Primal-dual forward-reflected splitting with linesearch for distributed min-max problems under local Lipschitz smoothness

Felipe Atenas[1] Minh N. Dao[2] Matthew K. Tam[3] David Torregrosa-Belén[4]

Abstract

We consider a network of finitely many agents aiming to cooperatively solve a convex-concave min-max problem combining nonsmooth terms with a smooth saddle function whose partial gradients are only locally Lipschitz continuous. Without requiring prior knowledge of the local Lipschitz moduli of the partial gradients, we propose a distributed proximal-reflected-gradient algorithm which incorporates a distributed backtracking procedure to adjust the stepsizes at each iteration. The algorithm is derived by reformulating the min-max problem as a composite monotone inclusion involving the sum of a maximally monotone operator, a linearly composed maximally monotone operator, and a single-valued locally Lipschitz continuous monotone operator. To solve this inclusion, we introduce a novel primal-dual forward-reflected splitting method with linesearch and establish its convergence using a Lyapunov-based analysis. In addition, when specialised to the min-max formulation, we derive an $\mathcal{O}(1/N)$ ergodic convergence rate for a consensus-penalised gap function. Finally, applying this splitting framework with mixing matrices yields our distributed method for consensus min-max optimisation.

Keywords distributed consensus optimisation, min-max problems, proximal algorithm, primal-dual splitting algorithm, backtracking linesearch

MSC2020 68W15, 90C25, 90C47, 47N10, 65K05

1 Introduction

In this note, we devise a first-order distributed algorithm for the following *min-max problem.*

Problem 1.1. *For all $i \in \{1, 2, \ldots, n\}$, let the functions $f_i : \mathbb{R}^p \to (-\infty, +\infty]$ and $g_i : \mathbb{R}^q \to (-\infty, +\infty]$ be proper lower semicontinuous convex functions, and $\phi_i : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ be differentiable functions whose partial gradients $\nabla_x \phi_i(\cdot, \cdot)$ and $\nabla_y \phi_i(\cdot, \cdot)$ are locally Lipschitz continuous on $\overline{\operatorname{dom}} f_i \times \overline{\operatorname{dom}} g_i$, and such that $\phi_i(\cdot, y)$ is convex on $\operatorname{dom} f_i$ for every $y \in \operatorname{dom} g_i$, and $\phi_i(x, \cdot)$ is concave on $\operatorname{dom} g_i$ for every $x \in \operatorname{dom} f_i$. Find a solution of the min-max problem*

$$\min_{x \in \mathbb{R}^p} \max_{y \in \mathbb{R}^q} \Psi(x, y) := \sum_{i=1}^{n} \big(f_i(x) + \phi_i(x, y) - g_i(y) \big), \tag{1}$$

where solutions to (1) are understood in the sense of saddle-points. *Specifically, a point (x^*, y^*) is said to be a saddle point of (1) if*

$$\Psi(x^*, y) \leqslant \Psi(x^*, y^*) \leqslant \Psi(x, y^*), \quad \forall x \in \mathbb{R}^p, \forall y \in \mathbb{R}^q.$$

[1]Centro de Modelamiento Matemático (CNRS IRL2807), Universidad de Chile, Santiago, Chile. Email: fatenas@cmm.uchile.cl

[2]School of Science, RMIT University, Melbourne, Australia. Email: minh.dao@rmit.edu.au

[3]School of Mathematics & Statistics, The University of Melbourne, Australia. Email: matthew.tam@unimelb.edu.au

[4]Department of Mathematics, University of Alicante, Alicante, Spain. Email: david.torregrosa@ua.es

For further details on min-max problems and sufficient conditions for the existence of saddle-points, we refer the interested reader to [25].

Works focused on designing decentralised schemes for saddle-point problems often address particular instances of Problem 1.1 or require additional assumptions. Indeed, the authors in [15,20] make use of strong monotonicity assumptions, while in [7,16,26] the functions f_i and g_i are restricted to indicator functions of compact convex sets. More recently, a distributed algorithm for min-max problems in the form of (1) was proposed by Malitsky and Tam [19] under the more restrictive assumption of the partial gradients of the functions ϕ_i being globally Lipschitz continuous. Following this work, we assume that a network of n agents works cooperatively to solve Problem 1.1 under the setup: (i) each agent i computes the gradients of ϕ_i along with the proximity operators of f_i and g_i, and (ii) agents exchange information via their immediate neighbours in the network. Departing from [19], our primary theoretical innovation lies in addressing the lack of global Lipschitz continuity in the gradients of ϕ_i. Specifically, we assume: (iii) there exists no knowledge of the (local) Lipschitz constants of the gradients $\nabla_x \phi_i(\cdot,\cdot)$ and $\nabla_y \phi_i(\cdot,\cdot)$. Problems satisfying conditions (i)-(iii) arise, for instance, in decentralised approaches to distributionally robust optimisation [22,30] (see Section 3 below for a concrete example illustrating this).

Algorithms belonging to the PG-EXTRA family [28,29] represent some of the most prominent schemes in distributed optimisation. It is well established that PG-EXTRA can be derived as a specific instance of a primal–dual splitting algorithm—namely, the Condat–Vũ algorithm [12,33]—applied to an appropriate monotone operator inclusion; see, e.g., [19,27]. In this fashion, we derive our distributed algorithm by first reformulating the problem as a composite monotone inclusion problem of the following form.

Problem 1.2. *Let $\mathcal{H}$ and $\mathcal{G}$ be finite-dimensional real Hilbert spaces. Consider the inclusion problem given by*

$$\text{find } z \in \mathcal{H} \text{ such that } 0 \in \big(A + B\big)(z) + K^* C(Kz), \tag{2}$$

where $A : \mathcal{H} \rightrightarrows \mathcal{H}$, $C : \mathcal{G} \rightrightarrows \mathcal{G}$ are maximally monotone operators, $B : \mathcal{H} \to \mathcal{H}$ is monotone on $\operatorname{dom} A$ and locally Lipschitz continuous on $\overline{\operatorname{dom}} A$, $K : \mathcal{H} \to \mathcal{G}$ is a linear operator, and $K^* : \mathcal{G} \to \mathcal{H}$ its adjoint operator.

It is well-known that dealing with monotone Lipschitz continuous operators requires considering different numerical schemes than the ones used for cocoercive operators; see, e.g., [1,18,24,32]. In addition, when the operator fails to be globally Lipschitz continuous or the Lipschitz constant cannot be accurately estimated, further efforts are required to select appropriate stepsizes; see, e.g., [8,9,18]. Different splitting algorithms with linesearch have been proposed to handle monotone inclusions given by the sum of a maximally monotone operator and a locally Lipschitz continuous monotone operator; see, e.g., [8,9,18,32]. These methods can solve Problem 1.2 by reformulating (2) as the two-operator primal-dual inclusion

$$\text{find } \begin{bmatrix} u \\ z \end{bmatrix} \in \mathcal{G} \times \mathcal{H} \text{ such that } 0 \in (\mathcal{A} + \mathcal{B}) \begin{bmatrix} u \\ z \end{bmatrix},$$

where

$$\mathcal{A} := \begin{bmatrix} C^{-1} & 0 \\ 0 & A \end{bmatrix} \quad \text{and} \quad \mathcal{B} := \begin{bmatrix} 0 & -K \\ K^* & B \end{bmatrix}.$$

Specifically, they resort to forward-backward schemes that perform direct evaluations of the operator $\mathcal{B}$ and the resolvent of the set-valued operator $\mathcal{A}$. To illustrate this, we consider the *forward-reflected-backward method with linesearch* in [18] that defines the iterates as

$$\omega^{k+1} = J_{\gamma_k \mathcal{A}}\big(\omega^k - \gamma_k \mathcal{B}(\omega^k) - \gamma_{k-1}(\mathcal{B}(\omega^k) - \mathcal{B}(\omega^{k-1})\big),$$

with $\omega^k = (u^k, z^k) \in \mathcal{G} \times \mathcal{H}$ for all $k \geqslant -1$, and where the stepsize γ_k is taken as the largest number in $\{\lambda_{k-1}, \rho\lambda_{k-1}, \rho^2\lambda_{k-2}, \ldots\}$ such that

$$\gamma_k \|\mathcal{B}(\omega^{k+1}) - \mathcal{B}(\omega^k)\| \leqslant \frac{\beta}{2}\|\omega^{k+1} - \omega^k\|, \tag{3}$$

for $\rho, \beta \in (0, 1)$. Since the linesearch condition (3) requires the computation of $\mathcal{B}(\omega^{k+1})$, every linesearch trial evaluates B, K, K^* and the resolvents of both A and C^{-1}. In this manner, an accepted linesearch iteration uses two evaluations of B, two of K and two of K^*.

In this work, we develop a new *primal-dual forward-reflected algorithm with linesearch* (PDFRL) tailored to Problem 1.2 that alleviates the computational burden of the linesearch with respect to the aforementioned approaches. Specifically, PDFRL generates a pair of sequences converging to a primal-dual solution of Problem 1.2 by evaluating the resolvents of A and C^{-1} separately. The lack of cocoercivity of B is resolved by incorporating reflected-type terms of this operator. Notably, PDFLR's linesearch only involves the sequence of primal variables generated by the algorithm, thereby it does not require additional evaluations of the resolvent of C^{-1}. Furthermore, only one call to K and K^* are needed per iteration, independently of the number of linesearch trials performed. This, in turn, will permit us to derive a distributed algorithm with a single round of communication per iteration by specialising PDFRL to Problem 1.1.

It is worth noting that when the monotone operator B is assumed to be globally Lipschitz, a fixed stepsize can be taken for all the iterations of our method, and no linesearch is needed. In such a case, our scheme specialises to the *primal-dual twice reflected* algorithm in [19] with extrapolation performed on the dual variable. As explained in [19] (see also Remark 5.1 below), this method admits a representation as an instance of the forward-backward-reflected splitting [18] under a different metric, which is used to establish its convergence. While this suggests that it may be possible to analyze the convergence of PDFRL with linesearch using a similar (variable) metric reformulation, it is unclear whether this would lead to a simpler presentation of the result. We therefore establish convergence to a primal-dual solution using a direct Lyapunov-based analysis. This also allows us to obtain ergodic convergence rates of $\mathcal{O}(1/N)$, where N is the number of iterations of the algorithm, for a consensus-penalised gap function associated to Problem 1.1 (see (41) below).

We derive our proposed *distributed proximal-reflected-gradient method with linesearch* (DPRGL) for solving Problem 1.1, by appropriately applying PDFRL using mixing matrices (see Definition 6.1 below). In particular, devising DPRGL required tailoring the linesearch step in PDFRL to agent-wise operations. Recently, [3] and [11] have proposed distributed linesearch mechanisms for proximal-gradient methods for minimisation problems. Here, the framework developed in [3] is extended to the min-max setting to equip our algorithm with a backtracking linesearch that relies on a global scalar reduction across all agents.

The remainder of this work is organised as follows. Section 2 introduces some preliminary notation. Section 3 presents examples of applications of Problem 1.1 in two regression models. In Section 4, we present the monotone inclusion reformulation of Problem 1.1, which will be crucial for the design of our distributed algorithm. Subsequently, in Section 5, we develop a new primal-dual forward reflected algorithm with linesearch for solving Problem 1.2 and analyse its convergence. This method is leveraged to obtain a distributed algorithm for Problem 1.1 in Section 6. Finally, conclusions and future lines of research are discussed in Section 7.

2 Notation and definitions

Throughout this work, we use $\mathcal{H}$ and $\mathcal{G}$ to denote finite-dimensional real *Hilbert spaces*. Unless otherwise stated, and to simplify the notation, we will employ $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ to denote the *inner product* and the *induced norm*, respectively, of any space. In most of the cases, $\mathcal{H}$ will just stand

for the Euclidean space, which is denoted as $\mathbb{R}^p$ for a given dimension p, with norm given by the standard dot product. The set of natural numbers is $\mathbb{N} := \{0, 1, 2, \ldots\}$. For a sequence in $\mathcal{H}$ we use (z^k) instead of the more lengthy $(z^k)_{k\in\mathbb{N}}$. We use $\to$ for denoting convergence of a sequence, $\operatorname{ri} D$ for the *relative interior* of a set D, and establish $1/0 = +\infty$ by convention.

When addressing Problem 1.1, it will be convenient to decouple the variables and work on the product space

$$\mathcal{H}^n := \big\{\mathbf{x} = (x_1, x_2, \ldots, x_n) \,:\, x_1, x_2, \ldots, x_n \in \mathcal{H}\big\},$$

endowed with the inner product

$$\langle \mathbf{x}, \mathbf{y}\rangle = \langle x_1, y_1\rangle + \langle x_2, y_2\rangle + \cdots + \langle x_n, y_n\rangle, \quad \mathbf{x}, \mathbf{y} \in \mathcal{H}^n.$$

We use boldface for vectors in product spaces. The *diagonal subspace* of $\mathcal{H}^n$ is defined as

$$\mathcal{D}_{\mathcal{H}^n} := \big\{\mathbf{x} \in \mathcal{H}^n \,:\, x_1 = x_2 = \ldots = x_n\big\}.$$

If $\mathcal{H} = \mathbb{R}^p$, then we will identify $\mathcal{H}^n$ with the space $\mathbb{R}^{n\times p}$, so that $\mathbf{x} = (x_1, x_2, \ldots, x_n)^\top$ and the induced norm coincides with the Frobenius norm. We will often consider a matrix $V \in \mathbb{R}^{n\times n}$ as a linear operator from $\mathbb{R}^{n\times p}$ to $\mathbb{R}^{n\times p}$. Of special importance will be those matrices that are said to *enforce consensus* on the variables of $\mathbb{R}^{n\times p}$, namely, verify $\ker V = \mathcal{D}_{\mathbb{R}^{n\times p}}$. Equivalently, $V\mathbf{x} = 0$ if and only if the n rows of $\mathbf{x} \in \mathbb{R}^{n\times p}$ coincide. Observe that this property is independent of $p \in \mathbb{N}\backslash\{0\}$. Indeed, one can simply say that $V \in \mathbb{R}^{n\times n}$ enforces consensus if, when viewed as a linear operator on $\mathbb{R}^n$, it satisfies $\ker V = \mathcal{D}_{\mathbb{R}^n} = \operatorname{span}\{\mathbb{1}_n\}$ with $\mathbb{1}_n \in \mathbb{R}^n$ being the vector of all ones.

Let $f : \mathcal{H} \to (-\infty, +\infty]$ be a function. The *domain* of f is $\operatorname{dom} f := \{x \in \mathcal{H} \,:\, f(x) < +\infty\}$. The function f is said to be *proper* if $\operatorname{dom} f \neq \varnothing$, and *lower semicontinuous* if, for all $\bar{x} \in \mathcal{H}$, it holds $\liminf_{x\to\bar{x}} f(x) \geqslant f(\bar{x})$.

We use $A : \mathcal{H} \rightrightarrows \mathcal{H}$ for denoting a *set-valued operator* in $\mathcal{H}$, in opposition to $B : \mathcal{H} \to \mathcal{H}$, that stands for a *single-valued operator*. The *domain* and *graph* of A are given by $\operatorname{dom} A := \{x \in \mathcal{H} \,:\, A(x) \neq \varnothing\}$ and $\operatorname{gra} A := \{(x, u) \in \mathcal{H} \times \mathcal{H} \,:\, u \in A(x)\}$, respectively. In particular, observe that by writing $B : \mathcal{H} \to \mathcal{H}$ we readily imply $\operatorname{dom} B = \mathcal{H}$. The closure of the domain of an operator A is denoted by $\overline{\operatorname{dom}} A$. The *inverse operator* of A is the operator A^{-1} such that $\operatorname{gra} A^{-1} = \{(u, x) \in \mathcal{H} \times \mathcal{H} \,:\, u \in A(x)\}$. The *identity operator* is denoted by Id. Given a linear operator $K : \mathcal{H} \to \mathcal{G}$, we use $K^* : \mathcal{G} \to \mathcal{H}$ to denote its *adjoint*, which is the only linear operator such that $\langle Kx, y\rangle = \langle x, K^*y\rangle$, for all $(x, y) \in \mathcal{H} \times \mathcal{G}$. We use $\|K\|$ for the (2-)norm of K.

Let $U \subseteq \mathcal{H}$ be a nonempty set. An operator $B : \mathcal{H} \to \mathcal{H}$ is *Lipschitz continuous* on U if there exists a constant $L \geqslant 0$ such that

$$\|B(x) - B(y)\| \leqslant L\|x - y\|, \quad \text{for all } x, y \in U.$$

We say that B is *locally Lipschitz continuous* if it is Lipschitz continuous on some neighbourhood of every point in $\mathcal{H}$.

A (possibly) set-valued operator $A : \mathcal{H} \rightrightarrows \mathcal{H}$ is said to be *monotone* if

$$\langle x - y, u - v\rangle \geqslant 0, \quad \forall (x, u), (y, v) \in \operatorname{gra} A.$$

Furthermore, if there exists no monotone operator $C : \mathcal{H} \rightrightarrows \mathcal{H}$ such that $\operatorname{gra} C$ properly contains $\operatorname{gra} A$, we say that A is *maximally monotone*. A well-known example of maximally monotone operator is the *subdifferential* of a proper lsc convex function $f : \mathcal{H} \to (-\infty, +\infty]$, which is the set-valued operator defined as

$$\partial f \colon \mathcal{H} \rightrightarrows \mathcal{H} \colon x \mapsto \{u \in \mathcal{H} \,:\, f(x) + \langle u, y - x\rangle \leqslant f(y),\ \forall y \in \mathcal{H}\}.$$

If $f = \iota_U$ is the *indicator function* of a nonempty closed convex set $U \subseteq \mathcal{H}$, its subdifferential is given by the *normal cone* of U, that is,

$$\partial_f(x) = N_U(x) := \begin{cases} \{u \in \mathcal{H} \,:\, \sup \langle u, U - x\rangle \leqslant 0\}, & \text{if } x \in U; \\ \varnothing, & \text{otherwise.} \end{cases}$$

Numerical algorithms devoted to addressing problems involving monotone operators usually make use of the so-called *resolvent operator*. The resolvent of a maximally monotone operator $A : \mathcal{H} \rightrightarrows \mathcal{H}$ is the single-valued operator $J_A : \mathcal{H} \to \mathcal{H}$ given by

$$J_A = (\mathrm{Id} + A)^{-1}.$$

While resolvent operators are not always easily calculated, closed formulas are known for a wide range of useful operators; see, e.g., [5, Chapter 23]. In particular, when $A = \partial f$ with f proper lsc and convex, the resolvent reduces to the *proximity operator* of f, namely,

$$J_{\partial f}(x) = \mathrm{prox}_f(x) := \arg\min_z \big\{ f(z) + \frac{1}{2}\|z - x\|^2 \big\}.$$

Let $A_1 : \mathcal{H} \rightrightarrows \mathcal{H}$ and $A_2 : \mathcal{G} \rightrightarrows \mathcal{G}$ be two maximally monotone operators. The Cartesian product $A := A_1 \times A_2 : \mathcal{H} \times \mathcal{G} \rightrightarrows \mathcal{H} \times \mathcal{G}$, defined as $A(\mathbf{x}) = (A_1(x_1), A_2(x_2))$, where $\mathbf{x} = (x_1, x_2) \in \mathcal{H} \times \mathcal{G}$, is also a maximally monotone operator [5, Proposition 20.23]. Furthermore, its resolvent is given component-wise as $J_A = (J_{A_1}, J_{A_2})$, see [5, Proposition 23.18].

3 Application to regression models

The study of decentralised algorithms for Problem 1.1 is motivated by two regression models.

3.1 Distributionally robust Poisson regression

The first of our example consists of a distributionally robust optimisation approach to a Poisson regression problem.

Example 3.1. *Suppose n agents cooperate to estimate a parameter vector $x \in \mathbb{R}^p$ for a Poisson regression model across q operational environments. More precisely, each agent $i \in \{1, 2, \ldots, n\}$ collects feature vectors $a_{i,j} \in \mathbb{R}^p$ and event counts $b_{i,j} \in \mathbb{N}$ under environment $j \in \{1, 2, \ldots, q\}$. Maximum-likelihood estimation leads to maximising the probability terms*

$$x \mapsto \frac{\big(\exp(\langle a_{i,j}, x\rangle)\big)^{b_{i,j}} \exp\big(-\exp(\langle a_{i,j}, x\rangle)\big)}{b_{i,j}!}, \quad i \in \{1, 2, \ldots, n\},\ j \in \{1, 2, \ldots, q\}.$$

This is in turn equivalent to minimising the negative log-likelihood, which leads to the following family of convex local loss functions

$$\ell_{i,j}(x) = \exp(\langle a_{i,j}, x\rangle) - b_{i,j}\langle a_{i,j}, x\rangle, \quad i \in \{1, 2, \ldots, n\},\ j \in \{1, 2, \ldots, q\}.$$

To ensure robustness against unknown environment variations, the maximum-likelihood estimation can be framed into a distributionally robust optimisation framework, leading to the problem

$$\min_{x \in \mathbb{R}^p} \max_{y \in \Delta^q} \ \lambda\|x\|_1 + \sum_{i=1}^{n} \bigg(\sum_{j=1}^{q} y_j \ell_{i,j}(x) - \frac{\mu}{12n} \sum_{j=1}^{q} y_j^4 \bigg), \tag{4}$$

where Δ^q is the probability simplex of dimension q and the regulariser $\lambda\|\cdot\|_1$, with $\lambda > 0$, aims to promote sparsity on the parameter vector. Here, the dual variable $y = (y_1, \ldots, y_q)^T \in \Delta^q$

represents an adversarial probability distribution over the q environmental conditions. Finally, the power penalty term $-\frac{\mu}{12n}\sum_{j=1}^{q} y_j^4$ derives from the Cressie–Read family of divergences [13] and is used to prevent adversarial collapse onto a single worst-case environment; see, e.g., [21, 30].

It is easy to see that (4) is in the form of (1) by setting $f_i(x) := \frac{\lambda}{n}\|x\|_1$, $\phi_i(x,y) := \sum_{j=1}^{q}\big(y_j\ell_{i,j}(x) - \frac{\mu}{12n}y_j^4\big)$, and $g_i(y) := \frac{1}{n}\iota_{\Delta^q}(y)$. Observe that $\phi_i(\cdot,y)$ is convex for all $y \in [0,+\infty)^q$, $\phi_i(x,\cdot)$ is concave for all $x \in \mathbb{R}^p$, ϕ_i is smooth on the entire space $\mathbb{R}^p \times \mathbb{R}^q$, and both partial gradients $\nabla_x\phi_i(\cdot,\cdot)$ and $\nabla_y\phi_i(\cdot,\cdot)$ are locally Lipschitz continuous, but not globally.

Further examples illustrating the scope of Problem 1.1 can be obtained by considering different loss functions. For instance, one can frame the multiple image deconvolution problem from corrupted Poisson data (see, e.g., [3, 6]) within a distributionally robust setting.

3.2 Distributed regression with high-order powers

The second example involves linear regression for data contaminated with light-tailed noise. Consider the linear model $b = \langle a, x\rangle + \varepsilon$, where the noise vector ε distributes according to a density known a priori to have light tails. In this setting, measuring the misfit with power of higher order may be statistically more efficient than least-squares, as the former can control larger residuals more aggressively.

Example 3.2. *Let n agents in a network cooperate to estimate a sparse parameter vector $x \in \mathbb{R}^p$ of a linear model. Each agent $i \in \{1,2,\ldots,n\}$ privately holds a design matrix $A_i \in \mathbb{R}^{m_i\times p}$, with rows $a_{i,1}^\top,\ldots,a_{i,m_i}^\top$, and a vector of observations $b_i \in \mathbb{R}^{m_i}$ whose noise is known to be light-tailed. In addition, the estimate is required to satisfy a system of q linear inequality constraints encoding prior knowledge on the model, whose data are split among the agents. Given $\lambda > 0$, the optimisation problem can thus be formulated as*

$$\min_{x\in\mathbb{R}^p}\ \lambda\|x\|_1 + \sum_{i=1}^{n}\ell_i(x) \quad s.t. \quad \sum_{i=1}^{n}(C_ix - d_i) \leqslant 0, \tag{5}$$

where, for all $i \in \{1,2,\ldots,n\}$, $C_i \in \mathbb{R}^{q\times p}$, $d_i \in \mathbb{R}^q$ and

$$\ell_i(x) := \frac{1}{4}\|A_ix - b_i\|_4^4 = \frac{1}{4}\sum_{j=1}^{m_i}\big(\langle a_{i,j}, x\rangle - b_{i,j}\big)^4.$$

Dualising the coupling constraint with a multiplier $y \in \mathbb{R}_+^q$ yields the Lagrangian formulation

$$\min_{x\in\mathbb{R}^p}\max_{y\in\mathbb{R}_+^q}\ \lambda\|x\|_1 + \sum_{i=1}^{n}\Big(\ell_i(x) + \big\langle y, C_ix - d_i\big\rangle\Big). \tag{6}$$

Problem (6) is in the form of (1) by setting $f_i(x) := \frac{\lambda}{n}\|x\|_1$, $g_i(y) := \frac{1}{n}\iota_{\mathbb{R}_+^q}(y)$ and $\phi_i(x,y) := \ell_i(x) + \langle y, C_ix - d_i\rangle$. Note that ϕ_i is convex-concave, and the partial gradients $\nabla_x\phi_i(\cdot,\cdot)$ and $\nabla_y\phi_i(\cdot,\cdot)$ are locally Lipschitz continuous.

4 Reformulation as a monotone inclusion

Under the assumption that Problem 1.1 has at least a solution, we now obtain a reformulation as a monotone inclusion. First, observe that (1) can be alternatively written as

$$\min_{(x_1,\ldots,x_n)\in\mathbb{R}^{p\times n}}\ \max_{(y_1,\ldots,y_n)\in\mathbb{R}^{q\times n}}\ \sum_{i=1}^{n}\big(f_i(x_i) + \phi_i(x_i,y_i) - g_i(y_i)\big) \quad \text{s.t.} \quad \begin{cases} x_1 = x_2 = \ldots = x_n, \\ y_1 = y_2 = \ldots = y_n. \end{cases} \tag{7}$$

For convenience, let $f : \mathbb{R}^{n\times p} \to (-\infty, +\infty]$, $g : \mathbb{R}^{n\times q} \to (-\infty, +\infty]$ and $\phi : \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q} \to \mathbb{R}$ denote the functions

$$f(\mathbf{x}) = \sum_{i=1}^{n} f_i(x_i), \quad g(\mathbf{y}) = \sum_{i=1}^{n} g_i(y_i) \quad \text{and} \quad \phi(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \phi_i(x_i, y_i), \tag{8}$$

where $\mathbf{x} := (x_1, \ldots, x_n)^\top \in \mathbb{R}^{n\times p}$ and $\mathbf{y} := (y_1, \ldots, y_n)^\top \in \mathbb{R}^{n\times q}$. Let $K_1, K_2 \in \mathbb{R}^{n\times n}$ be matrices ensuring consensus on $\mathbb{R}^{n\times p}$ and $\mathbb{R}^{n\times q}$, respectively. Then (7) is equivalent to the problem

$$\min_{\mathbf{x}\in\mathbb{R}^{n\times p}} \max_{\mathbf{y}\in\mathbb{R}^{n\times q}} f(\mathbf{x}) + \iota_{\{0\}}(K_1\mathbf{x}) + \phi(\mathbf{x}, \mathbf{y}) - \iota_{\{0\}}(K_2\mathbf{y}) - g(\mathbf{y}). \tag{9}$$

We will tackle (9) by addressing the monotone inclusion resulting from its first order optimality conditions. Let $\mathbf{z} := (\mathbf{x}, \mathbf{y})$. Let the operators $A : \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q} \rightrightarrows \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$, $B : \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q} \to \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ and the linear operator $K : \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q} \to \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ be

$$A(\mathbf{z}) := \big(\partial f(\mathbf{x}), \partial g(\mathbf{y})\big), \quad B(\mathbf{z}) := \big(\nabla_{\mathbf{x}}\phi(\mathbf{x}, \mathbf{y}), -\nabla_{\mathbf{y}}\phi(\mathbf{x}, \mathbf{y})\big) \quad \text{and} \quad K\mathbf{z} := (K_1\mathbf{x}, K_2\mathbf{y}), \tag{10}$$

where $\partial f = \bigtimes_{i=1}^{n} \partial f_i$ and $\partial g = \bigtimes_{i=1}^{n} \partial g_i$ by [5, Proposition 16.9]. In particular, the operator A is maximally monotone as the Cartesian product of maximally monotone operators, while B is locally Lipschitz continuous on $\overline{\operatorname{dom} A}$ and monotone on $\operatorname{dom} A$ by assumption. Hence, by the subdifferential inclusion [5, Proposition 16.6], a sufficient condition for a vector $\mathbf{z} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ to be a solution of (9) is to solve the monotone inclusion problem consisting in

$$\text{find } \mathbf{z} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q} \text{ such that } 0 \in \big(A + B\big)(\mathbf{z}) + K^* N_{\{0\}}\big(K\mathbf{z}\big). \tag{11}$$

Clearly, the above is in the form of Problem 1.2. If in addition we want to ensure that both (1) and (11) are equivalent, we further need the following subdifferential sum rules to hold:

$$\partial f + K_1^* \circ \partial\iota_{\{0\}} \circ K_1 = \partial\left(f + \iota_{\{0\}} \circ K_1\right), \quad \partial g + K_2^* \circ \partial\iota_{\{0\}} \circ K_2 = \partial\left(g + \iota_{\{0\}} \circ K_2\right). \tag{12}$$

By [5, Theorem 16.47 & Proposition 6.19(x)] for (12) to hold it suffices that $\cap_{i=1}^{n} \operatorname{ri}(\operatorname{dom} f_i) \neq \varnothing$ and $\cap_{i=1}^{n} \operatorname{ri}(\operatorname{dom} g_i) \neq \varnothing$. Observe that the latter trivially hold in Examples 3.1 and 3.2. We summarise all of the above in the following proposition.

Proposition 4.1. *Suppose that Problem 1.1 has at least a solution and that the subdifferential sum rules in (12) hold. Then a vector $\mathbf{z}^* = (\mathbf{x}^*, \mathbf{y}^*) \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ solves (11) if and only if $x^* := x_1^* = \ldots = x_n^* \in \mathbb{R}^p$, $y^* := y_1^* = \ldots = y_n^* \in \mathbb{R}^q$ and (x^*, y^*) is a saddle point of (1).*

Proposition 4.1 motivates the development of numerical schemes for solving Problem 1.2.

5 A new primal-dual forward-reflected splitting algorithm with linesearch

In this section, we propose a new splitting method with linesearch for addressing Problem 1.2. Hence, in what follows we assume that $\mathcal{H}$ and $\mathcal{G}$ are finite-dimensional Hilbert spaces.

The presence of the linear compositions over the operator C makes it convenient to address the resolution of (2) by means of its associated *primal-dual problem*. That is, we search for a pair $(z^*, u^*) \in \mathcal{H} \times \mathcal{G}$ such that

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} \in \begin{bmatrix} C^{-1} & -K \\ K^* & A + B \end{bmatrix} \begin{bmatrix} u^* \\ z^* \end{bmatrix}. \tag{13}$$

We refer to such a pair as a *primal-dual solution* of (2). It is a simple algebraic exercise to check that a pair (z^*, u^*) is a primal-dual solution if and only if z^* is a solution of the (primal) problem (2) and u^* provides a solution of an *Attouch–Théra dual* problem [4].

In order to obtain a primal-dual solution of (2), we develop a splitting algorithm that incorporates a linesearch procedure to account for the lack of knowledge of the (local) Lipschitz modulus of the operator B. Our method does assume knowledge of the norm of the linear operator K. We name this method *primal-dual forward-reflected with linesearch* (PDFRL in short) and present its pseudocode in Algorithm 1. Observe that the resolvent in the dual update uses the stepsize computed in the previous iteration of the algorithm. In principle, the order of Step 1 and Step 2 can be interchanged, however this would require computing an extra resolvent of C^{-1} for every inner iteration of the linesearch, increasing the computational cost of the algorithm.

Algorithm 1: Primal-dual forward-reflected with linesearch (PDFRL) for Problem 1.2

Choose initial points $z^{-1} = z^0 \in \operatorname{dom} A$, $u^0 \in \mathcal{G}$;
Choose parameters $\delta \in (0,1)$ and $\beta \in (0,1)$;
Choose an initial stepsize $\tau_{-1} \in \left(0, \delta \frac{1-\beta}{2\|K\|}\right)$;
for $k = 0, 1, \dots$ **do**

▷ **Step 1** (Dual update). Define

$$u^{k+1} = J_{\tau_{k-1}C^{-1}}\big(u^k + \tau_{k-1} K z^k\big); \tag{14}$$

▷ **Step 2** (Abstract linesearch and primal update). Compute

$$z^{k+1} = J_{\tau_k A}\big(z^k - \tau_k K^* u^{k+1} - \tau_{k-1} K^*(u^{k+1} - u^k) - \tau_k B(z^k) - \tau_{k-1}(B(z^k) - B(z^{k-1}))\big), \tag{15}$$

where $\tau_k \in \left(0, \delta \frac{1-\beta}{2\|K\|}\right)$ satisfies

$$\tau_k \|B(z^{k+1}) - B(z^k)\| \leqslant \frac{\sqrt{\beta}}{2} \|z^{k+1} - z^k\|. \tag{16}$$

Remark 5.1. *We collect some algorithms that can be recovered as special cases or are related to Algorithm 1.*

(i) If B is globally Lipschitz, a fixed stepsize τ can be taken according to the Lipschitz modulus. The update of the algorithm will then read as

$$\begin{cases} u^{k+1} = J_{\tau C^{-1}}\big(u^k + \tau K z^k\big), \\ z^{k+1} = J_{\tau A}\big(z^k - \tau K^*(2u^{k+1} - u^k) - 2\tau B(z^k) + \tau B(z^{k-1})\big). \end{cases} \tag{17}$$

This scheme is the primal-dual twice reflected *algorithm in [19], but with extrapolation performed on the dual variable. Indeed, analogously to the algorithm in [19], iteration (17) can be recovered as an instance of the* forward-reflected-backward method *(FoRB) [18] under a different metric. Namely, (17) is given by the scheme*

$$\omega^{k+1} = J_{M^{-1}G}\big(\omega^k - 2M^{-1}F(\omega^k) + M^{-1}F(\omega^{k-1})\big),$$

with $\omega^k = (z^k, u^k)$ and where the matrix M and the operators G, F are given by

$$M = \begin{bmatrix} \frac{1}{\tau}\operatorname{Id} & K^* \\ K & \frac{1}{\tau}\operatorname{Id} \end{bmatrix}, \quad G = \begin{bmatrix} A & K^* \\ -K & C^{-1} \end{bmatrix} \quad \text{and} \quad F = \begin{bmatrix} B & 0 \\ 0 & 0 \end{bmatrix}.$$

(ii) If in addition $B = 0$, we recover the scheme

$$\begin{cases} u^{k+1} = J_{\tau C^{-1}}\big(u^k + \tau K z^k\big), \\ z^{k+1} = J_{\tau A}\big(z^k - \tau K^*(2u^{k+1} - u^k)\big), \end{cases}$$

which corresponds to an instance of the primal-dual hybrid gradient *method [10].*

(iii) If $K = 0$, then updates (14)-(15) become

$$\begin{cases} u^{k+1} = J_{\tau_{k-1} C^{-1}}\big(u^k\big), \\ z^{k+1} = J_{\tau_k A}\big(z^k - \tau_k B(z^k) - \tau_{k-1}(B(z^k) - B(z^{k-1}))\big). \end{cases} \tag{18}$$

Hence, two independent iterations are performed by the method: a variable stepsize proximal point algorithm *for C^{-1} and the* forward-reflected-backward with linesearch *[18, Algorithm 3.1] for $A + B$.*

(iv) If in addition $A = 0$ and B is globally Lipschitz continuous, the (z^k) iteration in (18) reduces to

$$z^{k+1} = z^k - 2\tau B(z^k) + \tau B(z^{k-1}).$$

This is the so-called optimistic gradient descent ascent *(OGDA) (also known as* optimistic mirror descent*) within the GAN literature [14], which in turn is recovered from the unconstrained version of* Popov's algorithm *[23]*

$$\begin{cases} \bar{z}^{k+1} = \bar{z}^k - \tau B(z^k), \\ z^{k+1} = \bar{z}^{k+1} - \tau B(z^k), \end{cases}$$

under the initialisation $\bar{z}^0 = z^0 + \tau B(z^{-1})$.

5.1 Convergence analysis

In this section, the convergence of Algorithm 1 is established. First, the following lemma ensures that the linesearch in Step 2 is well-defined.

Lemma 5.2. *Suppose that the assumptions of Problem 1.2 hold. Consider the sequences (z^k) and (u^k) generated by Algorithm 1. Then, for all $k \in \mathbb{N}$, there exists $\zeta_k > 0$ such that*

$$\tau\|B(z^{k+1}) - B(z^k)\| \leqslant \frac{\sqrt{\beta}}{2}\|z^{k+1} - z^k\|, \quad \textit{for all } \tau \in (0, \zeta_k).$$

Proof. Let $k \in \mathbb{N}$ fixed. For convenience let us denote

$$z^{k+1}(\tau) := J_{\tau A}\big(z^k - \tau K^* u^{k+1} - \tau_{k-1} K^*(u^{k+1} - u^k) - \tau B(z^k) - \tau_{k-1}(B(z^k) - B(z^{k-1}))\big), \tag{19}$$

with τ taking values in the nonnegative real line. In particular, we have

$$z^{k+1}(0) = z^k - \tau_{k-1} K^*(u^{k+1} - u^k) - \tau_{k-1}(B(z^k) - B(z^{k-1})). \tag{20}$$

By [2, Proposition 3.4], $z^{k+1}(\tau) \to P_{\overline{\operatorname{dom} A}}\big(z^k - \tau_{k-1} K^*(u^{k+1} - u^k) - \tau_{k-1}(B(z^k) - B(z^{k-1}))\big) = P_{\overline{\operatorname{dom} A}}\big(z^{k+1}(0)\big)$ as $\tau \downarrow 0$, where $P_{\overline{\operatorname{dom} A}}$ stands for the projection onto the closure of the domain of A.

Let $U \subseteq \overline{\operatorname{dom}} A$ be a bounded convex set such that it is open with respect to $\overline{\operatorname{dom}} A$ and that contains the pair $\big\{z^k, P_{\overline{\operatorname{dom} A}}\big(z^{k+1}(0)\big)\big\}$. Since we have proved that $z^{k+1}(\tau) \subset \operatorname{dom} A$ converges to $P_{\overline{\operatorname{dom} A}}\big(z^{k+1}(0)\big)$ as $\tau \downarrow 0$, then for all sufficiently small τ we can ensure that $z^{k+1}(\tau)$ belongs to U. Recall now that as a locally Lipschitz continuous operator on a finite dimensional space, B

is Lipschitz continuous on any nonempty compact convex subset, and in particular on $\overline{U} \subseteq \overline{\operatorname{dom} A}$. Hence, there exist constants $L \geqslant 0$ and $\eta_k > 0$ such that

$$\tau\big\|B\big(z^{k+1}(\tau)\big) - B\big(z^k\big)\big\| \leqslant \tau L\big\|z^{k+1}(\tau) - z^k\|, \quad \text{for all } \tau \in (0, \eta_k).$$

To conclude the claim it suffices to take $\zeta_k := \min\{\eta_k, \sqrt{\beta}/(2L)\}$.

□

As mentioned in Remark 5.1(i), Algorithm 1 is closely related to the *primal-dual twice reflected* (PDTR) method in [19]. While the convergence analysis of PDTR draws upon its identification as an instance of the forward-reflected-backward algorithm [18] under a matrix-induced metric, we here make use of a direct Lyapunov-based analysis to establish the convergence of Algorithm 1. Fix a pair $(z, u) \in \mathcal{H} \times \mathcal{G}$, and for all $k \in \mathbb{N}$, we define the energy functions

$$\mathcal{E}_k^{(z,u)} := \frac{1}{2}\big(\|z^k - z\|^2 + \|u^k - u\|^2\big) + \tau_{k-1}\langle K(z^k - z), u^k - u\rangle, \tag{21}$$

$$\mathcal{V}_k^{(z,u)} := \mathcal{E}_k^{(z,u)} - \tau_{k-1}\langle B(z^k) - B(z^{k-1}), z^k - z\rangle + \frac{1}{4}\|z^k - z^{k-1}\|^2. \tag{22}$$

Note that, by Cauchy–Schwarz and Young's inequalities, $\mathcal{E}_k^{(z,u)} \geqslant 0$ whenever $0 < \tau_{k-1}\|K\| \leqslant 1$. In addition, $\mathcal{V}_k^{(z,u)}$ is also nonnegative thanks to the linesearch condition in Algorithm 1. We prove this in the following result.

Lemma 5.3. *Suppose that the assumptions of Problem 1.2 hold. Consider the sequences (z^k) and (u^k) generated by Algorithm 1. Then, for any fixed $(z, u) \in \mathcal{H} \times \mathcal{G}$ and for all $k \in \mathbb{N}$, it holds*

$$\begin{aligned}\mathcal{V}_k^{(z,u)} &\geqslant \mathcal{E}_k^{(z,u)} - \frac{\beta}{4}\|z^k - z\|^2 \\ &\geqslant \frac{1}{2}\left(1 - \frac{\beta}{2} - \tau_{k-1}\|K\|\right)\|z^k - z\|^2 + \frac{1}{2}\left(1 - \tau_{k-1}\|K\|\right)\|u^k - u\|^2 \geqslant 0.\end{aligned} \tag{23}$$

Proof. For the first inequality, observe that $z^0 = z^{-1}$, by initialisation of Algorithm 1. This implies $\mathcal{V}_0^{(z,u)} = \mathcal{E}_0^{(z,u)}$, and the inequality trivially holds for $k = 0$. For $k \in \mathbb{N}\backslash\{0\}$, using Cauchy–Schwarz and (16) at iteration $k-1$ followed by Young's inequality, we get

$$\begin{aligned}-\tau_{k-1}\langle B(z^k) - B(z^{k-1}), z^k - z\rangle &\geqslant -\frac{\sqrt{\beta}}{2}\|z^k - z^{k-1}\|\|z^k - z\| \\ &\geqslant -\frac{1}{4}\|z^k - z^{k-1}\|^2 - \frac{\beta}{4}\|z^k - z\|^2,\end{aligned} \tag{24}$$

and hence $\mathcal{V}_k^{(z,u)} \geqslant \mathcal{E}_k^{(z,u)} - \frac{\beta}{4}\|z^k - z\|^2$. Bounding the bilinear term in (21) by Cauchy–Schwarz and Young's inequalities yields the second inequality in (23). The third one follows from the fact that $\tau_{k-1}\|K\| < \delta(1-\beta)/2 < 1 - \beta/2$, for all $k \in \mathbb{N}$. □

To conduct our convergence analysis, we first prove a technical lemma on the functions $\mathcal{V}_k^{(z,u)}$ that involves the following sequences:

$$\omega^{k+1} := \frac{1}{\tau_{k-1}}\big(u^k - u^{k+1} + \tau_{k-1}Kz^k\big), \tag{25}$$

$$\nu^{k+1} := \frac{1}{\tau_k}\big(z^k - z^{k+1} - \tau_k K^* u^{k+1} - \tau_{k-1}K^*(u^{k+1} - u^k) - \tau_k B(z^k) - \tau_{k-1}(B(z^k) - B(z^{k-1}))\big). \tag{26}$$

Note that (14) and (15) yield $\omega^{k+1} \in \operatorname{gra} C^{-1}(u^{k+1})$ and $\nu^{k+1} \in \operatorname{gra} A(z^{k+1})$, for all $k \in \mathbb{N}$.

Lemma 5.4. *Suppose that the assumptions of Problem 1.2 hold. Consider the sequences (z^k) and (u^k) generated by Algorithm 1. For any fixed pair $(z,u) \in \mathcal{H} \times \mathcal{G}$ and all $k \in \mathbb{N}$, define the residual*

$$\mathcal{R}_k^{(z,u)} := \tau_k \big\langle z^{k+1} - z,\, \nu^{k+1} + K^* u + B(z^{k+1}) \big\rangle + \tau_{k-1} \big\langle u^{k+1} - u,\, \omega^{k+1} - Kz \big\rangle. \tag{27}$$

Then

$$\begin{aligned} \mathcal{R}_k^{(z,u)} + \mathcal{V}_{k+1}^{(z,u)} \leqslant \mathcal{V}_k^{(z,u)} &- \frac{1}{2}\left(\frac{1-\beta}{2} - \tau_{k-1}\|K\|\right)\|z^{k+1} - z^k\|^2 \\ &- \frac{1}{2}(1 - \tau_{k-1}\|K\|)\|u^{k+1} - u^k\|^2. \end{aligned} \tag{28}$$

Proof. From (26), we have

$$\begin{aligned} &\tau_k \big\langle z^{k+1} - z, \nu^{k+1} + K^* u + B(z^{k+1}) \big\rangle \\ &\quad = \langle z^{k+1} - z, z^k - z^{k+1} \rangle - \tau_k \langle z^{k+1} - z, K^*(u^{k+1} - u) \rangle - \tau_{k-1} \langle z^{k+1} - z, K^*(u^{k+1} - u^k) \rangle \\ &\qquad - \tau_k \langle z^{k+1} - z, B(z^k) - B(z^{k+1}) \rangle - \tau_{k-1} \langle z^{k+1} - z, B(z^k) - B(z^{k-1}) \rangle, \end{aligned} \tag{29}$$

where the first term in the second line above can be rearranged as

$$\langle z^{k+1} - z, z^k - z^{k+1} \rangle = \frac{1}{2}\big(\|z^k - z\|^2 - \|z^{k+1} - z\|^2 - \|z^{k+1} - z^k\|^2\big).$$

Similarly, from (25),

$$\begin{aligned} &\tau_{k-1} \big\langle u^{k+1} - u, \omega^{k+1} - Kz \big\rangle \\ &\quad = \langle u^{k+1} - u, u^k - u^{k+1} + \tau_{k-1} K z^k - \tau_{k-1} K z \rangle \\ &\quad = \frac{1}{2}\big(\|u^k - u\|^2 - \|u^{k+1} - u\|^2 - \|u^{k+1} - u^k\|^2\big) + \tau_{k-1} \langle u^{k+1} - u, K(z^k - z) \rangle. \end{aligned} \tag{30}$$

Adding (29)-(30) and rearranging terms, we get

$$\begin{aligned} \mathcal{R}_k^{(z,u)} + \mathcal{E}_{k+1}^{(z,u)} = &\frac{1}{2}\big(\|z^k - z\|^2 + \|u^k - u\|^2\big) - \frac{1}{2}\big(\|z^{k+1} - z^k\|^2 + \|u^{k+1} - u^k\|^2\big) \\ &+ \tau_k \langle B(z^{k+1}) - B(z^k), z^{k+1} - z \rangle - \tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^{k+1} - z \rangle \\ &+ \tau_{k-1} \langle K(z^k - z), u^{k+1} - u \rangle - \tau_{k-1} \langle K(z^{k+1} - z), u^{k+1} - u^k \rangle. \end{aligned} \tag{31}$$

The last two terms in (31) can be rearranged as

$$\begin{aligned} &\tau_{k-1} \langle K(z^k - z), u^{k+1} - u \rangle - \tau_{k-1} \langle K(z^{k+1} - z), u^{k+1} - u^k \rangle \\ &\quad = \tau_{k-1} \langle K(z^k - z), u^k - u \rangle + \tau_{k-1} \langle K(z^k - z), u^{k+1} - u^k \rangle - \tau_{k-1} \langle K(z^{k+1} - z), u^{k+1} - u^k \rangle \\ &\quad = \tau_{k-1} \langle K(z^k - z), u^k - u \rangle - \tau_{k-1} \langle K(z^{k+1} - z^k), u^{k+1} - u^k \rangle \\ &\quad \leqslant \tau_{k-1} \langle K(z^k - z), u^k - u \rangle + \frac{\tau_{k-1}}{2}\|K\|\big(\|z^{k+1} - z^k\|^2 + \|u^{k+1} - u^k\|^2\big), \end{aligned} \tag{32}$$

where the last inequality is due to Cauchy–Schwarz and Young's inequalities. We now draw our attention to the second term in the second line of (31). By adding and subtracting $\tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^k \rangle$, we get

$$\begin{aligned} &-\tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^{k+1} - z \rangle \\ &\quad = -\tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^k - z \rangle - \tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^{k+1} - z^k \rangle \\ &\quad \leqslant -\tau_{k-1} \langle B(z^k) - B(z^{k-1}), z^k - z \rangle + \frac{1}{4}\big(\|z^k - z^{k-1}\|^2 + \beta\|z^{k+1} - z^k\|^2\big), \end{aligned} \tag{33}$$

where the inequality is due to (16) and follows similarly to (24). Combining equations (32)-(33) and replacing in (31), we obtain

$$\begin{aligned}\mathcal{R}_k^{(z,u)} + \mathcal{E}_{k+1}^{(z,u)} - \tau_k\langle B(z^{k+1}) - B(z^k), z^{k+1} - z\rangle \\ \leqslant \frac{1}{2}\big(\|z^k - z\|^2 + \|u^k - u\|^2\big) + \tau_{k-1}\langle K(z^k - z), u^k - u\rangle + \frac{1}{4}\|z^k - z^{k-1}\|^2 \\ - \tau_{k-1}\langle B(z^k) - B(z^{k-1}), z^k - z\rangle - \frac{1}{2}(1 - \tau_{k-1}\|K\|)\|u^{k+1} - u^k\|^2 \\ - \frac{1}{2}\left(1 - \frac{\beta}{2} - \tau_{k-1}\|K\|\right)\|z^{k+1} - z^k\|^2.\end{aligned}$$

Finally, adding $\|z^{k+1} - z^k\|^2/4$ to both sides of the equation above we obtain (28) by noting that the first three terms in the right-hand side define $\mathcal{V}_k^{(z,u)}$. □

Proposition 5.5. *Suppose that the assumptions of Problem 1.2 hold. Let $(z^*, u^*) \in \mathcal{H} \times \mathcal{G}$ be a solution of the primal-dual inclusion (13). Let (z^k) and (u^k) be the sequences generated by Algorithm 1. The following assertions hold:*

(i) for all $k \in \mathbb{N}$, it holds

$$\begin{aligned}\mathcal{V}_{k+1}^{(z^*,u^*)} \leqslant \mathcal{V}_k^{(z^*,u^*)} - \frac{1}{2}\left(\frac{1-\beta}{2} - \tau_{k-1}\|K\|\right)\|z^{k+1} - z^k\|^2 \\ - \frac{1}{2}(1 - \tau_{k-1}\|K\|)\|u^{k+1} - u^k\|^2;\end{aligned} \tag{34}$$

(ii) the sequences $(\|z^{k+1} - z^k\|)$ and $(\|u^{k+1} - u^k\|)$ converge to 0;

(iii) the sequences (z^k) and (u^k) are bounded.

Proof. (i) By Lemma 5.4 it suffices to show that the residual (27) is nonnegative at $(z, u) = (z^*, u^*)$. Indeed, from (13), $c^* := Kz^* \in C^{-1}(u^*)$ and $a^* := -K^*u^* - B(z^*) \in A(z^*)$, and thus

$$\begin{aligned}\mathcal{R}_k^{(z^*,u^*)} &= \tau_k\big\langle z^{k+1} - z^*, \nu^{k+1} - a^* + B(z^{k+1}) - B(z^*)\big\rangle + \tau_{k-1}\big\langle u^{k+1} - u^*, \omega^{k+1} - c^*\big\rangle \\ &= \tau_k\big\langle z^{k+1} - z^*, \nu^{k+1} - a^*\big\rangle + \tau_k\big\langle z^{k+1} - z^*, B(z^{k+1}) - B(z^*)\big\rangle + \tau_{k-1}\big\langle u^{k+1} - u^*, \omega^{k+1} - c^*\big\rangle \\ &\geqslant 0,\end{aligned}$$

where the inequality follows from the monotonicity of A, of B, and of C^{-1}, by recalling that $\omega^{k+1} \in \operatorname{gra} C^{-1}(u^{k+1})$ and $\nu^{k+1} \in \operatorname{gra} A(z^{k+1})$, for all $k \in \mathbb{N}$.

(ii) Let $N \in \mathbb{N}\backslash\{0\}$. By summing (34) from $k = 0$ to N, telescoping and rearranging we get

$$\begin{aligned}\mathcal{V}_0^{(z^*,u^*)} &\geqslant \mathcal{V}_0^{(z^*,u^*)} - \mathcal{V}_{N+1}^{(z^*,u^*)} \\ &\geqslant \frac{1}{2}\sum_{k=0}^{N}\left(\frac{1-\beta}{2} - \tau_{k-1}\|K\|\right)\|z^{k+1} - z^k\|^2 + \frac{1}{2}\sum_{k=0}^{N}(1 - \tau_{k-1}\|K\|)\|u^{k+1} - u^k\|^2,\end{aligned} \tag{35}$$

where the first inequality is due to Lemma 5.3. Hence, the sums in the right-hand side of (35) are finite when $N \to \infty$. Furthermore, since $1-\tau_{k-1}\|K\| > (1-\beta)/2-\tau_{k-1}\|K\| > (1-\delta)(1-\beta)/2 > 0$, for all $k \in \mathbb{N}$, we conclude that $(\|z^{k+1} - z^k\|)$ and $(\|u^{k+1} - u^k\|)$ converge to 0 as $k \to \infty$.

(iii) By combining (35) and (23), there exists a constant $R > 0$ such that, for all $N \in \mathbb{N}\backslash\{0\}$,

$$\begin{aligned}R &\geqslant \mathcal{V}_{N+1}^{(z^*,u^*)} \\ &\geqslant \frac{1}{2}\left(1 - \frac{\beta}{2} - \tau_N\|K\|\right)\|z^{N+1} - z^*\|^2 + \frac{1}{2}\left(1 - \tau_N\|K\|\right)\|u^{N+1} - u^*\|^2.\end{aligned} \tag{36}$$

Since $\tau_N < \delta(1-\beta)/(2\|K\|)$, we have $1-\tau_N\|K\| > 1-\beta/2-\tau_N\|K\| > 1/2+(1-\delta)(1-\beta)/2 > 1/2$. This leads to the bound

$$\frac{1}{4}\|z^{N+1} - z^*\|^2 + \frac{1}{4}\|u^{N+1} - u^*\|^2 \leqslant R, \quad \text{for all } N \in \mathbb{N}\backslash\{0\}. \tag{37}$$

Thus proving that the sequences (z^k) and (u^k) are bounded. □

For the convergence result of Algorithm 1 we need the following assumption on the stepsize sequences (τ_k). Later, in Remark 5.8, we will show that the assumption holds when a backtracking procedure is used for selecting τ_k.

Assumption 5.6. *The sequence (τ_k) generated by the linesearch in Algorithm 1 is bounded away from 0 whenever the sequence (z^k, u^k) is bounded.*

We are now in a position to state the convergence result of Algorithm 1.

Theorem 5.7. *Suppose that the assumptions of Problem 1.2 hold. In addition, suppose Assumption 5.6 holds. Suppose there exists at least one solution of the monotone inclusion (2). Then the sequence (z^k, u^k) generated by Algorithm 1 converges to a primal-dual solution of (2).*

Proof. By Proposition 5.5(iii), the sequence (z^k, u^k) is bounded. Let (z^{k_j}, u^{k_j}) be a subsequence of (z^k, u^k) converging to some accumulation point $(\tilde{z}, \tilde{u})$. In what follows we prove that $(\tilde{z}, \tilde{u})$ is a primal-dual solution of (2).

Indeed, the iteration defined by (14)-(15) along this subsequence can be expressed in matrix form as

$$\underbrace{\begin{bmatrix} C^{-1} & -K \\ K^* & A+B \end{bmatrix}}_{M:=} \begin{bmatrix} u^{k_j+1} \\ z^{k_j+1} \end{bmatrix} \ni \begin{bmatrix} \frac{u^{k_j}-u^{k_j+1}}{\tau_{k_j-1}} - K(z^{k_j+1} - z^{k_j}) \\ \frac{z^{k_j}-z^{k_j+1}}{\tau_{k_j}} - \frac{\tau_{k_j-1}}{\tau_{k_j}} K^*(u^{k_j+1} - u^{k_j}) - b_{k_j} + \frac{\tau_{k_j-1}}{\tau_{k_j}} b_{k_j-1} \end{bmatrix}, \tag{38}$$

where $b_\ell := B(z^\ell) - B(z^{\ell+1})$. Observe that the operator M can be decomposed as the sum of the Cartesian product $C^{-1} \times A$, which is maximally monotone, with a continuous operator on $\mathcal{G} \times \overline{\operatorname{dom} A}$. Since both the graphs of continuous operators and maximally monotone operators are closed (see, e.g., [5, Proposition 20.38]) the graph of M is closed on $\mathcal{G} \times \overline{\operatorname{dom} A}$. By Proposition 5.5(iii), the sequence (z^k, u^k) is bounded, and Assumption 5.6 yields $\inf_{k\in\mathbb{N}} \tau_k > 0$. Together with the uniform upper bound on (τ_k) on Step 2 of Algorithm 1, this guarantees that the quotients $1/\tau_{k_j-1}$, $1/\tau_{k_j}$ and τ_{k_j-1}/τ_{k_j} in (38) remain bounded. Combining all these with Proposition 5.5(ii) and recalling the continuity of B, by passing to the limit in (38) as $j \to \infty$ the closedness of the graph of M yields

$$\begin{bmatrix} C^{-1} & -K \\ K^* & A+B \end{bmatrix} \begin{bmatrix} \tilde{u} \\ \tilde{z} \end{bmatrix} \ni \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

which proves that $(\tilde{z}, \tilde{u})$ is a primal-dual solution of (2).

By Proposition 5.5(i) the sequence $\left(\mathcal{V}_k^{(\tilde{z},\tilde{u})}\right)$ is nonincreasing and nonnegative due to Lemma 5.3. Hence, the limit $\lim_{k\to\infty} \mathcal{V}_k^{(\tilde{z},\tilde{u})}$ exists, and since $z^k - z^{k-1} \to 0$, we have

$$\begin{aligned} \lim_{k\to\infty} \mathcal{V}_k^{(\tilde{z},\tilde{u})} &= \lim_{k\to\infty} \mathcal{E}_k^{(\tilde{z},\tilde{u})} \\ &= \lim_{j\to\infty} \mathcal{E}_{k_j}^{(\tilde{z},\tilde{u})} \\ &= \lim_{j\to\infty} \frac{1}{2}\left(\|z^{k_j} - \tilde{z}\|^2 + \|u^{k_j} - \tilde{u}\|^2\right) + \tau_{k_j-1}\langle K(z^{k_j} - \tilde{z}), u^{k_j} - \tilde{u}\rangle) = 0. \end{aligned}$$

Finally, using the same type of estimation as in (36)-(37), we obtain

$$\lim_{k\to\infty} \mathcal{V}_k^{(\tilde{z},\tilde{u})} \geqslant \lim_{k\to\infty} \frac{1}{4}\|z^k - \tilde{z}\|^2 + \frac{1}{4}\|u^k - \tilde{u}\|^2 = 0,$$

which proves that (z^k, u^k) converge to the primal-dual solution $(\tilde{z}, \tilde{u})$. □

Remark 5.8 (Backtracking linesearch)**.** *The linesearch in Step 2 of Algorithm 1 can be conducted by a backtracking procedure. For instance, for all $k \in \mathbb{N}$, we can take $\alpha \in (0,1)$, choose a fixed initial trial stepsize $\bar{\tau} \in \big(0, \delta(1-\beta)/(2\|K\|)\big)$, and select τ_k as $\tau_k = \alpha^j \bar{\tau}$ with j being the smallest integer such that τ_k satisfies (16). In this case, it is easy to check that Assumption 5.6 holds. To see this, let us first observe that the set of trial points corresponding to $\tau \in (0, \bar{\tau}]$ is bounded for all $k \in \mathbb{N}$ whenever the sequences (z^k) and (u^k) are bounded (which in practice holds thanks to Proposition 5.5(iii)). That is, we study the boundedness of the set $Z := \{z^{k+1}(\tau) : \tau \in (0, \bar{\tau}], k \in \mathbb{N}\} \subseteq \operatorname{dom} A$, where $z^{k+1}(\tau)$ is defined as in (19). For all $\tau \in (0, \bar{\tau}]$, we obtain the estimation*

$$\begin{aligned} \big\|z^{k+1}(\tau)\big\| &\leqslant \big\|z^{k+1}(\tau) - J_{\tau A}\big(z^{k+1}(0)\big)\big\| + \big\|J_{\tau A}\big(z^{k+1}(0)\big)\big\| \\ &\leqslant \bar{\tau}\big\|K^* u^{k+1} + B(z^k)\big\| + \big\|J_{\tau A}\big(z^{k+1}(0)\big)\big\|, \end{aligned} \tag{39}$$

for all $k \in \mathbb{N}$. Now, let $(a, a^*) \in \operatorname{gra} A$. This in turn implies that $a = J_{\tau A}(a + \tau a^*)$ for all $\tau > 0$. Hence, the nonexpansivity of the resolvent and the triangle inequality yield

$$\begin{aligned} \big\|J_{\tau A}\big(z^{k+1}(0)\big)\big\| &\leqslant \big\|J_{\tau A}\big(z^{k+1}(0)\big) - J_{\tau A}\big(a + \tau a^*\big)\big\| + \big\|J_{\tau A}\big(a + \tau a^*\big)\big\| \\ &\leqslant \|z^{k+1}(0) - a\| + \bar{\tau}\|a^*\| + \|a\|, \end{aligned} \tag{40}$$

for all $\tau \in (0, \bar{\tau}]$ and all $k \in \mathbb{N}$. Since (z^k) and (u^k) are bounded, we conclude that the sequence $\big(z^{k+1}(0)\big)$ is also bounded (recall (20)), which together with (39)-(40) ensures that the set Z is bounded. Hence, we can take a compact convex set $U \subseteq \overline{\operatorname{dom}} A$ containing $\{z^k\} \cup Z$ such that B is Lipschitz continuous on U with Lipschitz constant $L \geqslant 0$. Thus, for all $k \in \mathbb{N}$, we can conclude that the inequality

$$\tau_k \|B\big(z^{k+1}(\tau_k)\big) - B\big(z^k\big)\| \leqslant \frac{\sqrt{\beta}}{2}\|z^{k+1}(\tau_k) - z^k\|,$$

holds for some $\tau_k \in [\alpha^\ell \bar{\tau}, \bar{\tau}]$, with ℓ being the smallest integer such that $\alpha^\ell \bar{\tau} \leqslant \sqrt{\beta}/(2L)$.

The previous remark leads to the following corollary of Theorem 5.7.

Corollary 5.9. *Suppose that the assumptions of Problem 1.2 hold. Suppose there exists at least one solution of the monotone inclusion (2). Let (z^k, u^k) be the sequence generated by Algorithm 1 where the linesearch is performed following a backtracking procedure as detailed in Remark 5.8. Then (z^k, u^k) converges to a primal-dual solution of (2).*

5.2 Ergodic convergence rates

In this section, we focus on the consensus min-max problems (9). In particular, we obtain ergodic convergence rates for a gap function associated with this problem. We start by recalling the primal variable $\mathbf{z} := (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ of (11). In addition, we will use $\mathbf{u} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ for the dual variable for this problem. In accordance with these variables, we will also employ boldface for the sequence $(\mathbf{z}^k, \mathbf{u}^k)$ of primal-dual variables generated by Algorithm 1 applied to (11).

The following notation will be used throughout this section. For $\mathbf{z} := (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$, we define

$$h(\mathbf{z}) := f(\mathbf{x}) + g(\mathbf{y}) \quad \text{and} \quad \Phi(\mathbf{x}, \mathbf{y}) := f(\mathbf{x}) + \phi(\mathbf{x}, \mathbf{y}) - g(\mathbf{y}),$$

where f, g and ϕ are defined as in (8). Observe that $\partial h = A$, where A is the maximally monotone operator defined in (10), while Φ is the saddle function in (7). Given a reference point $\tilde{\mathbf{z}} := (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ and a vector $\tilde{\mathbf{u}} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$, we consider the *consensus-penalised gap function*

$$\mathcal{G}(\mathbf{z}; \tilde{\mathbf{z}}, \tilde{\mathbf{u}}) := \Phi(\mathbf{x}, \tilde{\mathbf{y}}) - \Phi(\tilde{\mathbf{x}}, \mathbf{y}) + \langle K\mathbf{z}, \tilde{\mathbf{u}}\rangle, \quad \mathbf{z} := (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}. \tag{41}$$

The following lemma shows that the consensus-penalised gap is nonnegative when the reference points are given by a primal-dual solution of (11). The proof is included for completeness.

Proposition 5.10. *Let $(\mathbf{z}^*, \mathbf{u}^*)$ be a primal-dual solution in the sense of (13) for the monotone inclusion problem (11). Then*

$$\mathcal{G}(\mathbf{z}; \mathbf{z}^*, \mathbf{u}^*) \geqslant 0, \quad \textit{for all } \mathbf{z} = (\mathbf{x}, \mathbf{y}) \in \operatorname{dom} h.$$

Proof. The primal-dual inclusion (13) with the operators in (10) reduces to

$$\begin{cases} 0 = K\mathbf{z}^*, \\ \partial f(\mathbf{x}^*) \ni -K_1^*\mathbf{u}_x^* - \nabla_{\mathbf{x}}\phi(\mathbf{x}^*, \mathbf{y}^*), \\ \partial g(\mathbf{y}^*) \ni -K_2^*\mathbf{u}_y^* + \nabla_{\mathbf{y}}\phi(\mathbf{x}^*, \mathbf{y}^*), \end{cases} \tag{42}$$

where $\mathbf{u}^* = (\mathbf{u}_x^*, \mathbf{u}_y^*)$. Let $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \in \operatorname{dom} h$. The convexity of f and g then yields

$$f(\mathbf{x}) - f(\mathbf{x}^*) \geqslant \langle K_1^*\mathbf{u}_x^* + \nabla_{\mathbf{x}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{x}^* - \mathbf{x}\rangle, \quad g(\mathbf{y}) - g(\mathbf{y}^*) \geqslant \langle K_2^*\mathbf{u}_y^* - \nabla_{\mathbf{y}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{y}^* - \mathbf{y}\rangle. \tag{43}$$

Furthermore, by the convexity of $\phi(\cdot, \mathbf{y}^*)$ and the concavity of $\phi(\mathbf{x}^*, \cdot)$, we have

$$\phi(\mathbf{x}, \mathbf{y}^*) - \phi(\mathbf{x}^*, \mathbf{y}^*) \geqslant \langle \nabla_{\mathbf{x}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{x} - \mathbf{x}^*\rangle, \quad \phi(\mathbf{x}^*, \mathbf{y}^*) - \phi(\mathbf{x}^*, \mathbf{y}) \geqslant \langle -\nabla_{\mathbf{y}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{y} - \mathbf{y}^*\rangle.$$

By adding the two equations above, we get

$$\phi(\mathbf{x}, \mathbf{y}^*) - \phi(\mathbf{x}^*, \mathbf{y}) \geqslant \langle \nabla_{\mathbf{x}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{x} - \mathbf{x}^*\rangle + \langle -\nabla_{\mathbf{y}}\phi(\mathbf{x}^*, \mathbf{y}^*), \mathbf{y} - \mathbf{y}^*\rangle. \tag{44}$$

Therefore, for all $\mathbf{z} \in \operatorname{dom} h$, making use of the relations (43) and (44) we obtain

$$\begin{aligned} \mathcal{G}(\mathbf{z}; \mathbf{z}^*, \mathbf{u}^*) &= \Phi(\mathbf{x}, \mathbf{y}^*) - \Phi(\mathbf{x}^*, \mathbf{y}) + \langle K\mathbf{z}, \mathbf{u}^*\rangle \\ &\geqslant \langle K_1^*\mathbf{u}_x^*, \mathbf{x}^* - \mathbf{x}\rangle + \langle K_2^*\mathbf{u}_y^*, \mathbf{y}^* - \mathbf{y}\rangle + \langle K\mathbf{z}, \mathbf{u}^*\rangle \\ &= \langle K\mathbf{z}^*, \mathbf{u}^*\rangle = 0, \end{aligned}$$

where the last equality is due to the first equation in (42). □

Below, we rely on the following technical lemma, which is a specialisation of Lemma 5.4 to our current setting.

Lemma 5.11. *Suppose that the functions f_i, g_i and ϕ_i satisfy the assumptions of Problem 1.1, for all $i \in \{1, 2, \ldots, n\}$. Let $\tilde{\mathbf{z}} = (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \left(\mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}\right) \cap \operatorname{dom} h$ and $\tilde{\mathbf{u}} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$. Let $(\mathbf{z}^k)$ and $(\mathbf{u}^k)$ be the sequences generated by Algorithm 1 applied to (11). Then, for all $k \in \mathbb{N}$, it holds*

$$\begin{aligned} \tau_k\mathcal{G}(\mathbf{z}^{k+1}; \tilde{\mathbf{z}}, \tilde{\mathbf{u}}) + \mathcal{V}_{k+1}^{(\tilde{\mathbf{z}}, \tilde{\mathbf{u}})} \leqslant \mathcal{V}_k^{(\tilde{\mathbf{z}}, \tilde{\mathbf{u}})} &- \frac{1}{2}\left(\frac{1-\beta}{2} - \tau_{k-1}\|K\|\right)\|\mathbf{z}^{k+1} - \mathbf{z}^k\|^2 \\ &- \frac{1}{2}(1 - \tau_{k-1}\|K\|)\|\mathbf{u}^{k+1} - \mathbf{u}^k\|^2. \end{aligned} \tag{45}$$

Proof. By Lemma 5.4, it suffices to show that when specialised to problem (11) the residual $\mathcal{R}_k^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})}$ is lower bounded by $\tau_k \mathcal{G}(\mathbf{z}^{k+1}; \tilde{\mathbf{z}}, \tilde{\mathbf{u}})$, for all $k \in \mathbb{N}$.

Fix $k \in \mathbb{N}$. Since $C = N_{\{0\}}$, the element $\omega^{k+1} \in C^{-1}(u^{k+1})$ in (25) vanishes and $K\tilde{\mathbf{z}} = 0$ by assumption, then the second summand in the residual (27) becomes zero. Thus, we get

$$\mathcal{R}_k^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})} = \tau_k \big\langle \mathbf{z}^{k+1} - \tilde{\mathbf{z}},\, \nu^{k+1} \big\rangle + \tau_k \big\langle K\mathbf{z}^{k+1}, \tilde{\mathbf{u}} \big\rangle + \tau_k \big\langle B(\mathbf{z}^{k+1}), \mathbf{z}^{k+1} - \tilde{\mathbf{z}} \big\rangle.$$

The subdifferential inequality for $A = \partial h$ then yields $\langle \mathbf{z}^{k+1} - \tilde{\mathbf{z}}, \nu^{k+1} \rangle \geqslant h(\mathbf{z}^{k+1}) - h(\tilde{\mathbf{z}})$. Next, using the convexity of $\phi(\cdot, \mathbf{y}^{k+1})$ and the concavity of $\phi(\mathbf{x}^{k+1}, \cdot)$, we get the inequalities

$$\phi(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}) - \phi(\tilde{\mathbf{x}}, \mathbf{y}^{k+1}) \leqslant \langle \nabla_{\mathbf{x}} \phi(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}), \mathbf{x}^{k+1} - \tilde{\mathbf{x}} \rangle$$

and

$$\phi(\mathbf{x}^{k+1}, \tilde{\mathbf{y}}) - \phi(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}) \leqslant \langle \nabla_{\mathbf{y}} \phi(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}), \tilde{\mathbf{y}} - \mathbf{y}^{k+1} \rangle.$$

Combining both inequalities yields

$$\langle B(\mathbf{z}^{k+1}), \mathbf{z}^{k+1} - \tilde{z} \rangle \geqslant \phi(\mathbf{x}^{k+1}, \tilde{\mathbf{y}}) - \phi(\tilde{\mathbf{x}}, \mathbf{y}^{k+1}).$$

Altogether, we conclude that

$$\mathcal{R}_k^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})} \geqslant \tau_k \big(h(\mathbf{z}^{k+1}) - h(\tilde{\mathbf{z}}) + \phi(\mathbf{x}^{k+1}, \tilde{\mathbf{y}}) - \phi(\tilde{\mathbf{x}}, \mathbf{y}^{k+1}) + \langle K\mathbf{z}^{k+1}, \tilde{\mathbf{u}} \rangle \big) = \tau_k \mathcal{G}(\mathbf{z}^{k+1}; \tilde{\mathbf{z}}, \tilde{\mathbf{u}}),$$

which yields the desired result. □

We are now in a position to state our result on ergodic convergence rates. To this end, for all $N \in \mathbb{N}$, we define the ergodic sequences

$$S_N := \sum_{k=0}^{N} \tau_k \quad \text{and} \quad \widehat{\mathbf{z}}^N = \big(\widehat{\mathbf{x}}^N, \widehat{\mathbf{y}}^N\big) := \frac{1}{S_N} \sum_{k=0}^{N} \tau_k \left(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}\right), \tag{46}$$

where $(\mathbf{z}^k) = (\mathbf{x}^k, \mathbf{y}^k)$ and (τ_k) stand for the sequences generated by Algorithm 1 applied to (11).

Theorem 5.12. *Suppose that the functions f_i, g_i and ϕ_i satisfy the assumptions of Problem 1.1 for all $i \in \{1, 2, \ldots, n\}$. Let $(\widehat{\mathbf{z}}^N)$ be the ergodic sequence defined in (46). The following assertions hold.*

(i) For every $\tilde{\mathbf{z}} = (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in \big(\mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}\big) \cap \operatorname{dom} h$ and $\tilde{\mathbf{u}} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$, and for all $N \in \mathbb{N}$, it holds

$$\mathcal{G}(\widehat{\mathbf{z}}^N; \tilde{\mathbf{z}}, \tilde{\mathbf{u}}) \leqslant \frac{\mathcal{V}_0^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})}}{S_N}. \tag{47}$$

(ii) In addition, suppose that (11) has at least one solution and that Assumption 5.6 holds. Then the sequence $(\widehat{\mathbf{z}}^N)$ converges to $\mathbf{z}^* \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ as $N \to \infty$, with $\mathbf{z}^*$ being the solution of (11) to which the sequence $(\mathbf{z}^k)$ generated by Algorithm 1 converges by Theorem 5.7.

(iii) Furthermore, for all $N \in \mathbb{N}$, we get

$$\big\|K\widehat{\mathbf{z}}^N\big\| = \mathcal{O}\left(\frac{1}{N}\right) \quad \text{and} \quad \big|\Phi(\widehat{\mathbf{x}}^N, \bar{\mathbf{y}}) - \Phi(\bar{\mathbf{x}}, \widehat{\mathbf{y}}^N)\big| = \mathcal{O}\left(\frac{1}{N}\right), \tag{48}$$

for any primal solution $\bar{\mathbf{z}} = (\bar{\mathbf{x}}, \bar{\mathbf{y}}) \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ of (11).

Proof. (i) Let $N \in \mathbb{N}$. Summing inequality (45) from $k = 0$ to N, dropping the nonpositive terms in the right-hand side (recall that $\tau_{k-1}\|K\| < \delta(1-\beta)/2$) and using the nonnegativity of $\mathcal{V}_{N+1}^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})}$ (recall Lemma 5.3), we get

$$\sum_{k=0}^{N} \tau_k \mathcal{G}(\mathbf{z}^{k+1}; \tilde{\mathbf{z}}, \tilde{\mathbf{u}}) \leqslant \mathcal{V}_0^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})} - \mathcal{V}_{N+1}^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})} \leqslant \mathcal{V}_0^{(\tilde{\mathbf{z}},\tilde{\mathbf{u}})}.$$

Dividing by S_N and using the convexity of $\mathcal{G}(\cdot\,; \tilde{\mathbf{z}}, \tilde{\mathbf{u}})$, then (47) follows.

(ii) By Proposition 5.5(iii), the sequence $(\mathbf{z}^k, \mathbf{u}^k)$ is bounded, and then Assumption 5.6 yields, for all $k \in \mathbb{N}$, $\tau_k \geqslant \tau_{\min}$ for some $\tau_{\min} > 0$. In particular, $S_N \geqslant (N+1)\tau_{\min} \to \infty$. Moreover, the upper bound in Step 2 of Algorithm 1 yields $\tau_k < \delta(1-\beta)/(2\|K\|)$, for all $k \in \mathbb{N}$. Let $\varepsilon > 0$. By Theorem 5.7, there exists $N_\varepsilon \in \mathbb{N}$ such that $\|\mathbf{z}^{k+1} - \mathbf{z}^*\| \leqslant 2\tau_{\min}\varepsilon\|K\|/(\delta(1-\beta))$ for all $k \geqslant N_\varepsilon$. Hence, for all $N \geqslant N_\varepsilon$,

$$\begin{aligned}
\|\widehat{\mathbf{z}}^N - \mathbf{z}^*\| &\leqslant \frac{1}{S_N}\sum_{k=0}^{N} \tau_k \|\mathbf{z}^{k+1} - \mathbf{z}^*\| \\
&\leqslant \frac{1}{S_N}\sum_{k=0}^{N_\varepsilon - 1} \tau_k \|\mathbf{z}^{k+1} - \mathbf{z}^*\| + \frac{1}{S_N}\sum_{k=N_\varepsilon}^{N} \tau_k \|\mathbf{z}^{k+1} - \mathbf{z}^*\| \\
&< \frac{1}{(N+1)\tau_{min}}\sum_{k=0}^{N_\varepsilon - 1} \tau_k \|\mathbf{z}^{k+1} - \mathbf{z}^*\| + \varepsilon,
\end{aligned}$$

and letting $N \to \infty$ gives $\limsup_{N\to\infty} \|\widehat{\mathbf{z}}^N - \mathbf{z}^*\| \leqslant \varepsilon$. Since $\varepsilon > 0$ was arbitrary, $\widehat{\mathbf{z}}^N \to \mathbf{z}^*$. Finally, recall that $\mathbf{z}^* \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ since $K\mathbf{z}^* = 0$.

(iii) We now show the first estimate in (48). If $K\widehat{\mathbf{z}}^N = 0$ there is nothing to prove. Otherwise, let $(\bar{\mathbf{z}}, \bar{\mathbf{u}})$ be a primal-dual solution of (11) and set $\tilde{\mathbf{u}} := \bar{\mathbf{u}} + K\widehat{\mathbf{z}}^N/\|K\widehat{\mathbf{z}}^N\|$ to get

$$\mathcal{G}(\widehat{\mathbf{z}}^N; \bar{\mathbf{z}}, \tilde{\mathbf{u}}) = \mathcal{G}(\widehat{\mathbf{z}}^N; \bar{\mathbf{z}}, \bar{\mathbf{u}}) + \|K\widehat{\mathbf{z}}^N\|.$$

Set $R_0 := \max\{\mathcal{V}_0^{(\bar{\mathbf{z}},\mathbf{u})} : \|\mathbf{u} - \bar{\mathbf{u}}\| \leqslant 1\} \geqslant 0$. As $(\bar{\mathbf{z}}, \bar{\mathbf{u}})$ is a primal-dual solution, $\mathcal{G}(\widehat{\mathbf{z}}^N; \bar{\mathbf{z}}, \bar{\mathbf{u}}) \geqslant 0$ by Proposition 5.10. It then follows from (47) that

$$\|K\widehat{\mathbf{z}}^N\| \leqslant \mathcal{G}(\widehat{\mathbf{z}}^N; \bar{\mathbf{z}}, \tilde{\mathbf{u}}) \leqslant \frac{\mathcal{V}_0^{(\bar{\mathbf{z}},\tilde{\mathbf{u}})}}{S_N} \leqslant \frac{R_0}{S_N} \leqslant \frac{R_0}{(N+1)\tau_{\min}},$$

which proves the desired rate. For the second estimate in (48), take $\tilde{\mathbf{z}} := \bar{\mathbf{z}}$ and $\tilde{\mathbf{u}} := 0$ in (47) to obtain

$$\Phi(\widehat{\mathbf{x}}^N, \bar{\mathbf{y}}) - \Phi(\bar{\mathbf{x}}, \widehat{\mathbf{y}}^N) \leqslant \frac{R_1}{(N+1)\tau_{\min}}, \tag{49}$$

where $R_1 := \mathcal{V}_0^{(\bar{\mathbf{z}},0)} \geqslant 0$. Moreover, using the Cauchy-Schwarz inequality and the estimate above for $\|K\widehat{\mathbf{z}}^N\|$,

$$\begin{aligned}
0 &\leqslant \mathcal{G}(\widehat{\mathbf{z}}^N; \bar{\mathbf{z}}, \bar{\mathbf{u}}) \\
&= \Phi(\widehat{\mathbf{x}}^N, \bar{\mathbf{y}}) - \Phi(\bar{\mathbf{x}}, \widehat{\mathbf{y}}^N) + \langle K\widehat{\mathbf{z}}^N, \bar{\mathbf{u}}\rangle \\
&\leqslant \Phi(\widehat{\mathbf{x}}^N, \bar{\mathbf{y}}) - \Phi(\bar{\mathbf{x}}, \widehat{\mathbf{y}}^N) + \frac{R_0}{(N+1)\tau_{\min}}\|\bar{\mathbf{u}}\|.
\end{aligned} \tag{50}$$

Hence, (49) and (50) conclude the second estimate in (48) □

6 Distributed proximal-reflected-gradient algorithm with linesearch for min-max problems

We now turn our attention to deriving a distributed algorithm for Problem 1.1. This will be achieved by an appropriate application of Algorithm 1 to the problem reformulation outlined in Section 4.

Let us recall the primal variable $\mathbf{z} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$ of problem (11), the operators A, B and the matrix K defined in (10). In addition, define the dual variable $\mathbf{u} \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$. By noting that $C = N_{\{0\}}$, [5, Corollary 16.3] yields that $J_{C^{-1}} = J_0 = \mathrm{Id}$ and the update rules (14)-(15) in Algorithm 1 applied to (11) can be written as

$$\begin{cases} \mathbf{u}^{k+1} = \mathbf{u}^k + \tau_{k-1} K\mathbf{z}^k, \\ \mathbf{w}^{k+1} = \mathbf{z}^k - \tau_k K^*\mathbf{u}^{k+1} - \tau_{k-1} K^*(\mathbf{u}^{k+1} - \mathbf{u}^k) - \tau_k B(\mathbf{z}^k) - \tau_{k-1}\big(B(\mathbf{z}^k) - B(\mathbf{z}^{k-1})\big), \\ \mathbf{z}^{k+1} = J_{\tau_k A}(\mathbf{w}^{k+1}). \end{cases} \tag{51}$$

Note that the subroutines related to the linesearch are omitted in (51). We will later address these details and discuss a procedure that allows for a distributed linesearch implementation, but for now we will concentrate on ensuring that the updates described by iteration (51) are performed in a decentralised fashion. To this end, observe that information exchange between agents occurs exclusively when evaluating the linear operators K and K^*. To better understand the role of these matrix-vector operations in the communication pattern, let us take as a representative example the $\mathbf{x}$ component in the product $K\mathbf{z} = (K_1\mathbf{x}, K_2\mathbf{y})$. The premultiplication of the vector $\mathbf{x} \in \mathbb{R}^{n\times p}$ by the matrix $K_1 = (K_{1,ij})_{i,j=1}^n$ leads to a vector $K_1\mathbf{x} \in \mathbb{R}^{n\times p}$ whose i-th row is given by

$$(K_1\mathbf{x})_i = \sum_{j=1}^n K_{1,ij} x_j^\top, \quad \text{for all } i \in \{1, 2, \dots, n\}.$$

Thus, the new vectors given by the rows of $K_1\mathbf{x}$ are obtained as linear combinations of the agent vectors $x_1, x_2, \dots, x_n$. We aim to adapt these communication rounds to the desired network topology, whose structure we assume to be determined by the (undirected) network graphs $\mathcal{G}_1 := (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 := (\mathcal{V}_2, \mathcal{E}_2)$, for the variables $\mathbf{x}$ and $\mathbf{y}$, respectively. Ideally, information exchange should be kept to a minimum—preferably to a single round per iteration. Distributed algorithms typically achieve this by leveraging *mixing matrices*.

Definition 6.1 (Mixing matrix)**.** *Let $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ be an undirected connected graph with $|\mathcal{V}| = n$ nodes. A matrix $W \in \mathbb{R}^{n\times n}$ is said to be a* mixing matrix *if*

(i) $W_{ij} = 0$, *whenever* $(i, j) \notin \mathcal{E}$;

(ii) $W = W^T$;

(iii) $\ker(I - W) = \mathcal{D}_{\mathbb{R}^n}$;

(iv) $W \preceq I$, *where I is the identity matrix.*[5]

Generally, one can construct a mixing matrix of a graph $\mathcal{G}$ as $I - \mathcal{L}/\sigma$ with $\mathcal{L}$ the Laplacian matrix and $\sigma > \lambda_{max}(\mathcal{L})/2$. Further examples can be consulted, for instance, in [28, Section 2.4]. Property (i) ensures that only connected agents can exchange information. Conditions (ii)-(iv) are used to define a proper consensus matrix that allows for a single communication round in an algorithmic iteration. This will become clear shortly.

[5] Usually mixing matrices are also required to also satisfy the condition $-I \prec W$, but this is not required in our analysis.

Recall that every agent in our framework is in charge of both variables x_i and y_i. We will allow different network topologies for each of these variables. That is, the graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ associated to the variables $\mathbf{x}$ and $\mathbf{y}$ respectively are not necessarily the same. Hence, two different mixing matrices will be considered for each of such graphs. For $t \in \{1, 2\}$, let W_t be a mixing matrix of $\mathcal{G}_t$. We set $K_t = \big(I - W_t\big)^{1/2}$, which is well-defined by conditions (ii) and (iv). In particular, K in (51) will be taken as $K = (K_1, K_2)$ to obtain a satisfactory distributed implementation of our algorithm. The following proposition presents some useful properties of K that motivate this choice.

Proposition 6.2. *The following properties hold:*

(i) $K^* = K$ *and* $KK = (I - W)$ *with* $W := (W_1, W_2)$*;*

(ii) $\ker K = \mathcal{D}_{\mathbb{R}^n} \times \mathcal{D}_{\mathbb{R}^n}$*;*

(iii) $\|K\| = \sqrt{1 - \min\big\{\lambda_{\min}(W_1), \lambda_{\min}(W_2)\big\}}$.

Proof. (i) It easily follows by construction. (ii) By the definition of the square root of a positive semidefinite matrix and Definition 6.1(iii), $\ker K_i = \ker(I - W_i)^{1/2} = \ker(I - W_i)$. Let $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times q}$. Then $\mathbf{z} \in \ker K \iff (\mathbf{x}, \mathbf{y}) \in \ker(I - W_1) \times \ker(I - W_2) \iff (\mathbf{x}, \mathbf{y}) \in \mathcal{D}_{\mathbb{R}^n} \times \mathcal{D}_{\mathbb{R}^n}$. (iii) Finally, to compute the norm of K it suffices to notice that this operator can be written in matrix form as

$$K = \begin{bmatrix} (I - W_1)^{1/2} & 0 \\ 0 & (I - W_2)^{1/2} \end{bmatrix}.$$

Hence, its norm is given by its maximum eigenvalue, which is simply

$$\|K\| = \max\big\{\lambda_{\max}\big((I - W_1)^{1/2}\big), \lambda_{\max}\big((I - W_2)^{1/2}\big)\big\} = \sqrt{1 - \min\big\{\lambda_{\min}(W_1), \lambda_{\min}(W_2)\big\}},$$

concluding the proof. □

By Proposition 6.2(ii), the linear operator K enforces consensus on both variables $\mathbf{x}$ and $\mathbf{y}$. In addition, Proposition 6.2(i) allows us to obtain an iteration equivalent to (51) that requires only a single communication round. Indeed, by defining the new variable $\mathbf{v}^k = K\mathbf{u}^k$, premultiplying the first equation in (51) by K, and performing the appropriate change of variables, we get the equivalent iteration

$$\begin{cases} \mathbf{v}^{k+1} = \mathbf{v}^k + \tau_{k-1}(I - W)\mathbf{z}^k, \\ \mathbf{w}^{k+1} = \mathbf{z}^k - \tau_k \mathbf{v}^{k+1} - \tau_{k-1}^2 (I - W)\mathbf{z}^k - \tau_k B(\mathbf{z}^k) - \tau_{k-1}\big(B(\mathbf{z}^k) - B(\mathbf{z}^{k-1})\big), \\ \mathbf{z}^{k+1} = J_{\tau_k A}(\mathbf{w}^{k+1}), \end{cases} \tag{52}$$

where by the definition of the sequence $(\mathbf{v}^k)$ it is required that $\mathbf{v}^0 \in \operatorname{range} K$. The latter trivially holds if $\mathbf{v}^0 = 0$. Consequently, (52) describes a distributed update where only a single communication round is performed per iteration to transmit the information encoded by the vector $\mathbf{z}^k$.

The last step to complete our distributed algorithm for Problem 1.1 is to tailor the linesearch in Step 2 of Algorithm 1 to agent-wise operations. To this end, we follow the linesearch proposed in [3, Section 4.2] and devise a linesearch procedure where the n agents communicate to conduct a global scalar sum. The distributed algorithm resulting from combining iteration (52) with our proposed linesearch procedure is presented in Algorithm 2. For conciseness, we have included the auxiliary sequences $(\mathbf{s}_x^k)$ and $(\mathbf{s}_y^k)$. Clearly, these variables agree with our distributed implementation, as each agent locally computes their corresponding component in these vectors.

Algorithm 2: Distributed proximal-reflected-gradient algorithm with linesearch (DPRGL) for Problem 1.1

Choose mixing matrices $W_1, W_2 \in \mathbb{R}^{n\times n}$;
Choose parameters $\delta \in (0,1)$, $\beta \in (0,1)$ and $\alpha \in (0,1)$;
Choose a positive lower bound for the initial trial stepsizes $\underline{\tau} < \delta \frac{1-\beta}{2\sqrt{1-\min\{\lambda_{\min}(W_1),\lambda_{\min}(W_2)\}}}$;
Choose an initial trial stepsize $\tau_{-1} \in \left(\underline{\tau}, \delta \frac{1-\beta}{2\sqrt{1-\min\{\lambda_{\min}(W_1),\lambda_{\min}(W_2)\}}}\right)$;
Choose initial points $\mathbf{x}^{-1} = \mathbf{x}^0 \in \operatorname{dom} f$, $\mathbf{y}^{-1} = \mathbf{y}^0 \in \operatorname{dom} g$, $\mathbf{v}_x^0 = 0$, $\mathbf{v}_y^0 = 0$ and set

$$\mathbf{s}_x^{-1} := \nabla_x \phi(\mathbf{x}^{-1}, \mathbf{y}^{-1}), \quad \mathbf{s}_y^{-1} := -\nabla_y \phi(\mathbf{x}^{-1}, \mathbf{y}^{-1});$$

for $k = 0, 1, \ldots$ **do**

▷ **Step 1** (Dual update) **for** $i = 1, 2, \ldots, n$ **do**

$$\begin{cases} v_{x,i}^{k+1} = v_{x,i}^k + \tau_{k-1}\big(x_i^k - \sum_{j=1}^n W_{1,ij} x_j^k\big), \\ v_{y,i}^{k+1} = v_{y,i}^k + \tau_{k-1}\big(y_i^k - \sum_{j=1}^n W_{2,ij} y_j^k\big). \end{cases}$$

▷ **Step 2** (Linesearch and primal update). Take $\tau_{k(0)} \in \left(\underline{\tau}, \delta \frac{1-\beta}{2\sqrt{1-\min\{\lambda_{\min}(W_1),\lambda_{\min}(W_2)\}}}\right)$ and set

$$\mathbf{s}_x^k := \nabla_x \phi(\mathbf{x}^k, \mathbf{y}^k), \quad \mathbf{s}_y^k := -\nabla_y \phi(\mathbf{x}^k, \mathbf{y}^k).$$

for $\ell = 0, 1, \ldots$ **do**

▷ **Step 2.a** (Primal agent-wise operations) **for** $i = 1, 2, \ldots, n$ **do**

Compute

$$\begin{cases} x_i^{k(\ell)+1} = \operatorname{prox}_{\tau_{k(\ell)} f_i}\Big(x_i^k - \tau_{k(\ell)} v_{x,i}^{k+1} - \tau_{k-1}^2\big(x_i^k - \sum_{j=1}^n W_{1,ij} x_j^k\big) - \tau_{k(\ell)} s_{x,i}^k - \tau_{k-1}(s_{x,i}^k - s_{x,i}^{k-1})\Big), \\ y_i^{k(\ell)+1} = \operatorname{prox}_{\tau_{k(\ell)} g_i}\Big(y_i^k - \tau_{k(\ell)} v_{y,i}^{k+1} - \tau_{k-1}^2\big(y_i^k - \sum_{j=1}^n W_{2,ij} y_j^k\big) - \tau_{k(\ell)} s_{y,i}^k - \tau_{k-1}(s_{y,i}^k - s_{y,i}^{k-1})\Big). \end{cases}$$

and the auxiliary scalar quantities

$$\begin{cases} a_i := \tau_{k(\ell)}^2 \|\nabla_x \phi_i(x_i^{k(\ell)+1}, y_i^{k(\ell)+1}) - s_{x,i}^k\|^2 - \frac{\beta}{4}\|x_i^{k(\ell)+1} - x_i^k\|^2, \\ c_i := \tau_{k(\ell)}^2 \| - \nabla_y \phi_i(x_i^{k(\ell)+1}, y_i^{k(\ell)+1}) - s_{y,i}^k\|^2 - \frac{\beta}{4}\|y_i^{k(\ell)+1} - y_i^k\|^2. \end{cases} \tag{53}$$

▷ **Step 2.b** (Check the linesearch condition). Compute the global sum $\sum_{i=1}^n (a_i + c_i)$, and if

$$\sum_{i=1}^n (a_i + c_i) > 0, \tag{54}$$

set $\tau_{k(\ell+1)} = \alpha \tau_{k(\ell)}$ and go back to Step 2.a. Otherwise set $\tau_k = \tau_{k(\ell)}$, $\mathbf{x}^{k+1} = \mathbf{x}^{k(\ell)+1}$, $\mathbf{y}^{k+1} = \mathbf{y}^{k(\ell)+1}$ and go back to Step 1.

Specifically, the linesearch in Algorithm 2 is a backtracking-type procedure consisting of two stages. First, in Step 2.a, every agent computes locally its contribution to the global sum in the linesearch. Secondly, this global scalar sum is computed in Step 2.b. The following result formalises the connection between Algorithm 1 and Algorithm 2. In addition, it ensures that the linesearch in Algorithm 2 satisfies Assumption 5.6.

Proposition 6.3. *Suppose that the functions f_i, g_i and ϕ_i satisfy the assumptions of Problem 1.1, for all $i \in \{1, 2, \ldots, n\}$. Then the following assertions hold.*

(i) The sequence $(\mathbf{z}^k) = (\mathbf{x}^k, \mathbf{y}^k)$ generated by Algorithm 2 coincides with that of Algorithm 1 applied to (11) initialised with $\mathbf{u}^0 = 0$ and under a backtracking linesearch.

(ii) Furthermore, the sequence (τ_k) generated by Algorithm 2 satisfies Assumption 5.6.

Proof. (i) Let us start by formalising the iterative scheme of Algorithm 1 under the conditions in (i). A general iteration of Algorithm 1 with a backtracking linesearch applied to (11) is written in compact form as (51) where the stepsize τ_k is given by $\tau_{k(\ell)} := \alpha^\ell \tau_{k(0)}$ with $\tau_{k(0)} \in \big(0, \delta(1-\beta)/(2\sqrt{1-\min\{\lambda_{\min}(W_1), \lambda_{\min}(W_2)\}})\big)$ (since $\|K\| = \sqrt{1-\min\{\lambda_{\min}(W_1), \lambda_{\min}(W_2)\}}$ by Proposition 6.2(iii)) and where ℓ is the smallest integer such that

$$\tau_{k(\ell)}\|B(\mathbf{z}^{k(\ell)+1}) - B(\mathbf{z}^k)\| \leqslant \frac{\sqrt{\beta}}{2}\|\mathbf{z}^{k(\ell)+1} - \mathbf{z}^k\|, \tag{55}$$

where we use $k(\ell)+1$ on the superscript of $\mathbf{z}^{k(\ell)+1}$ to emphasize the dependence of this vector on the stepsize. Recall that the iterations conducted by Algorithm 2 are simply the agent-wise counterpart of (52). Hence, to prove the desired statement, two things are needed. On the one hand, to show that, under an equivalent initialisation, the schemes (51) and (52) generate the same sequence $(\mathbf{z}^k)$. Further, that the condition (54) used for the backtracking linesearch in Algorithm 2 is equivalent to (55). We start with the latter. Indeed, by recalling the definition of a_i and c_i in (53) we have

$$\begin{aligned}
&\sum_{i=1}^n (a_i + c_i) \leqslant 0 \\
&\qquad \iff \sum_{i=1}^n \Big(\tau_{k(\ell)}^2\|\nabla_x \phi_i(x_i^{k(\ell)+1}, y_i^{k(\ell)+1}) - s_{x,i}^k\|^2 + \tau_{k(\ell)}^2\| - \nabla_y \phi_i(x_i^{k(\ell)+1}, y_i^{k(\ell)+1}) - s_{y,i}^k\|^2\Big) \\
&\qquad\qquad \leqslant \sum_{i=1}^n \left(\frac{\beta}{4}\|x_i^{k(\ell)+1} - x_i^k\|^2 + \frac{\beta}{4}\|y_i^{k(\ell)+1} - y_i^k\|^2\right) \\
&\qquad \iff \tau_{k(\ell)}\|B(\mathbf{z}^{k(\ell)+1}) - B(\mathbf{z}^k)\| \leqslant \frac{\sqrt{\beta}}{2}\|\mathbf{z}^{k(\ell)+1} - \mathbf{z}^k\|,
\end{aligned}$$

as desired.

Now, consider the iterations (51) and (52) initialised with the same choice of $\mathbf{z}^{-1}$ and $\mathbf{z}^0$ and with $\mathbf{u}^0 = 0$ for (51) and $\mathbf{v}^0 = 0$ for (52). It trivially holds that $\mathbf{z}^0$ coincides for the two iterations and that $\mathbf{v}^0 = K\mathbf{u}^0$. By induction, assume that there exists some $\hat{k} \in \mathbb{N}$ such that these relationships hold for all $k \leqslant \hat{k}$. We next show that they are preserved for $\hat{k}+1$.

Multiplying the first equation of (51) by K and using the identity $KK = I - W$ from Proposition 6.2(i), we obtain

$$K\mathbf{u}^{\hat{k}+1} = K\mathbf{u}^{\hat{k}} + \tau_{\hat{k}-1}KK\mathbf{z}^{\hat{k}} = \mathbf{v}^{\hat{k}} + \tau_{\hat{k}-1}(I-W)\mathbf{z}^{\hat{k}}.$$

This exactly matches the update for $\mathbf{v}^{\hat{k}+1}$ in (52), establishing that $\mathbf{v}^{\hat{k}+1} = K\mathbf{u}^{\hat{k}+1}$. Next, we substitute this relation into the primal update. Using $K^* = K$, we have $K^*\mathbf{u}^{\hat{k}+1} = \mathbf{v}^{\hat{k}+1}$.

Furthermore, the difference term in the second equation of (51) becomes

$$K^*(\mathbf{u}^{\hat{k}+1} - \mathbf{u}^{\hat{k}}) = \mathbf{v}^{\hat{k}+1} - \mathbf{v}^{\hat{k}} = \tau_{\hat{k}-1}(I - W)\mathbf{z}^{\hat{k}},$$

where the last equality is due to the first equation in (52). Substituting these into the second equation of (51) yields

$$\mathbf{w}^{\hat{k}+1} = \mathbf{z}^{\hat{k}} - \tau_{\hat{k}}\mathbf{v}^{\hat{k}+1} - \tau_{\hat{k}-1}^2(I - W)\mathbf{z}^{\hat{k}} - \tau_{\hat{k}}B(\mathbf{z}^{\hat{k}}) - \tau_{\hat{k}-1}\big(B(\mathbf{z}^{\hat{k}}) - B(\mathbf{z}^{\hat{k}-1})\big),$$

which is exactly the update for $\mathbf{w}^{\hat{k}+1}$ in (52). This finally proves that the sequence $(\mathbf{z}^k)$ generated by the iterative scheme (52) is identical to the one generated by (51).

(ii) Furthermore, since we showed above that the linesearch in Algorithm 2 is given by a backtracking mechanism, by tailoring the argument in Remark 5.8 to this setting, we conclude that the sequence of stepsizes (τ_k) in Algorithm 2 is bounded away from 0 by $\alpha^\ell \underline{\tau}$, where $\underline{\tau}$ is the lower bound imposed on the initial trial stepsizes in Algorithm 2 and ℓ is the smallest integer such that $\alpha^\ell \underline{\tau} \leqslant \sqrt{\beta}/(2L)$, with L being the analogous to the Lipschitz constant considered in Remark 5.8. This shows that Assumption 5.6 holds. □

We finally state our main result on the convergence of Algorithm 2 to solutions of Problem 1.1.

Theorem 6.4. *Suppose that Problem 1.1 has at least a solution and that the subdifferential sum rules in (12) hold. Then the sequence $(\mathbf{x}^k, \mathbf{y}^k)$ generated by Algorithm 2 converges to a point $(\mathbf{x}^*, \mathbf{y}^*) \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ such that any pair of rows (x^*, y^*) of $(\mathbf{x}^*, \mathbf{y}^*)$ solve Problem 1.1. In addition, for the sequences $(\widehat{\mathbf{x}}^N)$ and $(\widehat{\mathbf{y}}^N)$ defined as in (46), it holds that $(\widehat{\mathbf{x}}^N, \widehat{\mathbf{y}}^N) \to (\mathbf{x}^*, \mathbf{y}^*)$ and*

$$\left|\Phi(\widehat{\mathbf{x}}^N, \mathbf{y}^*) - \Phi(\mathbf{x}^*, \widehat{\mathbf{y}}^N)\right| = \mathcal{O}\left(\frac{1}{N}\right).$$

Proof. By Proposition 4.1, the monotone inclusion (11) has at least a solution. Then, by Proposition 6.3 and Theorem 5.7, the sequence $(\mathbf{x}^k, \mathbf{y}^k)$ converges to a point $(\mathbf{x}^*, \mathbf{y}^*) \in \mathcal{D}_{\mathbb{R}^{n\times p}} \times \mathcal{D}_{\mathbb{R}^{n\times q}}$ solving (11). Finally, by again resorting to Proposition 4.1 the conclusion follows. The last assertion is a direct consequence of Proposition 6.3 and Theorem 5.12. □

7 Concluding remarks

In this work, we proposed a distributed proximal-reflected-gradient method with linesearch (DPRGL) to solve convex-concave min-max problems in the form of Problem 1.1. The key novelty of DPRGL is its ability to handle saddle functions with locally Lipschitz continuous partial gradients without requiring prior knowledge of the local Lipschitz moduli, thereby generalising the framework in [19]. Our analysis relies on a novel primal-dual forward-reflected splitting algorithm with linesearch for composite monotone inclusion problems involving single-valued, locally Lipschitz continuous operators. A practical limitation of the proposed scheme is that the distributed linesearch implementation relies on a global scalar reduction across all agents (similar to the routines in [3]). A promising focus for future research is to design stepsize selection mechanisms that achieve fully decentralised execution without requiring global coordination. This issue has been addressed in different settings by some recent works but under the more restrictive condition of global Lipschitz continuity of the gradients; see, e.g., [17, 31].

Statements and Declarations

Funding FA was partially supported by Centro de Modelamiento Matemático (CMM), ACE210010 and FB210005, BASAL funds for center of excellence from ANID-Chile. The research of MND and

MKT was supported in part by Australian Research Council grant DP230101749. DTB was partially supported by grant PID2025-168246NB-I00 funded by MICIU/AEI/10.13039/501100011033 and ERDF/EU, as well as by grant CIGE/2026/80 by Conselleria d'Educació, Cultura i Universitats, Generalitat Valenciana.

Competing interests The authors declare no competing interests.

References

[1] F. J. Aragón-Artacho, Y. Malitsky, M. K. Tam, and D. Torregrosa-Belén. Distributed forward-backward methods for ring networks. *Comput. Optim. Appl.*, 86(3):845–870, 2023.

[2] F. Atenas, H. H. Bauschke, M. N. Dao, and M. K. Tam. Relocated fixed-point iterations with applications to variable stepsize resolvent splitting. *SIAM J. Optim.*, 36(3):1297–1325, 2026.

[3] F. Atenas, M. N. Dao, and M. K. Tam. A distributed proximal splitting method with linesearch for locally Lipschitz gradients. *J. Glob. Optim.*, 2025. Published online.

[4] H. Attouch and M. Théra. A general duality principle for the sum of two operators. *J. Convex Anal.*, 3(1):1–24, 1996.

[5] H. H. Bauschke and P. L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, Cham, second edition, 2017.

[6] M. Bertero and P. Boccacci. Application of the OS-EM method to the restoration of LBT images. *Astron. Astrophys. Suppl. Ser.*, 144(1):181–186, 2000.

[7] A. Beznosikov, V. Samokhin, and A. Gasnikov. Distributed saddle point problems: lower bounds, near-optimal and robust algorithms. *Optim. Methods Softw.*, 40(5):1249–1266, 2025.

[8] L. M. Briceño Arias and D. Davis. Forward-backward-half forward algorithm for solving monotone inclusions. *SIAM J. Optim.*, 28(4):2839–2871, 2018.

[9] L. M. Briceño Arias and F. Roldán. Four-operator splitting via a forward-backward-half-forward algorithm with line search. *J. Optim. Theory Appl.*, 195(1):205–225, 2022.

[10] A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision*, 40(1):120–145, 2011.

[11] X. Chen, I. Kuruzov, G. Scutari, and A. Gasnikov. A parameter-free decentralized algorithm for composite convex optimization. In *2025 IEEE 64th Conf. Decis. Control (CDC)*, pages 1544–1551. IEEE, 2025.

[12] L. Condat. A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *J. Optim. Theory Appl.*, 158(2):460–479, 2013.

[13] N. Cressie and T. R. C. Read. Multinomial goodness-of-fit tests. *J. Roy. Statist. Soc. Ser. B*, 46(3):440–464, 1984.

[14] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng. Training GANs with optimism. In *ICLR*, 2018.

[15] D. Kovalev, A. Beznosikov, A. Sadiev, M. Persiianov, P. Richtarik, and A. Gasnikov. Optimal algorithms for decentralized stochastic variational inequalities. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Adv. Neural Inf. Process. Syst.*, volume 35, pages 31073–31088. Curran Associates, Inc., 2022.

[16] I. Kuruzov, A. Rogozin, D. Yarmoshik, and A. Gasnikov. The mirror-prox sliding method for non-smooth decentralized saddle-point problems. *Optim. Methods Softw.*, 40(5):1197–1221, 2025.

[17] Z. Li, W. Shi, and M. Yan. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Trans. Signal Process.*, 67(17):4494–4506, 2019.

[18] Y. Malitsky and M. K. Tam. A forward-backward splitting method for monotone inclusions without cocoercivity. *SIAM J. Optim.*, 30(2):1451–1472, 2020.

[19] Y. Malitsky and M. K. Tam. A first-order algorithm for decentralised min-max problems. *Numer. Math.*, 158(3):1187–1206, 2026.

[20] S. Mukherjee and M. Chakraborty. A decentralized algorithm for large scale min-max problems. In *59th IEEE Conf. Decis. Control (CDC)*, pages 2967–2972, 2020.

[21] H. Namkoong and J. Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Adv. Neural Inf. Process. Syst.*, volume 29. Curran Associates, Inc., 2016.

[22] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM J. Optim.*, 19(4):1574–1609, 2008.

[23] L. D. Popov. A modification of the Arrow–Hurwicz method for search of saddle points. *Math. Notes Acad. Sci. USSR*, 28(5):845–848, 1980.

[24] J. Rieger and M. K. Tam. Backward-forward-reflected-backward splitting for three operator monotone inclusions. *Appl. Math. Comput.*, 381:125248, 10, 2020.

[25] R. T. Rockafellar. *Convex analysis.* Princeton University Press, Princeton, 1970.

[26] A. Rogozin, A. Beznosikov, D. Dvinskikh, D. Kovalev, P. Dvurechensky, and A. Gasnikov. Decentralized saddle point problems via non-Euclidean mirror prox. *Optim. Methods Softw.*, 40(5):1127–1152, 2025.

[27] E. K. Ryu and W. Yin. *Large-scale convex optimization—algorithms & analyses via monotone operators.* Cambridge University Press, Cambridge, 2023.

[28] W. Shi, Q. Ling, G. Wu, and W. Yin. EXTRA: an exact first-order algorithm for decentralized consensus optimization. *SIAM J. Optim.*, 25(2):944–966, 2015.

[29] W. Shi, Q. Ling, G. Wu, and W. Yin. A proximal gradient algorithm for decentralized composite optimization. *IEEE Trans. Signal Process.*, 63(22):6013–6023, 2015.

[30] X. A. Sun and A. J. Conejo. *Robust optimization in electric energy systems.* Springer, Cham, 2021.

[31] M. K. Tam, L. Timms, and L. Zhang. A decentralised forward-backward-type algorithm with network-independent heterogenous agent step sizes. *arXiv preprint: 2512.12502*, 2025.

[32] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. *SIAM J. Control Optim.*, 38(2):431–446, 2000.

[33] B. C. Vũ. A splitting algorithm for dual monotone inclusions involving cocoercive operators. *Adv. Comput. Math.*, 38(3):667–681, 2013.