the theory discussed in the previous sections and then we will study PDLP numerically. We will soon see that PDLP is PDHG with adaptive step sizes and restart, applied on a presolved version of LP with diagonal preconditioning.

6.1 The algorithm

PDHG is known to converge (see Theorem 3.5 or Theorem 3.9 for non constant step size) when the stepsizes τ, σ satisfy $\tau\sigma\|K\|^2 < 1$. To have a better control of the scaling between the primal and dual iterates, in PDLP the stepsizes are reparameterized by:

$$\tau = \eta/\omega \quad \text{and} \quad \sigma = \omega\eta \quad \text{with } \eta \in (0,\infty) \quad \text{and} \quad \omega \in (0,\infty). \tag{37}$$

With this parameterization, PDHG converges if the *step size* η is such that $\eta\|K\| < 1$ and we can thus use a single parameter, the *primal weight* ω, to monitor the scaling between the primal and dual iterates. The authors of [1] have chosen to use the norm:

$$\|z\|_w := \sqrt{w\|x\|_2^2 + \frac{\|y\|_2^2}{w}} \tag{38}$$

for PDLP and in particular the normalised duality gap. One can note that we have already encounter a really similar norm in our convergence analysis (Theorem 3.5):

$$\frac{\|x\|^2}{\tau} + \frac{\|y\|^2}{\sigma} = \frac{1}{\eta}\left(\omega\|x\|^2 + \frac{\|y\|^2}{\omega}\right).$$

Moreover, as PDLP uses non constant step size which potentially violate the condition $\eta_n\|K\| < 1$, it wouldn't have been possible to use instead $\|z\|_{M_{\eta_n/\omega,\eta_n\omega}}$ as in Section 5 as this quantity may be negative (as implicitely stated by Lemma 3.4).

The PDLP algorithm pseudo code is as follows:

Algorithm 4 PDLP (after preconditioning and presolve)

```
Require: $z_{0,0} \in Z$
$k \leftarrow 0$;                                              ▷ Total number of iterations
$n \leftarrow 0$ ;                                             ▷ Number of restarts
$\hat{\eta}_{0,0} \leftarrow 1/\|K\|_\infty$;                  ▷ Initialize the step size
$\omega_0 \leftarrow$ InitializePrimalWeight$(c, q)$;
repeat
    $t \leftarrow 0$;                                          ▷ Inner loop count
    repeat
        $z_{n,t+1}, \eta_{n,t+1}, \hat{\eta}_{n,t+1} \leftarrow$ adaptiveStepOfPDHG$(z_{n,t}, \omega_n, \hat{\eta}_{n,t}, k)$;
        $\bar{z}_{n,t+1} \leftarrow \frac{1}{\sum_{i=1}^{t+1}\eta_{n,i}} \sum_{i=1}^{t+1} \eta_{n,i} z_{n,i}$;   ▷ Compute the average iterate
        $z^c_{n,t+1} \leftarrow$ GetRestartCandidate$(z_{n,t+1}, \bar{z}_{n,t+1}, z_{n,0})$;
        $t \leftarrow t+1$, $k \leftarrow k+1$;
    until restart or termination criteria holds
    $z_{n+1,0} \leftarrow z^c_{n,t}$;                          ▷ Restart with the new candidate
    $n \leftarrow n+1$ ;
until termination criteria holds
Output $z_{n,0}$,
```

Algorithm 5 One PDHG step with backtracking step size calculation

```
function adaptiveStepOfPDHG($z_{n,t}, \omega_n, \hat{\eta}_{n,t}, k$)
    $(x, y) \leftarrow z_{n,t}$, $\eta \leftarrow \hat{\eta}_{n,t}$;
    for $i = 1, ..., \infty$ do
        $x' \leftarrow \mathbf{proj}_X(x - \frac{\eta}{\omega_n}(c - K^\intercal y))$;
        $y' \leftarrow \mathbf{proj}_Y(y - \eta\omega_n(q - K(2x' - x)))$;
        $\bar{\eta} = \frac{\|(x'-x, y'-y)\|_{\omega_n}^2}{2(y'-y)^\intercal K(x'-x)}$;
        $\eta' \leftarrow \min\{(1 - (k+1)^{-0,3})\bar{\eta}, (1 + (k+1)^{-0.6})\eta\}$;
        if $\eta \leq \bar{\eta}$ then
            return $(x', y'), \eta, \eta'$;
        end if
        $\eta \leftarrow \eta'$;
    end for
```

6.2 Step size choice

Algorithm 5 is a PDHG update where we ensure by backtracking the following bound on the step size η:

$$\eta \leq \frac{\|z_{k+1} - z_k\|_\omega^2}{2(y_{k+1} - y_k)^\intercal K(x_{k+1} - x_k)}. \tag{39}$$

In [1], it is claimed that this heuristic comes from the condition given in Theorem 1 of [5] (note that in our case $L_f = 0$ as $f(x) = c^\intercal x$):

$$\begin{aligned} &\frac{1}{\tau}\|x - x'\|_2^2 + \frac{1}{\sigma}\|y - y'\|_2^2 \geq 2(y - y')^\intercal K(x - x') \\ \Leftrightarrow &\frac{1}{\eta} \geq \frac{2(y - y')^\intercal K(x - x')}{\|z - z'\|_\omega^2} \end{aligned} \tag{40}$$

but (40) is different on multiple levels to (39). First (40) must hold for every $x, x' \in X$ and $y, y' \in Y$ to prove the convergence whereas (39) only asks for the condition to hold for consecutive iterates z_k, z_{k+1}. This is what has motivated Theorem 3.6 and Corollary 3.8 but we have only obtained results a bit weaker than convergence. Second, it is not clear that

$$2(y_{k+1} - y_k)^\intercal K(x_{k+1} - x_k) > 0 \tag{41}$$

and we found in practice that it is not always the case. That's why in our implementation of PDLP[2] we first check whether (41 holds) before doing the backtracking steps proposed in Algorithm 5. If it doesn't hold, then naturally:

$$\frac{1}{\eta} \geq 0 > \frac{2(y_k -_{k+1})^\intercal K(x_k - x_{k+1})}{\|z_k - z_{k+1}\|_\omega^2}$$

and we can resume to the other steps of the PDLP algorithm.

[2] https://github.com/CorentinTissier/PDLP

6.3 Restart Implementation

There are a few difference with the restarted PDHG algorithm proposed in Section 5. First of all, the norm used in the definition of the normalised duality gap is $\|z\|_{\omega_n}$ instead of $\|z\|_{M_{\eta/\omega_n,\eta\omega_n}}$ which was defined in Section 5.1. We then have:

$$\rho_r^n(z) = \frac{1}{r} \underset{\hat{z}\in Z:\|z-\hat{z}\|_{\omega_n}\leq r}{\text{maximize}} \{\mathcal{L}(x,\hat{y}) - \mathcal{L}(\hat{x},y)\} \tag{42}$$

and for brevity we define:

$$\mu_n(z, z_{\text{ref}}) := \rho^n_{\|z-z_{\text{ref}}\|_{\omega_n}}(z).$$

We will see in Section 7 how to compute $\rho_r^n(z)$ in linear time.

An other difference with Algorithm 3 is that the restart candidate is not necessarily the average iterate anymore $\bar{z}_{n,t+1}$ (or rather in this case the weighted average by the step sizes $(\eta_{n,i})_{i=1}^{t+1}$). The restart candidate $z^c_{n,t+1}$ is given by:

$$\texttt{GetRestartCandidate}(z_{n,t+1}, \bar{z}_{n,t+1}, z_{n,0}) := \begin{cases} z_{n,t+1} & \mu_n(z_{n,t+1}, z_{n,0}) < \mu_n(\bar{z}_{n,t+1}, z_{n,0}) \\ \bar{z}_{n,t+1} & \text{otherwise.} \end{cases}$$

Instead of always restarting with the average iterate $\bar{z}_{n,t+1}$, we do it only if the progress in terms of normalised duality gap is better with it than with the last iterate $z_{n,t+1}$.

Finally, the restart condition themselves are a bit different. The authors of [1] use three potential restart conditions:

(i) (**Sufficient decay in normalized duality gap**)
$\mu_n\left(z^c_{n,t+1}, z_{n,0}\right) \leq \beta_{\text{sufficient}}\mu_n\left(z_{n,0}, z_{n-1,0}\right)$,

(ii) (**Necessary decay + no local progress in normalized duality gap**)
$\mu_n\left(z^c_{n,t+1}, z_{n,0}\right) \leq \beta_{\text{necessary}}\mu_n\left(z_{n,0}, z_{n-1,0}\right)$ and
$\mu_n\left(z^c_{n,t+1}, z_{n,0}\right) > \mu_n\left(z^c_{n,t}, z_{n,0}\right)$

(iii) (**Long inner loop**)
$t \geq \beta_{\text{artificial}}k$

where $\beta_{\text{sufficient}} \in (0,1), \beta_{\text{necessary}} \in (0, \beta_{\text{sufficient}})$ and $\beta_{\text{artificial}} \in (0,1)$. In particular, they recommend using $\beta_{\text{sufficient}} = 0.9, \beta_{\text{necessary}} = 0.1$, and $\beta_{\text{artificial}} = 0.5$. The algorithm restarts if one of the three condition holds.

Condition (i) is the same as presented in Section 3 and it guarantees the linear convergence of restarted PDHG on LP problems. The second condition in (ii) is inspired by adaptive restart schemes for accelerated gradient descent where restarts are triggered if the function value increases. That is, in our case, if the normalised duality gap increases with our new candidate $z^c_{n,t+1}$. The first inequality prevents the algorithm restarting every inner iteration or never restarting. The third condition ensures that the algorithm restarts an infinite number of time and thus the primal weights ω_n are updated frequently enough to prevent progress from stalling for a long time.

6.4 Primal weight update

While the stepsize η controls the general convergence of the algorithm, the primal weight ω allows the algorithm to balance the progress between the primal part that lies in X of the iterate and its dual part that lies in Y. Contrary to what was proposed in Section 3.3, the authors of [1] have chosen to find this balance by choosing ω_n such that the distances to optimality in the primal and dual are the same, i.e.:

$$\begin{aligned} & \|(x_{n,t} - x^*, \mathbf{0})\|_{\omega_n} \simeq \|(\mathbf{0}, y_{n,t} - y^*)\|_{\omega_n} \\ \Leftrightarrow \quad & \sqrt{\omega_n}\|x_{n,t} - x^*\|_2 \simeq \frac{1}{\sqrt{\omega_n}}\|y_{n,t} - y^*\|_2 \\ \Leftrightarrow \quad & \omega_n \simeq \frac{\|y_{n,t} - y^*\|_2}{\|x_{n,t} - x^*\|_2}. \end{aligned}$$

(note the small typo in section 3.3 of [1] where $\|(x_{n,t} - x^*, \mathbf{0})\|_{\omega_n} = \omega_n\|x_{n,t} - x^*\|_2$ was used instead of $\|(x_{n,t} - x^*, \mathbf{0})\|_{\omega_n} = \sqrt{\omega_n}\|x_{n,t} - x^*\|_2$.)

Since the quantity $\frac{\|y_{n,t} - y^*\|_2}{\|x_{n,t} - x^*\|_2}$ is not computable before knowing the solution z^*, it is estimated using instead $\frac{\|y_{n,0} - y_{n-1,0}\|_2}{\|x_{n,0} - x_{n-1,0}\|_2}$:

Algorithm 6 Primal weight update

```
function PrimalWeightUpdate($z_{n,0}, z_{n-1,0}, \omega_{n-1}$)
    $\Delta_x^n = \|x_{n,0} - x_{n-1,0}\|_2$, $\Delta_y^n = \|y_{n,0} - y_{n-1,0}\|_2$;
    if $\Delta_x^n > \epsilon_{\text{zero}}$ and $\Delta_y^n > \epsilon_{\text{zero}}$ then
        return $\exp\left(0.5\log\left(\frac{\Delta_y^n}{\Delta_x^n}\right) + 0.5\log(\omega_{n-1})\right)$;
    else
        return $\omega_{n-1}$;
    end if
```

with ϵ_{zero} a small non zero tolerance. Note that instead of directly picking $\omega_n = \frac{\|y_{n,0} - y_{n-1,0}\|_2}{\|x_{n,0} - x_{n-1,0}\|_2}$, some exponential smoothing is used to dampen variations in ω_n as we only modify it at restart and it can therefore oscillate a lot. The primal weight is initialised at the start of PDLP with:

$$\texttt{InitializePrimalWeight}(c, q) := \begin{cases} \frac{\|c\|_2}{\|q\|_2} & \|c\|_2, \|q\|_2 > \epsilon_{\text{zero}} \\ 1 & \text{otherwise.} \end{cases}$$

6.5 Presolve and Diagonal Preconditioning

Presolve and Preconditioning are two methods that change the instance of the problem to make it easier to solve and are not directly part of the algorithm. More information about it can be found in Section 3.4 and 3.5 of [1] and we will not study the effects of these methods here.

6.6 Numerical Experiments

In this section, we compare PDHG, fixed restart PDHG, adaptive restart PDHG and our version of PDLP, i.e. adaptive restart PDHG with adaptive stepsizes. As before, we use the termination criteria (33) with $\epsilon = 10^{-4}$. We will consider three LP problems with sizes described in Table 2. All of them can be accessed from http://plato.asu.edu/ftp/lptestset/fctp/.

Problem name	n	m_1	m_2
bk4x3	24	12	7
ran10x10b	200	100	20
ran17x17	578	289	34

Table 2: Test problem sizes.

The results of our experiment are stated in Table 3. For the fixed restart schemes we have used a restart length of 1000. Note that in order to simplify the complexity of the adaptive restart methods (Adaptive Restart and PDLP) we only check the restart condition (and thus calculate the normalised duality gap) every 100 iterations. We can see that the fixed restart scheme doesn't necessarily improve the performance of PDHG as illustrated with ran10x10b. On the other hand, PDLP improves the convergence speed over PDHG significantly in every of these test instances which confirms the potential of these enhancements. We finally note that only adaptive restart with no stepsizes update is not always enough to improve the performance of PDHG as illustrated once again with ran10x10b. Our python version of PDLP is available at https://github.com/CorentinTissier/PDLP.

Problem name	PDHG	Fixed Restart	Adaptive Restart	PDLP
bk4x3	17600	7100	5500	3200
ran10x10b	42200	81000	42700	6100
ran17x17	21100	15300	12100	10000

Table 3: Number of iterations needed before convergence for different PDHG enhancements.

7 Linear time implementation of the normalised duality gap for PDLP

One of the key element of PDLP is the adaptive restart. This is why in order to ensure good performance of the algorithm, we need an efficient way of computing the normalised duality gap. Such a method with $\mathcal{O}(N)$ complexity has been introduced in [2] but with the euclidean norm. In this section, we present the linear time computation of the normalised duality gap with the norm $\|z\|_{\omega_n}$ as defined in (42).

Let $z = (x, y) \in Z$ and consider the problem (the factor $\frac{1}{r}$ is irrelevant for the optimisation problem that defines $\rho_r(z)$):

$$\begin{aligned}
&\max_{\hat{z} \in Z : \|\hat{z}\|_\omega \le r} \mathcal{L}(x, \hat{y}) - \mathcal{L}(\hat{x}, y) \\
&= \max_{\hat{z} \in Z : \|\hat{z}\|_\omega \le r} c^\intercal x - \hat{y}^\intercal K x + q^\intercal \hat{y} - (c^\intercal \hat{x} - y^\intercal K \hat{x} + q^\intercal y) \\
&= \max_{\hat{z} \in Z : \|\hat{z}\|_\omega \le r} - \begin{pmatrix} c - K^\intercal y \\ Kx - q \end{pmatrix}^\intercal \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} + (c^\intercal x - q^\intercal y) \\
&= - \min_{\hat{z} \in Z : \|\hat{z}\|_\omega \le r} \begin{pmatrix} c - K^\intercal y \\ Kx - q \end{pmatrix}^\intercal \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} - (c^\intercal x - q^\intercal y)\,.
\end{aligned} \tag{43}$$

Thus, to compute the normalised duality gap one needs to solve:

$$\operatorname*{argmin}_{\hat{z} \in \mathbb{R}^{n+m_1+m_2} : l' \le \hat{z} \le u' \|\hat{z}\|_\omega \le r} g^\intercal \hat{z} \tag{44}$$

with $g = \begin{pmatrix} \nabla_x \mathcal{L}(x, y) \\ -\nabla_y \mathcal{L}(x, y) \end{pmatrix}$, and l' (and u') are the combinations of the lower bound (and upper bound) of the primal space and dual space (in particular $l'_{1:n} = l$). By duality theory and the KKT conditions, there exists $\lambda \in [0, \infty]$ such that (44) is equivalent to:

$$\begin{aligned}
&\operatorname*{argmin}_{\hat{z} \in \mathbb{R}^{n+m_1+m_2} : l' \le \hat{z} \le u'} g^\intercal \hat{z} + \frac{\lambda}{2} \|\hat{z} - z\|_\omega^2 \\
&= \operatorname*{argmin}_{\hat{z} \in \mathbb{R}^{n+m_1+m_2} : l' \le \hat{z} \le u'} g^\intercal \hat{z} + \frac{\lambda}{2} \left(\omega \|\hat{x} - x\|_2^2 + \frac{1}{\omega} \|\hat{y} - y\|_2^2 \right) \\
&= \operatorname*{argmin}_{\hat{z} \in \mathbb{R}^{n+m_1+m_2} : l' \le \hat{z} \le u'} g_{1/\omega}^\intercal \hat{z}_\omega + \frac{\lambda}{2} \|\hat{z}_\omega - z_\omega\|_2^2 \\
&= \operatorname*{argmin}_{\hat{z} \in \mathbb{R}^{n+m_1+m_2} : l' \le \hat{z} \le u'} \|\hat{z}_\omega - (z_\omega - \lambda g_{1/\omega})\|_2^2 \\
&= \operatorname*{argmin}_{\hat{z}_\omega \in \mathbb{R}^{n+m_1+m_2} : l'_\omega \le \hat{z}_\omega \le u'_\omega} \|\hat{z}_\omega - (z_\omega - \lambda g_{1/\omega})\|_2^2
\end{aligned} \tag{45}$$

where we have introduced for $\xi > 0$ the notation $z_\xi = \begin{pmatrix} \sqrt{\xi} z_{1:n} \\ \frac{1}{\sqrt{\xi}} z_{n+1:m_1+m_2} \end{pmatrix}$.

Thus, the optimal value of (44) is obtained for

$$\begin{aligned}
\hat{z}(\lambda) &= \mathbf{proj}_{\{l'_\omega \le z \le u'_\omega\}} (z_\omega - \lambda g_{1/\omega}) \\
&= \min\{\max\{z_\omega - \lambda g_{1/\omega}, l'_\omega\}, u'_\omega\} \\
&= \min\{\max\{z - \lambda g_{1/\omega^2}, l'\}, u'\}
\end{aligned} \tag{46}$$

for some $\lambda \in [0, \infty]$ and where by our previous definition, $g_{1/\omega^2} = \begin{pmatrix} 1/\omega \; g_{1:n} \\ \omega \; g_{n+1:m_1+m_2} \end{pmatrix}$.
In the rest of the argument we will write $g^\omega := g_{1/\omega^2}$ for simplicity.
Thus, (44) reduces to the univariate problem:

$$\max_{\lambda \in [0, \infty]} \; : \; \|\hat{z}(\lambda) - z\|_\omega \le r. \tag{47}$$

Note that if g has some zero coordinates for an index $i \in \{1, .., n + m_1 + m_2\}$, say $g_i = 0$, then we can remove this dimension of the problem as the value of the problem (44) would be independent of the ith coordinates of $\hat{z}$. Thus we can assume without loss of generality that $g_i^\omega \neq 0$ for all indices $i \in \{1, .., n + m_1 + m_2\}$. Moreover, for each index $i \in \{1, ..., n + m_1 + m_2\}$, only l'_i or u'_i will be constraining in problem (46). Indeed, $z \in X \times Y$, so we have $l' \leq z \leq u'$ and thus if $g_i^\omega > 0$, then $z_i - \lambda g_i^\omega < u'_i$ for all $\lambda \in [0, \infty]$ and similarly if $g_i^\omega < 0$, then $z_i - \lambda g_i^\omega > l'_i$ for all $\lambda \in [0, \infty]$. Then if we let:

$$\begin{aligned} L_i &= \begin{cases} u'_i & \text{if} \quad g_i^\omega < 0 \\ l'_i & \text{if} \quad g_i^\omega > 0, \end{cases} \\ \hat{\lambda}_i &= \frac{z_i - L_i}{g_i} \\ \omega_i &= \begin{cases} \omega & \text{if } i \in \{1, ..., n\} \\ \frac{1}{\omega} & \text{if } i \in \{n+1, ..., m_1 + m_2\}, \end{cases} \end{aligned}$$

we have:

$$\begin{aligned} \|\hat{z}(\lambda) - z\|_\omega &= \sum_{i:\hat{z}(\lambda_i) = l'_i} \omega_i (z_i - l'_i)^2 + \sum_{i:\hat{z}(\lambda_i) = u'_i} \omega_i (z_i - u'_i)^2 + \lambda^2 \sum_{i:\hat{z}(\lambda_i) = z_i - \lambda g_i^\omega} \omega_i (g_i^\omega)^2 \\ &= \sum_{i:\hat{\lambda}_i \leq \lambda} \omega_i (z_i - L_i)^2 + \lambda^2 \sum_{i:\hat{\lambda}_i > \lambda} \omega_i (g_i^\omega)^2. \end{aligned} \tag{48}$$

Now, note that the function $\lambda \to \|\hat{z}(\lambda) - z\|_\omega^2$ is non decreasing and thus, in order to solve problem (47) we need to find λ such that: $\|\hat{z}(\lambda) - z\|_\omega^2 = r^2$, i.e.

$$\sum_{i:\hat{\lambda}_i \leq \lambda} \omega_i (z_i - L_i)^2 + \lambda^2 \sum_{i:\hat{\lambda}_i > \lambda} \omega_i (g_i^\omega)^2 = r^2.$$

To this end, we need to attribute each coordinates i to the right sum: consider λ_{med}, the medians of the $\hat{\lambda}_i$. Then if $\|\hat{z}(\lambda) - z\|_\omega^2 < r^2$, it means that the solution λ is greater than λ_{med}, and thus we know that all the coordinates i such that $\lambda_i \leq \lambda_{\text{med}}$ will be attributed in the left sum $\sum_{i:\hat{\lambda}_i \leq \lambda} \omega_i (z_i - L_i)^2$. Conversely if λ is lesser than λ_{med}, all the coordinates i such that $\lambda_i > \lambda_{\text{med}}$ will be attributed in the right sum $\sum_{i:\hat{\lambda}_i > \lambda} \omega_i (g_i^\omega)^2$. Repeating the process, as each step divides by two the number of coordinates left to assign, at some point we have assigned all of them and we find some $\hat{\lambda}_{\text{med}}$ such that:

$$\begin{aligned} &\sum_{i:\hat{\lambda}_i \leq \lambda} \omega_i (z_i - L_i)^2 + \lambda^2 \sum_{i:\hat{\lambda}_i > \lambda} \omega_i (g_i^\omega)^2 = r^2 \\ \Leftrightarrow &\sum_{i:\hat{\lambda}_i \leq \hat{\lambda}_{\text{med}}} \omega_i (z_i - L_i)^2 + \lambda^2 \sum_{i:\hat{\lambda}_i > \hat{\lambda}_{\text{med}}} \omega_i (g_i^\omega)^2 = r^2 \\ \Leftrightarrow &\lambda = \sqrt{\frac{r^2 - \sum_{i:\hat{\lambda}_i \leq \hat{\lambda}_{\text{med}}} \omega_i (z_i - L_i)^2}{\sum_{i:\hat{\lambda}_i > \hat{\lambda}_{\text{med}}} \omega_i (g_i^\omega)^2}}. \end{aligned} \tag{49}$$

As the median of a set of N elements can be found in $\mathcal{O}(N)$ time and we divide the number of coordinates we consider by two at each steps, the maximum total time taken by the algorithm is $\mathcal{O}(\sum_{i=0}^{\infty} \frac{n+m_1+m_2}{2^i}) = \mathcal{O}(n + m_1 + m_2)$ and the algorithm is linear in time. A more formal pseudo code of this algorithm can be found in Appendix F of [2] and the corresponding python code is available in the file `trustregionutil.py` from our python implementation of PDLP https://github.com/CorentinTissier/PDLP.

8 Conclusion

Although PDLP doesn't offer solid theoretical foundations, all the enhancement that it provides over the baseline PDHG are strongly motivated by both theory and numerical tests. As expected it improves a lot the convergence speed of PDHG even without including the presolve and diagonal preconditioning. Further experimentations with bigger LP instances would be interesting to test the limits of this method.

References

[1] David Applegate, Mateo Díaz, Oliver Hinder, Haihao Lu, Miles Lubin, Brendan O'Donoghue, and Warren Schudy. Practical large-scale linear programming using primal-dual hybrid gradient. *Advances in Neural Information Processing Systems*, 34:20243–20257, 2021.

[2] David Applegate, Oliver Hinder, Haihao Lu, and Miles Lubin. Faster first-order primal-dual methods for linear programming using restarts and sharpness. *Mathematical Programming*, pages 1–52, 2022.

[3] Heinz H. Bauschke and Patrick L. Combettes. *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, 2017.

[4] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40:120–145, 2011.

[5] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal–dual algorithm. *Mathematical Programming*, 159(1-2):253–287, 2016.

[6] Tom Goldstein, Min Li, Xiaoming Yuan, Ernie Esser, and Richard Baraniuk. Adaptive primal-dual hybrid gradient methods for saddle-point problems. *arXiv preprint arXiv:1305.0546*, 2013.