First-order algorithms for robust optimization problems via convex-concave saddle-point Lagrangian reformulation

Krzysztof Postek
Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Delft, The Netherlands, k.s.postek@tudelft.nl,

Shimrit Shtern
Faculty of Industrial Engineering and Management, Technion - Israel Institute of Technology, Haifa, Israel, shimrits@technion.ac.il,

Robust optimization (RO) is one of the key paradigms for solving optimization problems affected by uncertainty. Two principal approaches for RO, the robust counterpart method and the adversarial approach, potentially lead to excessively large optimization problems. For that reason, first order approaches, based on online-convex-optimization, have been proposed Ben-Tal et al. (2015b), Ho-Nguyen and Kılınç-Karzan (2018) as alternatives for the case of large-scale problems. However, these methods are either stochastic in nature or involve a binary search for the optimal value. We show that this problem can also be solved with deterministic first-order algorithms based on a saddle-point Lagrangian reformulation that avoid both of these issues. Our approach recovers the other approaches’ $O(1/\epsilon^2)$ convergence rate in the general case, and offers an improved $O(1/\epsilon)$ rate for problems with constraints which are affine both in the decision and in the uncertainty. Experiment involving robust quadratic optimization demonstrates the numerical benefits of our approach.

1. Introduction

When solving optimization problems, one often has to deal with uncertainty present in the parameters of the objective and constraint functions. This uncertainty may stem from measurement, implementation or prediction errors. A common paradigm used to ensure that the solution remains feasible under uncertainty is robust optimization (RO) Ben-Tal et al. (2009). In RO, the uncertainty is assumed to be adversarial to the decision maker and to lie in a predefined uncertainty set, and the decision maker finds the best solution which remains feasible for all parameter values within this set.

Any RO algorithm involves simultaneous solving of two problems: the decision maker’s problem of finding the best-possible decision which is feasible for the given uncertainty set, and nature’s implicit adversarial problem of selecting the worst possible realization of the parameters.

Denoting the decision maker’s decision as $x$, taken from a predefined set $X$, and the uncertain parameter for constraint $i$ as $z_i$, taken from a predefined uncertainty set $Z^i$, a general RO problem
is given by:

\[
\min_{x \in X} \mathbf{c}^T x \quad \text{(1)}
\]

subject to:

\[
\sup_{z_i \in Z} g_i(x, z_i) \leq 0, \quad i \in [m].
\]

In this problem, the decision maker minimizes over \( x \in X \), while the nature selects \( z \in Z = Z^1 \times Z^2 \times \ldots \times Z^m \) such that the left-hand sides of the inequality constraints are maximized, in an attempt to violate the constraints.

**Standard methods.** There are two key methods for handling the opposite optimization problems simultaneously: (1) the robust counterpart (RC) reformulation method Ben-Tal et al. (2015a, 2009), (2) adversarial approach, also known as cutting planes Bienstock (2007), Mutapcic and Boyd (2009). In the RC approach, nature’s maximization problem for each constraint is dualized, and the objective of the dual problem is required to be nonpositive. In this way, a constraint is substituted by a system of inequalities that are satisfied for a given \( x \) if and only if the original left-hand sides are nonpositive. This method’s key advantage is that it requires to solve only one optimization problem and it ensures by-design the robustness of the solution. Its disadvantage is the increase of the problem size due to the added dual variables and constraints generated by dualizing nature’s problems. Moreover, for strong duality to hold for nature’s problem in each constraint, it has to be convex, or equivalent to a convex problem.

In the adversarial approach, a finite subset \( Z \subset Z \) of scenarios is iteratively built up, until it contains enough points to ensure that \( x \) is feasible for all realizations in the subset if and only if it is (almost) feasible for all realizations in \( Z \). The set \( Z \) is initialized using an arbitrarily chosen realization \( z \in Z \). Then two steps are repeated alternatingly: an optimization step, in which a solution \( x \) which minimizes the objective function and is feasible for \( z \in Z \) is found; a pessimization step, in which a new realization \( z \) violating at least one constraint is found and added to \( Z \). This iterative procedure continues until no violating scenario is found. While this method is simple and enables solving problems in which nature’s problem is not necessarily convex, the size of \( Z \) increases at each iteration, which may result in extremely large optimization problems in \( x \) during the optimization step.

**Need for lower-order approaches.** Both of the above methods potentially lead to excessively large optimization problems which creates space for approaches in which the decision maker’s and the adversary’s problems are simultaneously solved in a lightweight fashion. A recently suggested idea Ben-Tal et al. (2015b), Ho-Nguyen and Kilinç-Karzan (2018) is to use online convex optimization to solve problem (1), proving that the number of iterations needed to obtain an \( \epsilon \)-feasible \( \epsilon \)-optimal solution is \( O(\log(1/\epsilon)/\epsilon^2) \). Through the online optimization lens, the robust problem is seen
as a problem of minimizing a partly unknown objective with likewise constraints, whose shape is learned throughout the algorithm via samples.

The approach of Ben-Tal et al. (2015b) consists in iteratively solving a nominal version of (1) in which the set $Z$ is replaced by a fixed realization $z$. The value of $z$ is updated at each iteration through first-order updates/pessimization and randomization. The approach of Ho-Nguyen and Kilınç-Karzan (2018) is to use binary search to determine the minimal $\tau$ for which the feasibility problem

$$\{c^\top x \leq \tau, \sup_{z \in Z} g_i(x, z) \leq 0, \quad i \in [m]\}$$

has a solution up to a given accuracy. This requires running an online first-order algorithm for each tested $\tau$. Each of the feasibility problems is solved by a first order iteration on $x$, simultaneously with a pessimization/first order steps on the dual parameter $z$.

Thanks to the online optimization framework, both Ben-Tal et al. (2015b), Ho-Nguyen and Kilınç-Karzan (2018) work directly with the functions $\{g_i(\cdot, \cdot)\}_{i \in [m]}$, without the need to build an ever-increasing list of scenarios, or to dualize the constraints. This has a price: the functions $g_i$ have to be convex-concave, the set $X$ needs to be bounded, and the maximum achievable convergence rate is $O(1/\epsilon^2)$ to obtain an $\epsilon$-feasible and $\epsilon$-optimal solution. Also, Ben-Tal et al. (2015b) requires to solve multiple nominal problems, while in Ho-Nguyen and Kilınç-Karzan (2018) one needs to run binary search for the optimal objective value.

The method in Ho-Nguyen and Kilınç-Karzan (2018), was also extended in Ho-Nguyen and Kilınç-Karzan (2019) to cases in which the functions $g_i$ have additional characteristics to obtain $O(1/\epsilon)$ rate of convergence. Specifically, this result requires that for all $i \in [m]$ either $g_i(\cdot, z)$ or $-g_i(x, \cdot)$ is strongly convex for any $z \in Z$ or $x \in X$, respectively, and that the function is smooth in the remaining variable. In this paper, we chose for simplicity, to avoid making such further assumptions on the problem’s structure, and instead focus on two cases: the case where $g_i(\cdot, \cdot)$ are continuous real-valued functions, and the case where $g_i$ are biaffine functions.

**Research questions.** Two questions that arise from the above review are as follows.

1. **Under what conditions is** (1) **amenable to applying single-run deterministic, light-weight first order methods?**

2. **Can the guarantees obtained from such algorithms be transformed into meaningful guarantees for the robust problem** (1) **in terms of the optimality and feasibility gaps?**

**Contribution.** We address these two questions by leveraging a natural convex-concave saddle-point reformulation of (1), based on its Lagrangian. The lifted Lagrangian of problem 1, generally given by

$$\min_{x \in X, u \in U} \bar{L}(x, u),$$

(2)
where $\mathcal{L}$ is a convex in $\mathbf{x}$ and concave in the lifted variable $\mathbf{u}$. Although the existence of such a formulation was noted and discussed in (Ho-Nguyen and Kılınç-Karzan 2018, Appendix A), it has been claimed that the problem loses a lot of convenient structure due to the lifting. However, we show two simple settings under which (2) is amenable to the use of first order algorithms for convex-concave saddle point problems, not only theoretically but also practically.

Answering the first research question, we start by carefully deriving conditions under which we can map saddle points of this formulation to optimal solutions of problem (3). Next we turn to solve this formulation using simple first order algorithms.

There is an abundance of applicable first-order algorithms, usually derived for solving variational inequalities of monotone operators (see (Nemirovski 1994, Chapter 6)), each demanding a different set of assumptions. Indeed, most algorithms demand that the part of the saddle point function connecting the primal and dual variables be smooth in both (e.g., (Nemirovski 1994, 2004, Juditsky and Nemirovski 2011, Gidel et al. 2017) and references therein), which is not the case in our general Lagrange derivation. The convergence of such algorithms is usually given by $L(\bar{x}^N, \bar{u}^*) - L(x^*, \bar{u}^N) \leq \epsilon_N$. For convenience, in Table 1, we summarize the known state-of-the-art rates of convergence under different assumptions on the Lagrangian function, the implied assumptions on the structure of the original robust problem (See Section 3 for details), and well-known algorithms achieving these rates. Faster rates are only known for more restrictive assumptions on the functions involved (such as strong convexity (Ouyang and Xu 2021, Zhang et al. 2022)). In this paper, in order to avoid assumptions that do not generally hold in the robust setting, we focus on two simple settings.

In the first setting, we make similar assumptions to Ho-Nguyen and Kılınç-Karzan (2018), i.e., that for all $i \in [m]$, $g_i(\cdot, \cdot)$ are continuous real-valued functions, and $X$ and $Z_i$ are compact sets. Under this setting we require to use more general saddle-point algorithms, such as the one based on the subgradients of the functions, as the method discussed in (Nemirovski 1994, Chapter 9) and analyzed in Nedić and Ozdaglar (2009), Auslender and Teboulle (2009) (SGSP), which achieves a $O(1/\sqrt{N})$ rate of convergence, and can be easily extended to the Bregman setting (Mirror Descent).

In the second setting considered, we treat $g_i(\cdot, \cdot)$ that are biaffine. It is established that in this case, a first-order algorithm can not obtain a better iteration complexity than $O(1/\epsilon)$ Ouyang and Xu (2021), and thus we restrict ourselves to algorithms which achieve this optimal rate. Although the above mentioned algorithms for smooth functions fit this setting, they require additional assumptions such as compactness of the sets and multiple gradient and projection steps per iteration. Therefore, we focus on the algorithm presented in Chambolle and Pock (2011) (CP), which is less restrictive and cheaper computationally, as it only requires one gradient computation.
Table 1 Existing Saddle Point Algorithms

| SP Problem Assumptions | RO Problem Assumptions | Iteration cost | Rate of $\epsilon_N$ | Methods |
|------------------------|------------------------|----------------|-----------------------|---------|
| Compact $X$ and $U$, $\mathcal{L}$ Lipschitz continuous on $X \times U$ | Compact $X$ and $Z$, $g_i$ are Lipschitz continuous on $X \times Z$, Slater condition holds | $T_x(X) + T_u(U)$ | $O(1/\sqrt{N})$ | SPSG / Arrow-Hurwicz-Uzawa method (Auslender and Teboulle 2009, Nedić and Ozdaglar 2009) |
| $\mathcal{L}(x,u)$ biaffine | $g_i$ are biaffine | $T_x(X) + T_u(U)$ | $O(1/N)$ | Chambole-Pock (Chambolle and Pock 2011) |
| Compact $X$ and $U$, and $\mathcal{L}(x,u)$ has a jointly Lipschitz continuous gradient on $X \times U$ | Compact $X$ and $Z$, and $\lambda_i g_i(x,\tilde{z}_i/\lambda_i)$ have Lipschitz continuous gradient in both $x$ and $u_i = (\tilde{z}_i, \lambda_i)$ | $2(T_x(X) + T_u(U))$ | $O(1/N)$ | Extra gradient / Mirror-prox (Korpelevich 1976, Tseng 1991, Nemirovski 2004), Dual extrapolation (Nesterov 2007) |
| | | $T_x(X) + T_u(U)$ | $O(1/N)$** | Forward-Reflected Backward / Optimistic mirror descent (Malitsky and Tam 2020, Alacaoglu et al. 2021) |

* $T_y(C)$ indicate the computational cost of computing the (sub)gradient of $\mathcal{L}$ with respect to variable $y$ and performing an orthogonal projection on set $C$.

** This rate was proven without the compactness assumption for a weaker result.

and one projection on the sets per iteration. This algorithm can be used to obtain a sequence contained in $X$ with the gap converging at a rate of $O(1/N)$.

To address the second research question, we need to deal with the fact that saddle-point algorithms give convergence results in terms of ergodic duality gap. As we are interested in feasibility and optimality convergence rates for problem (1), we defined a general notion of Ergodically Bounded (EB) algorithms. We then that applying an EB algorithm to the saddle point formulation of (1) yields the same feasibility and optimality convergence rates as the algorithm itself. This definition allowed us to unify the analysis, as it is sufficient to show that a given algorithm, just like the two we consider, is an EB algorithm. Thus, the framework laid out in this paper can serve as a basis for analysis of other saddle point-algorithms in this context.

We note that the choice of algorithms for these two settings is somewhat arbitrary, since the aim of this paper is to illustrate the power of using the Lagrangian reformulation, and the general technique to convert a given saddle-point algorithm’s gap rate of convergence to rates of obtaining feasibility and optimality for the robust problem (1). Thus, a great deal of this paper is devoted to technical but important issues in applying the above mentioned algorithms in these settings. Specifically, we detailed sufficient and realistic assumptions needed to be satisfied by the robust problem (3), and derive bounds required for either the application of the algorithm or for showing that the algorithm assumptions are indeed satisfied. Moreover, to show the practicality of the suggested algorithms, we prove that when $Z^i$ are projection-friendly sets one may also find the projection onto the lifted space relatively easily, either analytically or by using bi-section. Finally,
we show that the saddle-point formulation actually allows for more flexibility and enables to tackle problems where either $Z^i$ or $X$ are more complicated by using splitting techniques, where these problems prove to be more challenging for the previously suggested methods.

**Paper Structure.** The remainder of the paper is structured as follows. In Section 2, we introduce the problem we solve along with the corresponding assumptions and the Lagrangian saddle-point reformulation. In Section 3, we introduce the two algorithms for the case of simple uncertainty sets and state the corresponding convergence results. In Section 4, we present the convergence analysis of the algorithms for a generalized problem form through a unified framework of showing that both are ergodically bounded (EB) algorithms. In Section 5, we compare the performance of our SGSP approach to the online first-order approaches of Ho-Nguyen and Kılınç-Karzan (2018) and the adversarial approach on randomly sampled robust quadratic optimization problems with and without constraints. Section 6 concludes the paper. All proofs not given in the body are given in the Supplementary Material.

**Notation** Throughout the paper we use bolded small letters $\mathbf{x}$, $\mathbf{z}$ for vectors, and bolded capital letters $\mathbf{P}$ for matrices, and capital letters for sets. For any $k \in \mathbb{N}$, we use shorthand notation $[k]$ to denote the set of indexes $\{1, 2, \ldots, k\}$. Unless specified otherwise, $\| \cdot \|$ refers to the Euclidean norm.

## 2. Problem setting and assumptions

### 2.1. Introduction

In this paper, we consider the following general RO problem.

\[
\min_{\mathbf{x} \in X} \mathbf{c}^T \mathbf{x} \quad \text{s.t. } f_i(\mathbf{x}) := \max_{\mathbf{z}_i \in Z^i} g_i(\mathbf{x}, \mathbf{z}_i) \leq 0 \quad i \in [m] \\
A \mathbf{x} = \mathbf{b}
\]

where $X \subseteq \mathbb{R}^n$ is a closed and convex set, for all $i \in [m]$ the set $Z^i \subseteq \mathbb{R}^{d_i}$ is a convex and compact set and without loss of generality $\mathbf{0} \in Z^i$, $g_i(\cdot, \mathbf{z}_i) : X \to \mathbb{R}$ is convex for any fixed $\mathbf{z}_i \in Z^i$, $g_i(\mathbf{x}, \cdot) : Z^i \to \mathbb{R}$ is concave for any fixed $\mathbf{x} \in X$, and $A \in \mathbb{R}^{r \times n}$, $\mathbf{b} \in \mathbb{R}^r$. Note that, contrary to Problem 1, here we allow to separate some of the affine constraints that may be involved in the definition of the domain of $\mathbf{x}$ from the set $X$.

Note that formulation (3) also encompasses robust problems involving uncertainty in the objective function, since such problems can be transformed to form (3) using the epigraph formulation of the objective. Thus, formulation (3) is general and includes many useful problems (c.f., Bertsimas et al. (2011), Gabrel et al. (2014)).
Functions $f_i(x)$ are known as the robust counterpart formulation of the robust constraint
\[ g_i(x, z_i) \leq 0, \forall z_i \in Z^i. \]

Note that while $f_i(x)$ are convex functions of $x$ (as a maximum of convex functions), they are not necessarily easily representable due to their implicit formulation as maxima.

Our aim is to solve (3) through its saddle point Lagrangian formulation. For this formulation to be well-defined, we make three standard assumptions.

**Assumption 1.** Problem (3) has an optimal solution.

**Assumption 2.** There exists $\hat{x} \in \text{int}(X)$ such that $A\hat{x} = b$ and there exists a $\epsilon_\hat{x} > 0$ such that $\hat{x} + y \in X$ and $f_i(\hat{x} + y) < 0$ for all $\|y\| \leq \epsilon_\hat{x}$.

**Assumption 3.** Matrix $A$ has full row rank.

With respect to Assumption 1, if the problem does not have an optimal solution there are three options: (i) it is infeasible, in which case the uncertainty sets defined for the problem may be too large, (ii) it is unbounded, i.e., it might not be constrained enough, or (iii) it is bounded but the optimal solution is not attained, in which case we can restrict the set $X$ to a subset containing $\epsilon$-optimal solutions of the original problem. Assumption 2, known as the Slater condition, ensures that the problem is stable, i.e., slight perturbations in the feasible set do not make the problem infeasible. Assumption 3 states that there are no redundant equality constraints.

In order to solve problem (3) we consider its Lagrangian:
\[
L(x, (\lambda, w)) \equiv c^T x + \sum_{i=1}^m \lambda_i f_i(x) + w^T (Ax - b)
\]  
Specifically, we are interested in saddle points of function $L(\cdot, \cdot)$, i.e., points $(x^*, \lambda^*)$ which satisfy
\[
L(x^*, (\lambda, w)) \leq L(x^*, (\lambda^*, w^*)) \leq L(x, (\lambda^*, w^*)), \quad \forall x \in X, \lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r.
\]

Under Assumptions 1 and 2, the Lagrangian function $L(\cdot, \cdot)$ has a saddle point, and $(x^*, (\lambda^*, w^*))$ is a saddle point of $L(\cdot, \cdot)$ if and only if $x^*$ and $(\lambda^*, w^*)$ are optimal solutions to the primal and dual problems respectively. Thus, instead of solving problem (3), we want to find a solution to
\[
\inf_{x \in X} \sup_{\lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r} L(x, (\lambda, w)).
\]

This reformulation eliminates the constraints and renders the problem as a saddle point one, enabling the use of various first-order methods. However, such methods require computing at each iteration not only the functions $f_i(x)$, but also their sub-gradients or proximal operators, which may be challenging with the implicitly-defined $f_i$ functions. We will therefore consider an alternative formulation where our goal is to work with functions $g_i$, and show how to obtain an $\epsilon$-optimal and $\epsilon$-feasible solution of problem (3).
2.2. Conversion to a convex-concave saddle point problem

We begin our transformation of problem (5) plugging in the explicit definitions of functions $f_i$, using functions $g_i$, as follows:

$$
\sup_{\lambda \in \mathbb{R}^m_+} \inf_{w \in \mathbb{R}^r} L(x, (\lambda, w)) = \sup_{\lambda \in \mathbb{R}^m_+\, w \in \mathbb{R}^r} \inf_{x \in X} c^\top x + \sum_{i=1}^m \lambda_i \sup_{z_i \in \mathcal{Z}^i} g_i(x, z_i) + w^\top (Ax - b)
$$

$$
= \sup_{\lambda \in \mathbb{R}^m_+\, w \in \mathbb{R}^r} \inf_{x \in X} \max_{z_i \in \mathcal{Z}^i, i \in [m]} c^\top x + \sum_{i=1}^m \lambda_i g_i(x, z_i) + w^\top (Ax - b)
$$

$$
= \sup_{\lambda \in \mathbb{R}^m_+\, w \in \mathbb{R}^r} \inf_{x \in X} c^\top x + \sum_{i=1}^m \lambda_i g_i(x, z_i) + w^\top (Ax - b)
$$

(6)

where the equalities follow from Sion’s Theorem, the fact $\mathcal{Z}^i$ are convex and compact, $X$ is convex, and $g_i$ are convex-concave. However, the resulting saddle point problem (6) is over a function which is convex in $x$ but is not jointly concave in $z = (z_1, \ldots, z_m)$ and $\lambda = (\lambda_1, \ldots, \lambda_m)$. Since convergence results for saddle point algorithms typically require a convex-concave structure, we need to reformulate the problem to achieve such a structure. For this, we will use a change of variables $\tilde{z}_i = \lambda_i z_i$, and inversely

$$
z_i = \begin{cases} \frac{\tilde{z}_i}{\lambda_i}, & \lambda_i > 0, \\ 0, & \text{otherwise.} \end{cases}
$$

Using this definition we have $\lambda_i g_i(x, z_i) = \lambda_i g_i(x, \tilde{z}_i/\lambda_i)$, and since $-g_i(x, \cdot)$ is convex for every $x$, $-\lambda_i g_i(x, \tilde{z}_i/\lambda_i)$ is jointly convex in $u_i = (\tilde{z}_i, \lambda_i)$ for every $x$, as a perspective of a convex function (Bauschke and Combettes 2011, Proposition 8.23). Moreover, $\lambda_i g_i(x, \tilde{z}_i/\lambda_i)$ is continuous for all $u_i \in U^i$, where $U^i = \{u_i = (\tilde{z}_i, \lambda_i) : \tilde{z}_i \in \lambda_i Z^i, \lambda_i \geq 0\}$, obtaining a value of 0 whenever $\lambda_i = 0$. Defining $u = (u_1, \ldots, u_m)$ and the set $U = U^1 \times \cdots \times U^m$, we have

$$
\sup_{u \in U, w \in \mathbb{R}^r} \inf_{x \in X} \tilde{L}(x, (u, w)) := \sup_{u \in U, w \in \mathbb{R}^r} \inf_{x \in X} c^\top x + \sum_{i=1}^m \lambda_i g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) + w^\top (Ax - b).
$$

(7)

For ease of notation, in henceforth we denote the perspective version of $g_i$ as $\tilde{g}_i(x, u_i) \equiv \lambda_i g_i(x, \tilde{z}_i/\lambda_i)$ where $u_i = (\tilde{z}_i, \lambda_i)$.

The following result shows that solving (7) is sufficient for solving (5), i.e., the saddle points of $\tilde{L}$ can be reduced to those of $L$.

**Proposition 1.** Let $(x^*, (u^*, w^*)) \in X \times U \times \mathbb{R}^r$, where $u^* = (u^*_1, \ldots, u^*_m)$ and $u^*_i = (\tilde{z}_i^*, \lambda^*_i)$ for $i \in [m]$. Then $(x^*, (u^*, w^*)) \in X \times U$ is a saddle point of $\tilde{L}$ over $X \times U \times \mathbb{R}^r$ if and only if $(x^*, (\lambda^*, w^*))$ is a saddle point of $L$ over $X \times \mathbb{R}^{m+} \times \mathbb{R}^r$.

3. Algorithms for solving saddle point formulation

With the robust problem (3) in the desired convex-concave structure (7), we move towards introducing two algorithms for solving it. As both algorithms will require boundedness of the optimal
dual solution \((\lambda^*, w^*)\), either for running or for obtaining the convergence guarantees, we begin with stating the result that uses Assumptions 1-3 to provide such bounds.

**Proposition 2.** Let Assumption 1 hold, let \(\hat{x} \) be the point satisfying Assumption 2, let \((x^*, (u^*, w^*))\) be a saddle point of \(\tilde{L}\) on the set \(X \times U \times \mathbb{R}^r\), and let \(v\) be a strict lower bound on the optimal value of problem (3). Then, \(u^* = (u^*_1, \ldots, u^*_m)\), where \(u^*_i = (\tilde{z}^*_i, \lambda^*_i)\), and \(w^*\) satisfy

\[
\lambda^*_i \leq \tilde{\lambda} := \frac{c^\top \hat{x} - v}{\max_{i \in [m]} f_i(\hat{x})}
\]

\[
\|u^*_i\| \leq \tilde{\lambda} \sqrt{1 + R_i^2}
\]

\[
\|w^*\| \leq R_w := \frac{1}{\sigma_{\min}(A)} \left( \frac{c^\top \hat{x} - v}{\varepsilon_x} + \|c\| \right),
\]

where \(R_i := \max_{x \in Z_i} \|z\|\) and \(\sigma_{\min}(A) > 0\) is the smallest singular value of \(A\).

**Proof.** **Boundedness of \(\lambda^*_i\).** Let \(x^*\) be an optimal solution to (3). Let \(\hat{x} \in X\) be a Slater point. We have

\[
c^\top x^* \leq c^\top \hat{x} + \sum_{i=1}^m \lambda^*_i f_i(\hat{x}) + (w^*)^\top (A\hat{x} - b)
\]

\[
= c^\top \hat{x} + \sum_{i=1}^m \lambda^*_i f_i(\hat{x})
\]

\[
\leq c^\top \hat{x} + \|\lambda^*\|_1 \max_{i \in [m]} f_i(\hat{x})
\]

whence

\[
\|\lambda^*\|_1 \leq \frac{c^\top (\hat{x} - x^*)}{-\max_{i \in [m]} f_i(\hat{x})} = \tilde{\lambda}.
\]

Thus, we have that \(\lambda^*_i \leq \|\lambda^*\|_1 \leq \tilde{\lambda}\) for all \(i \in [m]\).

**Boundedness of \(u^*_i\).** Since \(u^*_i = (\tilde{z}^*_i, \lambda^*_i) \in U^i\) we have that \(\tilde{z}^*_i \in \lambda^*_i Z_i\), by the Cauchy-Schwartz inequality we have that \(\|\tilde{z}^*_i\| \leq |\lambda^*_i| R_i \leq \tilde{\lambda} R_i\). Thus, \(\|u^*_i\|_* = \|\tilde{z}^*_i\|^2 + |\lambda^*_i|^2 \leq \tilde{\lambda}^2 (R_i^2 + 1)\).

**Boundedness of \(w^*\).** Consider the saddle point formulation:

\[
\inf_{x \in X} \sup_{u_i \in U_i, w} c^\top x + w^\top (A x - b) + \sum_{i=1}^m \tilde{g}_i(x, u_i) = \inf_{x \in U^i} c^\top x + w^\top (A x - b) + \sum_{i=1}^m \tilde{g}_i(x, u_i) + \delta_X(x)
\]

Assume \(x^*, w^*, u^*_i\) are the saddle point of this problem. The optimality conditions of the problem are:

\[
A x^* = b
\]

\[
c + A^\top w^* + \sum_{i=1}^m \alpha_i + \beta = 0
\]

where \(\alpha_i \in \partial_x \tilde{g}_i(x^*, u^*_i)\) and \(\beta \in \partial_x \delta_X(x^*)\). Let \(\hat{x} \in \text{int}(X)\) be the Slater point so we have

\[
\hat{x} + \kappa \in X, \quad \tilde{g}_i(\hat{x} + \kappa, u^*_i) \leq 0 \quad \forall \kappa : \|\kappa\| \leq \varepsilon_x
\]
Since \( \alpha_i \in \partial g_i(x^*, u_i^*) \), we have that
\[
\tilde{g}_i(x^*, u_i^*) + \alpha_i^T (\tilde{x} + \kappa - x^*) \leq \tilde{g}_i(\tilde{x} + \kappa, u_i^*)
\]
so that
\[
\sum_{i \in [m]} \tilde{g}_i(x^*, u_i^*) + \sum_{i \in [m]} \alpha_i^T (\tilde{x} + \kappa - x^*) \leq \sum_{i \in [m]} \tilde{g}_i(\tilde{x} + \kappa, u_i^*)
\]
where the first sum is equal to zero due to complementary slackness (since \( x^* \) and \( \lambda^* \) are optimal \( \tilde{g}_i(x^*, u_i^*) = \lambda^*_i g_i(x^*, z_i^*) = \lambda^*_i f_i(x^*) = 0 \)) and the second sum is nonpositive due to (8). Since \( \beta \in \partial \delta_X(x^*) \), we also have that
\[
\delta_X(x^*) + \beta^T (\hat{x} + \kappa - x^*) \leq \delta_X(\hat{x} + \kappa) = 0,
\]
which, combined with the optimality conditions, implies
\[
0 \leq \left( c + A^T w^* + \sum_{i=1}^m \alpha_i \right)^T (\hat{x} + \kappa - x^*) \leq (c + A^T w^*)^T (\hat{x} + \kappa - x^*), \quad \forall \|\kappa\| \leq \hat{\epsilon}.
\]
This gives us the property
\[
\epsilon \|c + A^T w^*\| \leq (c + A^T w^*)^T (\hat{x} - x^*)
\]
Because \( Ax^* = A\hat{x} = b \), we obtain \( \epsilon \|c + A^T w^*\| \leq c^T (\hat{x} - x^*) \). Since by Assumption 3, \( A \) has full row rank, by the reverse triangle inequality we can bound \( w^* \) as follows:
\[
\epsilon (\sigma_{\min}(A^T) \|w^*\| - \|c\|) \leq \epsilon (\|A^T w^*\| - \|c\|) \leq \epsilon \|c + A^T w^*\| \leq c^T (\hat{x} - x^*).
\]

We now move to presenting the two algorithms, summarized in Table 2. We will first state both for problem (7) and give their convergence without proofs. In Section 4, we shall prove the convergence of a generalized problem in a unified framework from which the ‘simple cases’ will follow as straightforward corollaries.

The first algorithm, presented in Section 3.1, applies to the case with the additional assumptions that \( X \) is bounded and the functions \( g_i \) have bounded subgradients over \( X \times Z^i \). For this setting, an iteration complexity of \( O(1/\epsilon^2) \) is attained, similar to the one obtained by Ho-Nguyen and Kilinc-Karzan (2018) under almost identical assumptions. In Section 3.2, we consider the case where \( g_i(x, z_i) \) are biaffine functions (or can be transformed to this form), and show that in this setting we can obtain a superior iteration complexity of \( O(1/\epsilon) \).
Table 2  Comparison of the two saddle point algorithms.

| Algorithm       | SGSP                  | CP                  |
|-----------------|-----------------------|---------------------|
| Domain X        | Bounded               | No restriction      |
| Structure $g_i(\cdot)$ | Any                  | Biaffine or reducible to biaffine |
| Optimality convergence rate | $O(1/\sqrt{N})$   | $O(1/N)$            |
| Feasibility convergence rate | $O(1/\sqrt{N})$   | $O(1/N)$            |
| Slater point needed to compute the stepsize | Yes              | No                 |

3.1. Subgradient Saddle Point algorithm

In this section, we show how problem (7) can be solved using the SGSP suggested in Nedić and Ozdaglar (2009). This algorithm requires that the both primal and dual variables be contained in compact sets. Thus, we first need to replace each set $U_i$ by its compact counterpart $\tilde{U}_i = \{(\tilde{z}_i, \lambda_i) \in U_i : \lambda_i \leq \bar{\lambda}\}$. Indeed, by the property of $\lambda^*$ presented in Proposition 2, this restriction of $U_i$ would not change the set of saddle points of problem (7). Similarly, we can restrict $w$ to reside in a set $W = \{w \in \mathbb{R}^r : \|w\| \leq R_w\}$. Using these new sets, the algorithm is as follows.

**SGSP: SubGradient Algorithm for Saddle Point**

**Input:** $\tau > 0, \theta_i > 0, \theta_w > 0$ for $i \in [m]$, and $N \in \mathbb{N}$

**Initialization.** Initialize $x^0 \in X$ and $u_i^0 \in \tilde{U}_i$, for $i \in [m]$, $w^0 \in W$.

**General step:** For $k \in [N]$

- Compute subgradients $v_k^x \in \partial_x \bar{L}(x_k, (u_k, w_k))$, $v_k^i \in \partial_{u_i} (-\bar{L}(x_k, (u_k, w_k)))$, for all $i \in [m]$
- $x_{k+1} = P_X(x_k - \tau v_k^x)$,
- $u_{i,k+1} = P_{\tilde{U}_i}(u_{i,k} - \theta_i v_k^i)$, $i \in [m]$
- $w_{k+1} = P_W(w_k + \theta_w (Ax_k - b))$

Note that the algorithm can be applied whenever the projections over $\{\tilde{U}_i\}_{i \in [m]}$ and $X$ can be easily computed. We will see in Section 4, that if any of the sets are intersections of multiple simpler sets, we can utilize splitting methods that enable the use of projections only on the components of the intersection. Moreover, note that at each iteration the steps for all $u_i, x$ and $w$ can be done in parallel.

As in most algorithms solving saddle point problems, the SGSP algorithm’s convergence is given in terms of the ergodic sequences, i.e., denoting

$$\bar{x}^N = \frac{1}{N} \sum_{k=1}^N x^k, \quad \bar{u}^N = \frac{1}{N} \sum_{k=1}^N u^k, \quad \bar{w}^N = \frac{1}{N} \sum_{k=1}^N w^k$$

the convergence result states the rate at which the sequence $\{(\bar{x}^N, \bar{u}^N, \bar{w}^N)\}_{N \in \mathbb{N}}$ converges to a saddle point. Here, we present the convergence in terms of the total constraint violation and the distance from the optimal value.

**Theorem 1.** Let $\{x^k, u^k, w^k\}_{k \in \mathbb{N}}$ be the sequences generated by the SGSP algorithm with step sizes

$$\tau := \frac{\hat{\tau}}{\sqrt{N}}, \quad \theta_i := \frac{\hat{\theta}_i}{\sqrt{N}}, \quad \theta_w := \frac{\hat{\theta}_w}{\sqrt{N}}$$
for $i \in [m]$. Assume that $X$ is compact and define $R_x := \max_{x \in X} \|x\|$. Further assume there exist constants $G_x, G_i$ for $i \in [m]$ such that the subgradients $\{v^k_x\}_{k \in \mathbb{N}}, \{v^k_i\}_{k \in \mathbb{N}}, i \in [m]$ generated by the algorithm satisfy
\[
\|v^k_x\| \leq G_x, \quad \|v^k_i\| \leq G_i, \quad i \in [m], \quad \forall k \in \mathbb{N}.
\]
Then, we have the following feasibility and optimality convergence guarantees.
\[
\sum_{i=1}^{m} [f_i(x^N)]_+ + \|Ax^N - b\| \leq \frac{2\max\{2, \max_i\{1 + 4R_i^2\}\}}{\sqrt{N}} \left( \frac{2\max\{\|x^0\|^2, R_x^2\}}{\tau} + \sum_{i=1}^{m} \frac{\max\{\bar{\lambda} + 1, \lambda^0_i\}^2}{\theta_i} + \frac{\max\{R_w + 1, \|w^0\|^2\}}{\theta_w^2} + \phi \right),
\]
and
\[
|c^T(x^N - x^*)| \leq \frac{\max\{2, \max_i\{1 + 4R_i^2\}\}}{\sqrt{N}} \left( \frac{2\max\{\|x^0\|^2, R_x^2\}}{\tau} + \sum_{i=1}^{m} \frac{\max\{2\bar{\lambda}, \lambda^0_i\}^2}{\theta_i} + \frac{\max\{2R_w, \|w^0\|^2\}}{\theta_w^2} + \phi \right),
\]
where
\[
\phi := \tau G_x^2 + \sum_{i=1}^{m} \theta_i G_i^2 + \theta_w G_w^2, \quad G_x := \|A\| R_x + \|b\|, \quad R_i := \max_{z \in Z^i} \|z\|.
\]
Note that almost all conditions used in Theorem 1 are also needed to apply the online first-order (OFO) approach of Ho-Nguyen and Kılınç-Karzan (2018). In fact, the two approaches give similar convergence results with a few differences:

1. **Assumptions.** SGSP requires the existence of a Slater point $\hat{x}$ while the OFO does not. Secondly, OFO requires boundedness of the subgradients of $g(x, \cdot)$ while SGSP requires boundedness of the subgradients of its prespective function. In Section 3.3.2 we show that under mild assumptions on the problems, these requirements are equivalent.

2. **Implementation.** Since the OFO approach is meant to solve a feasibility, rather than an optimality, problem, it requires to perform a binary search to approximate the optimal value of (3). Thus, the number of needed iteration to obtain feasibility and optimality guaranties is increased by a factor of $\log(1/\epsilon)$. In the SGSP in turn, we do not need to perform bi-section, however we must find a Slater point $\hat{x}$, the values of $f_i(\hat{x})$ for $i \in [m]$, and $\epsilon_{\hat{x}}$, as well as a lower bound on the objective function $\nu$, which are needed to compute both $\bar{\lambda}$ and $R_w$. While in some cases it may be easy to find these quantities, in general it requires solving an auxiliary optimization problem, as we discuss in Section 3.3.1.

3. **Projections.** While OFO requires projections on sets $X$ and $Z^i$, SGSP requires projection onto $X$ and the lifted set $\tilde{U}^i$. In Section 3.3.3 we show that for standard simple sets $Z^i$ the projections onto $\tilde{U}^i$ can be simply computed.
4. **Constants.** Although the convergence rate of both methods is $O(1/\epsilon^2)$, the constants obtained by the SGSP algorithm are worse than those of OFO, if the same first order method (subgradient/mirror decent) is used.

### 3.2. Chambolle-Pock algorithm

In this section, we present an algorithm with a superior rate of convergence of $O(1/\epsilon)$ which does not require boundedness of $X$. This algorithm requires the additional assumption that the functions $g_i(x, z_i)$ have a biaffine form:

$$g_i(x, z_i) = x^\top Q_i z_i + d_i^\top x + q_i^\top z_i + \gamma_i,$$

and that the primal variable is not constrained. In Remark 1, we show how more general problems of the form $g_i(x, z_i) := h_i(x)^\top k_i(z_i)$ with convex and concave $h_i$ and $k_i$, respectively, can be reformulated to fit this case.

To state the algorithm, we first simplify its form due to the biaffine structure. Indeed, under (9) function $\bar{L}$ reduces to

$$\bar{L}(x, (u, w)) := c^\top x + \sum_{i=1}^m \left(x^\top \tilde{Q}_i u_i + \tilde{q}_i^\top u_i\right) + w^\top (Ax - b)$$

$$= c^\top x + x^\top A^\top w + x^\top \tilde{Q} u + \tilde{q}^\top u - b^\top w,$$

where,

$$\tilde{Q}_i = [Q_i, d_i], \quad \tilde{q}_i = \begin{bmatrix} q_i \gamma_i \end{bmatrix}, \quad \tilde{Q} = (\tilde{Q}_1, \ldots, \tilde{Q}_m), \quad \tilde{q} = (\tilde{q}_1^\top, \ldots, \tilde{q}_m^\top)^\top, \quad B = (A^\top, \tilde{Q}).$$

We now state the CP algorithm of Chambolle and Pock (2011) for solving problem (10).

**CP: Chambolle-Pock first-order primal-dual algorithm**

**Input:** $\tau > 0, \sigma > 0, \tau \cdot \sigma \leq \frac{1}{\|B\|^2}$

**Initialization.** Initialize $x^0 \in \mathbb{R}^n$, $y^0 = (u^0, w^0) \in U \times \mathbb{R}^m$.

**General step:** For $k \in [N]$

- $u_i^k = P_{U_i}\left(u_i^{k-1} + \sigma(\tilde{q}_i + \tilde{Q}_i^\top \tilde{x}_i^{k-1})\right)$, $i \in [m]$
- $w^k = w^{k-1} + \sigma(Ax^{k-1} - b)$
- $x^k = P_{X}\left(x^{k-1} - \tau\left(c + A^\top w^k - \tilde{Q}_i u_i^{k}\right)\right)$
- $\tilde{x}^k = 2x^k - x^{k-1}$

Thus, similarly to the SGSP algorithm, CP can be applied whenever the projections over $U_i$ and $X$ are easily computed. Similarly to the SGSP, convergence results for the CP are in terms of the ergodic sequence.
Theorem 2. Let Assumptions 1 and 2 hold and let \( \{x^k, w^k\}_{k \in \mathbb{N}} \) be the sequence generated by the CP algorithm with some \( \tau > 0, \sigma > 0 \) satisfying \( \tau \sigma \|B\|^2 \leq 1 \). Then, we have the following convergence guarantees:

\[
\sum_{i=1}^{N} [f_i(\bar{x}^N)]_+ + \|A\bar{x}^N - b\| \leq \frac{\max\{2, \max_i (1 + 4R_i^2)\}}{2N} \left( \frac{\max\{\|x^*\|, \|x^0\|\}^2}{\tau} + \sum_{i=1}^{m} \max\{\bar{\lambda} + 1, \lambda_i^0\}^2 \frac{1}{\sigma} + \max\{R_w + 1, \|w^0\|\}^2 \right),
\]

and

\[
|c^T (x^N - x^*)| \leq \frac{\max\{2, \max_i (1 + 4R_i^2)\}}{2N} \left( \frac{\max\{\|x^*\|, \|x^0\|\}^2}{\tau} + \sum_{i=1}^{m} \max\{2\bar{\lambda}, \lambda_i^0\}^2 \frac{1}{\sigma} + \max\{2R_w, \|w^0\|\}^2 \right).
\]

3.3. Numerical implementation of saddle point algorithms

In this section, we discuss some technical aspects related to the implementation of the above algorithms. These aspects relate to (i) finding a Slater point for the SGSP algorithm using SGSP algorithm of an auxiliary problem, (ii) computing subgradients of the lifted functions \( \tilde{g}_i \) from the subgradients of the original functions \( g_i \), (iii) computing projections onto the lifted sets \( U_i \).

3.3.1. Search for a Slater point in the SGSP algorithm

To run the SGSP algorithm, we require parameter values dependent on the features of a Slater point of the problem. Therefore, before SGSP is run, we first need to identify a Slater point and plug the appropriate values into the algorithm. To find such a point, we need to solve the following optimization problem

\[
\min_{x, t} \quad t
\]

s.t. \( \max_{z_i \in Z^i} g_i(x, z_i) \leq t \quad \forall i \in [m] \)

\( Ax = b \)

\( x \in X \)

This problem satisfies the Slater condition for any \( x^0 \in \text{int}(X) \), \( Ax^0 = b \), and \( t = t^0 + \delta \) such that

\[
t^0 = \max_{i \in [m]} \max_{z_i \in Z^i} g_i(x^0, z_i).
\]

For that reason, we can first apply the SGSP algorithm to this problem by transforming it to a saddle point form. To keep the domain of \((x, t)\) compact, we can restrict \( t \) to belong to the interval \([-1, \bar{t}]\) where \( \bar{t} = t^0 + \delta \).
If we assume that the subgradients in the SGSP algorithm for the original problem (3) are bounded, then it also holds for (12). Thus, one can run the SGSP algorithm for (12), knowing that it will converge to the optimal value of $t$. A complication is the moment at which the algorithm stops. Assuming that the original problem satisfies the Slater condition with constant $\epsilon$ we would like to stop when both optimality and feasibility conditions are satisfied with $\frac{\epsilon}{3}$ thus ensuring that the $(\hat{x}, \hat{t})$ obtained by the procedure satisfies

$$\max_{z_i \in Z^i} g_i(\hat{x}, z_i) \leq \frac{\hat{t}}{3} + \frac{\epsilon}{3} + \frac{\epsilon}{3} = -\frac{\epsilon}{3}.$$ 

However, in practice, $\epsilon$ can be unknown in advance, and therefore, we construct a search procedure as follows:

**Slater Point Search**

**Input:** $\tau > 0, \theta_i > 0$ for $i \in [m]$, $\delta > 0$, $K = 2$

**Initialization.** Initialize $x^0 \in \text{int}(X)$, $t^0 = \max_{i \in [m]} f_i(x^0) + \delta$ and $u^0_i \in U_i$, for $i \in [m]$.

**General step:** For $k \in 1, 2, \ldots$

1. Run SGSP for the Lagrangian form of problem (12) starting from point $(x^{k-1}, t^{k-1})$ and $u^{k-1}$ for $K$ iterations, and obtain the ergodic values $(x^k, t^k)$ and $u^k$.
2. Update $t^k = \max_{i \in [m]} f_i(x^k)$.
3. If $t^k < 0$ stop and return the Slater point $\hat{x} = x^k$ and the Slater value $\epsilon = -t^k$. Otherwise, update $t = \min\{t, t^k + \delta\}$, $k = k + 1$ and $K = 2K$.

As stated above, this procedure is guaranteed to converge. Moreover, since $x^0 \in \text{int}(X)$ then by properties of convex sets we will also obtain that for each $k$ the iterate $x^k \in \text{int}(X)$, as the average of points in $X$ with one of them in the interior.

### 3.3.2. Determining the subgradients of $\tilde{g}_i$ from subgradients of $g_i$

To prove the convergence of SGSP we require that the subgradients of the perspective function $\tilde{g}_i$ are bounded. In this section, we show that the subgradients of $\tilde{g}_i$ can be easily derived from subgradients of $g_i$. Moreover, we will also show that under the following mild assumptions the boundedness of the subgradients of $\tilde{g}_i$ follows from the boundedness of the subgradients of $g_i$.

**Assumption 4.** For every $Z^i$ there exists a constant $\epsilon_i > 0$ such that $B(0, \epsilon_i) \subseteq Z^i$.

**Assumption 5.** There exists a constant $\bar{\mu}_i > 0$ such that $-g_i(x, 0) \leq \bar{\mu}_i$ for any feasible $x$ of problem (3).

Assumption 4 is a standard RO assumption that the uncertainty set is full dimensional, note that this is always true, since under a linear transformation we can always reduce the dimension of $Z^i$. Assumption 5 states the following: the feasible set of the robust problem does not contain rays that make any of the constraints of the ‘nominal problem’ (where $z = 0$) be arbitrarily satisfied,
i.e., make \( g_i(x,0) \) arbitrarily small. We note that this assumption can be verified by checking the following sufficient condition:

\[
\min \{ g_i(x,0) : x \in X, g_j(x,0) \leq 0, \forall j \in [m] \setminus \{i\} \} > -\infty, \forall i \in [m],
\]

which can, in turn, be shown to hold by solving \( m \) convex (non-robust) optimization problems.

In the course of the SGSP algorithm, one needs to compute the subgradients of the perspective functions \( \tilde{g}_i(x,u) \). Ideally, this is done using the subgradients of the original functions \( g_i(x,z) \), which should typically be available. The following lemma provides a ‘recipe’ for doing exactly this. The ‘recipe’ is based on the convex analysis results for perspective functions of Combettes and Müller (2018).

**Lemma 1.** Let \( x \in X \) and \( u_i = (\tilde{z}_i, \lambda_i) \in U^i \) for all \( i \in [m] \).

(i) If \( v_x \in \partial_x \tilde{L}(x,u) \), \( v_i \in \partial_{u_i} (-\tilde{L}(x,u)) \), then they are of the following form

\[
v_x = c + d_x \quad \text{(13)}
\]

\[
v_i = \left\{ \begin{array}{ll}
(d_i, -g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) - \frac{\tilde{z}_i^\top d_i}{\lambda_i}), & \lambda_i > 0, \\
(d_i, \phi_i), & \text{otherwise}
\end{array} \right. \quad \text{(14)}
\]

where \( d_x \in \sum_{i: \lambda_i > 0} \lambda_i \partial_x g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) \), \( d_i \in \partial_{z_i} \left( -g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) \right) \) for all \( i \in [m] \) such that \( \lambda_i > 0 \), and \( d_i \in \bigcup_{z_i \in \mathbb{R}^d} \left( \partial_{z_i} (-g_i(x,z_i)) \right) \), \( \phi_i + (-g_i)^*(x,d_i) \leq 0 \) for all \( i \in [m] \) such that \( \lambda_i = 0 \).

(ii) Let \( z_i = \tilde{z}_i / \lambda_i \) if \( \lambda_i > 0 \) and \( z_i = 0 \) otherwise, and let \( \tilde{d}_{x,i} \in \partial_x g_i(x,z_i) \) and \( \tilde{d}_{z,i} \in \partial_{z_i} (-g_i(x,z_i)) \) for all \( i \in [m] \). Then,

\[
v_x = c + \sum_{i=1}^m \lambda_i d_{x,i} \in \partial_x \tilde{L}(x,u) \quad \text{and} \quad v_i = (d_{z,i}, -g_i(x,z_i) - (z_i)^\top d_{z,i}) \in \partial_{u_i} (-\tilde{L}(x,u)).
\]

### 3.3.3. Computing the projections

In this section, we discuss the projections needed to apply SGSP and CP. Specifically, we will discuss how to project on set \( U^i \) and \( \tilde{U}^i \), which are more complicated than \( Z^i \). We shall assume that the projection on set \( Z^i \) is simple and formulate the projection on \( U^i \) in its terms. Note that, in this paper, we use \( U^i \)

\[
U^i = \{ (\tilde{z}_i, \lambda_i) : \tilde{z}_i \leq \lambda_i Z^i, \lambda_i \geq 0 \}. \quad \text{(15)}
\]

for the CP setting, and \( \tilde{U}^i \)

\[
\tilde{U}^i = \{ (\tilde{z}_i, \lambda_i) : \tilde{z}_i \leq \lambda_i Z^i, 0 \leq \lambda_i \leq \lambda \}, \quad \text{(16)}
\]

which uses an additional upper bound on \( \lambda_i \) for the more general SGSP setting. We will start by showing a general way to compute the projection over \( U^i \).
The projections onto their conic extensions can also be computed in \( O \) obtained either analytically or in \( O \) that satisfies the appropriate equality constraint.

\[
\begin{align*}
\{z: \|z\|_2 \leq 1\} & \quad \begin{cases} y & \text{if } y \in Z \\ \text{otherwise} & \end{cases} \quad \mu^* = \max \left\{ \min \left\{ \frac{1+\|y\|_2}{2}, \lambda \right\}, 0 \right\}^2 \\
\{z: \|z\|_\infty \leq 1\} & \quad \min \{\|e\|, |y|\} \circ \text{sign}(y) \quad \mu^* = \min \left\{ \sum_{i \leq j} \frac{\|y_{(j)}\|_{\infty} + \lambda}{j+1}, \lambda \right\} \quad j^* = \max \left\{ j \in [d]: |q_{(j)}| \geq \sum_{i \leq j} \frac{\|y_{(j)}\|_{\infty} + \lambda}{j+1} \right\}^1.
\end{align*}
\]

Table 3 illustrates three examples, the \( \ell_2, \ell_\infty, \ell_1 \) balls, for which the projection onto \( Z^i \) is obtained either analytically or in \( O(n) \), and similarly the value of the optimal \( \mu^* \), and therefore the projections onto their conic extensions can also be computed in \( O(n) \).

Proposition 3. The projection of \( u_i = (\bar{z}_i, \lambda_i) \in \mathbb{R}^{d_i+1} \) on the set \( U^i \) defined by (15) is given by

\[
P_{U^i}(u_i) = \begin{cases} u_i & \text{if } u_i \in U^i \\ \left( \mu^* P_{Z^i}(\frac{\bar{z}_i}{\mu^*}), \mu^* \right) & \text{if } u_i \notin U^i, \sigma_{Z^i}(\bar{z}_i) > -\lambda_i \\ 0 & \text{otherwise} \end{cases}
\]

where \( \mu^* > 0 \) is the unique solution of

\[
\mu \left\| P_{Z^i}(\frac{\bar{z}_i}{\mu}) \right\|^2 - \bar{z}_i^T P_{Z^i}(\frac{\bar{z}_i}{\mu}) + \mu - \lambda_i = 0.
\]

The same technique used to prove Proposition 3 can be used for the proof of projection over \( \tilde{U}^i \).

Corollary 1. The projection of \( u_i = (\bar{z}_i, \lambda_i) \in \mathbb{R}^{d_i+1} \) on the set \( \tilde{U}^i \) defined by (16) is given by

\[
P_{\tilde{U}^i}(u_i) = \left( \mu^* P_{Z^i}(\frac{\bar{z}_i}{\mu^*}), \mu^* \right)
\]

where \( \mu^* = \max\{\min\{\tilde{\mu}, \tilde{\lambda}\}, 0\} \) with \( \tilde{\mu} \) being the unique solution of

\[
\tilde{\mu} \left\| P_{Z^i}(\frac{\bar{z}_i}{\tilde{\mu}}) \right\|^2 - \bar{z}_i^T P_{Z^i}(\frac{\bar{z}_i}{\tilde{\mu}}) + \tilde{\mu} - \lambda_i = 0,
\]

and \( 0P_{Z^i}(\frac{\bar{z}_i}{0}) = 0 \).

Proposition 3 and Corollary 1 suggest that we can always obtain the projection onto sets \( U^i \) and \( \tilde{U}^i \), which are the conic extension of \( Z^i \), by applying a bi-section procedure to find the value of \( \mu \) that satisfies the appropriate equality constraint.

Table 3 illustrates three examples, the \( \ell_2, \ell_\infty, \ell_1 \) balls, for which the projection onto \( Z^i \) is obtained either analytically or in \( O(n) \), and similarly the value of the optimal \( \mu^* \), and therefore the projections onto their conic extensions can also be computed in \( O(n) \).
4. Convergence results

4.1. EB algorithms

To unify analysis for all forms of the Lagrangian function and the algorithms presented in this paper, we define now sufficient properties needed for an algorithm to prove the feasibility and optimality convergence. These sufficient properties will take the form of the algorithm being an *ergodically bounded* (EB) algorithm, as the following definition states.

**Definition 1.** Let $A$ be an iterative algorithm to solve the saddle point problem (3). We call the algorithm $A$ *ψ-ergodic bounded* ($\psi$-EB) if there exist constant scalars $\phi, \tau, \beta \geq 0$, $\theta_i, \theta_w \in \mathbb{R}_+$, and a function $\psi(N): \mathbb{N} \to \mathbb{R}_+$ with $\psi(N) \downarrow 0$ as $N \to +\infty$, such that for any initial point $(x^0; \lambda^0, w^0)$ the algorithm $A$ generates a sequence $\{(x^k, \lambda^k, w^k)\}_{k \in \mathbb{N}}$ satisfying

$$L(\bar{x}^N, (\lambda, w)) - L(\bar{x}^*, (\lambda^*, w^*)) + \alpha \text{Dist}(\bar{x}^N, X) \leq \psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}}{\tau} + \sum_{i=1}^{m} \theta_i^{-1} \max\{\lambda_i, \lambda^0_i\}^2 + \theta_w^{-1} \max\{\|w\|, \|w^0\|\}^2 + \phi(1 + \alpha^2) \right),$$

and

$$L(\bar{x}^N; \lambda^*, w^*) - L(x^*; \lambda^*, w^*) + \beta \text{Dist}(\bar{x}^N, X) \geq 0.$$ 

for all iterations $N \in \mathbb{N}$, dual variables $\lambda \in \mathbb{R}_+^m$, $w \in \mathbb{R}^r$, parameters $\alpha \geq 0$ and saddle points $(x^*, (\lambda^*, w^*))$ of (5).

We note that this definition allows to consider algorithms which generate sequences $\{x^k\}_{k \in \mathbb{N}}$ such that $x^k \notin X$. In this case, the convergence result to a feasible solution also considers the rate of convergence to $X$. The following theorem shows that the ergodic sequence of any $\psi$-EB algorithm converges to feasibility and optimality at the rate at which $\psi$ converges to zero.

**Theorem 3.** Let problem (3) satisfy Assumption 1 and 2. Let $A$ be an $\psi$-EB algorithm to solve the saddle point problem (3) with parameters $\phi, \tau, \beta$ and $\theta$. For a given starting point $(x^0, (\lambda^0, w^0)) \in \mathbb{R}^n \times \mathbb{R}_+^m \times \mathbb{R}^p$ let $\bar{x}^N$ be the ergodic primal sequence generated by the algorithm.

Then, for any optimal solution $x^*$ to (3) we have that

$$\sum_{i=1}^{m} [f_i(\bar{x}^N)]_+ + \|Ax^N - b\| + \text{Dist}(\bar{x}^N, X) \leq \psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}}{\tau} + \sum_{i=1}^{m} \frac{\tilde{\lambda}_i + 1}{\theta_i} \max\{\lambda_i, \lambda^0_i\}^2 + \frac{\theta_w}{\tilde{\lambda}_w} \max\{R_w + 1, \|w^0\|\}^2 + \phi(1 + (1 + \beta)^2) \right),$$

and

$$|c^\top(\bar{x}^N - x^*)| \leq \psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}}{\tau} + \sum_{i=1}^{m} \frac{2\tilde{\lambda}_i}{\theta_i} \max\{2\lambda_i, \lambda^0_i\}^2 + \frac{\theta_w}{\tilde{\lambda}_w} \max\{2R_w, \|w^0\|\}^2 + \phi(1 + 4\beta^2) \right).$$
Proof. Let $\kappa_1 > 0$ and let $\mathbf{r} \in \mathbb{R}^m_+$ be a multiplication of indicator vector of constraint violations by $\kappa_1$, i.e.:

$$r_i = \begin{cases} \kappa_1, & f_i(\bar{x}^N) > 0, \\ 0, & \text{otherwise,} \end{cases}, \quad i \in [m]$$

and let $\lambda = \lambda^* + \mathbf{r}$ and let $\kappa_2 > 0$ and $\mathbf{w} = \mathbf{w}^* + \kappa_2 \frac{(A\bar{x}^N - \mathbf{b})}{\|A\bar{x}^N - \mathbf{b}\|}$ where $(x^*; \lambda^*, \mathbf{w}^*)$ is a saddle point of (4). Then,

$$L(\bar{x}^N; \lambda, \mathbf{w}) = \mathbf{c}^T \bar{x}^N + \sum_{i=1}^{m} \lambda_i f_i(\bar{x}^N) + \mathbf{w}^T (A\bar{x}^N - \mathbf{b})$$

$$= \mathbf{c}^T \bar{x}^N + \sum_{i=1}^{m} (\lambda_i^* + r_i) f_i(\bar{x}^N) + \mathbf{w}^* (A\bar{x}^N - \mathbf{b}) + \kappa_2 \|A\bar{x}^N - \mathbf{b}\|$$

$$= \mathbf{c}^T \bar{x}^N + \sum_{i=1}^{m} \lambda_i^* f_i(\bar{x}^N) + \kappa_1 \sum_{i=1}^{m} [f_i(\bar{x}^N)]_+ + \mathbf{w}^* (A\bar{x}^N - \mathbf{b}) + \kappa_2 \|A\bar{x}^N - \mathbf{b}\|$$

$$= L(\bar{x}^N; \lambda^*, \mathbf{w}^*) + \kappa_1 \sum_{i=1}^{m} [f_i(\bar{x}^N)]_+ + \kappa_2 \|A\bar{x}^N - \mathbf{b}\|$$

(17)

Since $\mathcal{A}$ is an $\psi$-EB algorithm then

$$L(\bar{x}^N; \lambda^*, \mathbf{w}^*) + \beta \text{Dist}(\bar{x}^N, X) \geq \mathbf{c}^T x^* = L(x^*; \lambda^*, \mathbf{w}^*),$$

(18)

and for all $\kappa_3 > 0$

$$L(\bar{x}^N; \lambda, \mathbf{w}) - L(x^*; \lambda^*, \mathbf{w}^*) + \kappa_3 \text{Dist}(\bar{x}^N, X) \leq$$

$$\psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}^2}{\tau} + \sum_{i=1}^{m} \frac{\max\{\lambda_i, \lambda_i^0\}^2}{\theta_i} + \frac{\max\{\|\mathbf{w}\|, \|\mathbf{w}^0\|\}^2}{\theta_w} + \phi(1 + \kappa_3^2) \right).$$

(19)

Since Assumptions 1 and 2 hold, it follows from Proposition 2 that $\lambda_i^* \leq \bar{\lambda}$ and thus

$$\max\{\lambda_i, \lambda_i^0\} = \max\{\lambda_i^* + r_i, \lambda_i^0\} \leq \max\{\bar{\lambda} + \kappa_1, \lambda_i^0\}.$$ 

Similarly, since $\|\mathbf{w}^*\| \leq R_w$, thus,

$$\max\{\|\mathbf{w}\|, \|\mathbf{w}^0\|\} = \max\left\{ \|\mathbf{w}^* + \kappa_2 \frac{(A\bar{x}^N - \mathbf{b})}{\|A\bar{x}^N - \mathbf{b}\|}, \|\mathbf{w}^0\| \right\}$$

$$\leq \max\{\|\mathbf{w}^*\| + \kappa_2, \|\mathbf{w}^0\|\}$$

$$\leq \max\{R_w + \kappa_2, \|\mathbf{w}^0\|\}.$$ 

(20)

Combining inequalities (17)-(20) we obtain

$$\kappa_1 \sum_{i=1}^{m} [f_i(\bar{x}^N)]_+ + \kappa_2 \|A\bar{x}^N - \mathbf{b}\| + (\kappa_3 - \beta) \text{Dist}(\bar{x}^N, X)$$

$$\leq \psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}^2}{\tau} + \sum_{i=1}^{m} \frac{\max\{\bar{\lambda} + \kappa_1, \lambda_i^0\}^2}{\theta_i} + \frac{\max\{R_w + \kappa_2, \|\mathbf{w}^0\|\}^2}{\theta_w} + \phi(1 + \kappa_3^2) \right).$$

(21)
Setting $\kappa_1 = \kappa_2 = 1$ and $\kappa_3 = \beta + 1$ we obtain the first result. For the second result, we need to upper and lower bound $c^T (\bar{x}^N - x^*)$. First, we have

$$c^T (\bar{x}^N - x^*) = L(\bar{x}^N; 0, 0) - L(x^*; \lambda^*, w^*).$$

(22)

where the equalities follow from $L(x^*; \lambda^*, w^*) = c^T x^*$ and $L(\bar{x}^N; 0, 0) = c^T \bar{x}^N$. Using (19) with $\kappa_3 = 0$ we obtain

$$L(\bar{x}^N; 0, 0) - L(x^*; \lambda^*, w^*) \leq \psi(N) \left( \frac{\max\{\|x^*\|, \|x^0\|\}^2}{\tau} + \sum_{i=1}^{m} (\lambda_i^0)^2 + \|w^0\|^2 \theta_w + \phi \right).$$

(23)

Combining (22) with (23) we have that

$$c^T (\bar{x}^N - x^*) \leq \psi(N) \left( \tau^{-1}\|x^* - x^0\|^2 + \sum_{i=1}^{m} \theta_i^{-1}(\lambda_i^0)^2 + \theta_w^{-1}\|w^0\|^2 + \phi \right).$$

(24)

To obtain the other side of the bound, we use (18) to obtain

$$c^T (x^* - \bar{x}^N) - \sum_{i=1}^{m} \lambda_i^* f_i(\bar{x}^N) - (w^*)^T (A\bar{x}^N - b) = L(x^*; \lambda^*, w^*) - L(\bar{x}^N; \lambda^*, w^*) \leq \beta \text{Dist}(\bar{x}^N, X),$$

which in turn implies

$$c^T (x^* - \bar{x}^N) \leq \sum_{i=1}^{m} \lambda_i^* f_i(\bar{x}^N) + (w^*)^T (A\bar{x}^N - b) + \beta \text{Dist}(\bar{x}^N, X)$$

$$\leq \lambda \sum_{i=1}^{m} [f_i(\bar{x}^N)]_+ + R_w \|A\bar{x}^N - b\| + \beta \text{Dist}(\bar{x}^N, X),$$

(25)

where the last inequality follows from the the fact that $0 \leq \lambda_i^* \leq \bar{\lambda}$ and $\|w^*\| \leq R_w$. Using (21) with $\kappa_1 = \bar{\lambda}$, $\kappa_2 = R_w$, and $\kappa_3 = 2\beta$ to bound (25) we obtain that

$$c^T (x^* - \bar{x}^N) \leq \psi(N) \left( \max\{\|x^*\|, \|x^0\|\}^2 + \sum_{i=1}^{m} \max\{2\bar{\lambda}, \lambda_i^0\}^2 + \max\{2R_w, \|w^0\|\}^2 + \phi(1 + (2\beta)^2) \right).$$

(26)

Combining (24) and (26) we obtain the desired result.

\[\square\]

4.2. Generalized model

So far, we kept the problem formulations simple and stated the convergence results without proofs. In this section, we shall show the convergence of both the SGSP and CP algorithms for a more general model using the EB-algorithm definition.

The general model is needed to tackle the fact that projections onto sets $X$ or $Z^i$ are the main tools used in both algorithms, and therefore these projections should be simple. Two issues that might arise is that either the primal domain $X$ or at least one of the sets $Z^i$ is not ‘simple’, but instead, it is an intersection of several simple sets.
Example 1. One of the popular uncertainty sets in RO is the so-called budgeted uncertainty set, formulated as:

\[ Z = \{ z : -1 \leq z \leq 1, \|z\|_1 \leq \Gamma \} \]

which is an intersection of two simple sets: the \( \ell_\infty \) norm and \( \ell_1 \) norm balls.

Our strategy for dealing with these complex sets shall be to disentangle the projections onto the intersected sets. We will achieve this by including ‘copies’ of the respective primal or dual variables, together with the relevant equality constraints, which are to be relaxed in the Lagrangian. The saddle-point algorithm would then be applied to the new Lagrangian problem.

Consider, for example, the case where \( X = \bigcap_{j=1}^q X_i \) is an intersection of several ‘simple’ sets. Formulation 3 is ready to handle this situation easily. One can ‘expand’ the vector \( x \) by having \( q \) copies of it: \( x \mapsto [x_1, \ldots, x_q] \). In the next step, each inequality constraint in the problem is made dependent only on one of the \( x_i \)’s and at the same time, equality constraints \( x_1 = x_i, i = 2, \ldots, q \) become embedded in the \( Ax = b \) system (where the rank condition is easily verified).

Similarly, we can consider the case where \( Z^i = \bigcap_{l=1}^{s_i} Z^i_{l}, \) \( i = 1, \ldots, m \) with \( Z^i_{l} \) being compact convex sets. In this case, the constraint \( z_i \in \bigcap_{l=1}^{s_i} Z^i_{l} \) can be written using concatenated uncertain parameter vector \( \hat{z}_i = (z_{i,1}, \ldots, z_{i,s_i}) \), defining

\[ \hat{Z}^i = \{(z_{i,1}, \ldots, z_{i,s_i}) \in Z_{i,1} \times \ldots \times Z_{i,s_i} : z_{i,l} = z_{i,s_i}, l = 1, \ldots, s_i - 1 \} , \]

and formulating the constraints as

\[ \max_{\hat{z}_i \in \hat{Z}^i} \hat{g}_i(x_1, \hat{z}_i) \equiv g_i(x_1, z_{i,s_i}) \leq 0 \]

where \( Z_{i,l}, l = 1, \ldots, s_i \) are ‘simple’ sets on which it is easy to project. Thus, the general model we are going to address is the following:

\[
\min_{x \in X} \mathbf{c}^\top x \\
\text{s.t. } f_i(x) := \max_{z_i \in Z^i} g_i(x, z_i) \leq 0 \quad i \in [m] \\
Ax = b
\]  

(27)

where \( Z^i = \bigcap_{l=1}^{s_i} Z^i_{l} \). For both SGSP and CP we shall give the problem generalizations together with the expanded Lagrangian that is to be solved.

Consider problem (27) where we need to formulate a saddle point problem that disentangles different components of the \( x \) and \( z_i \). We start by considering a saddle point \( (x^*, (u^*, w^*)) \) of \( \hat{L}(x, (u, w)) \) over the sets \( x \in X \) and \( u_i \in U_i \), which we already proved exists. Thus, by the definition of the saddle point, we have that

\[ \tilde{g}_i(x^*, u_i^*) = \sup_{u_i \in U^i} \tilde{g}_i(x^*, u_i) \leq \sup_{u_i \in U^i, u_i,s_i = u_i,l \in [s_i-1]} \tilde{g}_i(x^*, u_i,s_i) \]
where the last equality follows from the definition of $U^i = \cap_{i \in [s_i]} U^{i,l}$ where $U^{i,l} = \{ (\tilde{z}, \lambda_i) : \tilde{z} \in \lambda_i \mathcal{Z}^{i,l} \}$, similarly to what was explained in the previous section. Note that the leftmost supremum has a solution $(u_{i,1} = u_{i,2} = \ldots = u_{i,s_i} = u_i^*)$. Moreover, since $U^i \subseteq U^{i,l}$ for $l \in [s_i]$, the sets $U^{i,l}$ have a nonempty interior, therefore, dualizing the equality constraints, we have that strong duality holds. Thus, denoting $\tilde{u}_i = (u_{i,1}, \ldots, u_{i,s_i}) \equiv (u_i^*, \ldots, u_i^*)$ there exists $\omega_i^* = (\omega_{i,1}^*, \ldots, \omega_{i,s_i}^*)$ such that $(\omega_i^*, \tilde{u}_i^*)$ satisfies

\[
\tilde{g}_i(x^*, u_i^*) = \sup_{u_{i,l} \in U^{i,l}, \ u_{i,s_i} = u_{i,l}, \ i \in [s_i]} \tilde{g}_i(x^*, u_{i,s_i}) = \sup_{\omega_i,l \in \Omega^{i,l}, \ i \in [s_i]} \sup_{u_{i,l} \in U^{i,l}, \ i \in [s_i]} g_i(x^*, u_{i,l}) + \sum_{l=1}^{s_i-1} \omega_i^T (u_{i,l} - u_{i,s_i})
\]

\[
= \omega_{i,l} \sup_{u_{i,s_i} \in U^{i,l}, i \in [s_i]} \tilde{g}_i(x^*, u_{i,s_i}) + \sum_{l=1}^{s_i-1} \omega_i^T (u_{i,l} - u_{i,s_i})
\]

\[
= \sup_{u_{i,l} \in U^{i,l}, \ i \in [s_i]} \tilde{g}_i(x^*, u_{i,l}) + \sum_{l=1}^{s_i-1} (\omega_i^*)^T (u_{i,l} - u_{i,s_i})
\]

(28)

Due to Proposition 2 we know that we can restrict $U^i$ to $\tilde{U}^i$, and thus, the existence of the saddle point in this case follows the same logic where $U^{i,l}$ is replaced by $\tilde{U}^{i,l} = \{ u_i \equiv (\tilde{z}_{i,l}, \lambda_i) \in U^{i,l} : \lambda_i \leq \tilde{\lambda} \}$. In the following, we will show that we can restrict the domain over which we optimize $\omega$ and still retrieve a saddle point of the original problem.

Denoting $\chi = (x, \omega)$ and $y = (\tilde{u}, w)$ where $\omega = (\omega_1, \ldots, \omega_m)$, $\tilde{u} = (\tilde{u}_1, \ldots, \tilde{u}_m)$, and $\omega_i = (\omega_{i,1}, \ldots, \omega_{i,s_i}^*)$, $\tilde{u}_i = (u_{i,1}, \ldots, u_{i,s_i})$ for all $i \in [m]$, we can define a saddle point in the lifted space. Indeed, let $(x^*, (\lambda^*, w^*))$ be a saddle point of $L(x, (\lambda, w))$, then, by Proposition 1, and the reasoning leading to (28) above, there exists $\tilde{u}^*$ and $\omega^*$ such that

\[
\sup_{\lambda \geq 0, w} L(x^*, (\lambda, w))
\]

\[
= \sup_{u_{i} \in U^{i,l}, i \in [m]} c^T x^* + w^T (Ax^* - b) + \sum_{i=1}^{m} \tilde{g}_i(x^*, u_{i})
\]

\[
= \sup_{u_{i} \in U^{i,l}, i \in [m]} c^T x^* + w^T (Ax^* - b) + \sum_{i=1}^{m} \tilde{g}_i(x, u_{i,s_i}) + \sum_{i=1}^{m} \sum_{l=1}^{s_i-1} (\omega_i^*)^T (u_{i,l} - u_{i,s_i})
\]

and

\[
\inf_{x \in \mathcal{X}} L(x, (\lambda^*, w^*))
\]

\[
= \inf_{x \in \mathcal{X}} c^T x + (w^*)^T (Ax - b) + \sum_{i=1}^{m} \tilde{g}_i(x, u_{i}^*)
\]

\[
= \inf_{\omega_{i,l} \in [m], l \in [s_i-1]} c^T x^* + (w^*)^T (Ax - b) + \sum_{i=1}^{m} \tilde{g}_i(x, u_{i,s_i}) + \sum_{i=1}^{m} \sum_{l=1}^{s_i-1} (\omega_i^*)^T (u_{i,l}^* - u_{i,s_i}^*)
\]
Defining
\[
\tilde{L}(\chi, y) := c^T x + w^T (Ax - b) + \sum_{i=1}^m \tilde{g}_i(x, u_{i,s_i}) + \sum_{i=1}^m \sum_{l=1}^{s_i-1} \omega_{i,l}^T (u_{i,l} - u_{i,s_l}).
\] (29)
we obtain that \((\chi^*, y^*)\) is a saddle point of \(\tilde{L}\). Following similar logic, we can also obtain that given a saddle point \((\chi^*, y^*)\) of \(\tilde{L}\) with \(u_{i,s_l}^* = (\tilde{x}_{i,s_l}^*, \lambda_{i,s_l}^*)\) we can obtain a saddle point \((\chi^*, (\lambda^*, w^*))\) of \(L\) by defining \(\lambda_i^* = \lambda_{i,s_l}^*, \lambda^* = (\lambda_1^*, \ldots, \lambda_m^*)\).

Since we proved that \(\tilde{L}\) has a saddle point, we can now show that SGSP algorithm applied to (29) meets the EB-algorithm assumptions.

However, to run the algorithm and prove its convergence, we need to show that the feasible sets of the variables can be restricted without losing a saddle point. Legitimacy of bounding \(u_{i,l}\) and \(w\) follows from Proposition 2 and (28). The following proposition establishes our ability to bound \(\omega_{i,l}\).

**Proposition 4.** Let Assumptions 1–5 hold. Let \(i \in [m]\) and consider the saddle point problem
\[
\min_{\omega_i} \max_{\mathbf{u}_{i,l} \in \Omega_{i,l}, \mathbf{t} \in [s_i], \mu_i \in \mathcal{M}} \tilde{g}_i(x^*, u_{i,s_i}) + \sum_{l=1}^{s_i-1} \omega_{i,l}^T (u_{i,l} - u_{i,s_l})
\] (30)
where \(x^*\) is an optimal solution of (27), and \(0 \leq \lambda \leq \bar{\lambda} \leq \hat{\lambda}\). Then, there exists a saddle point \((\omega_i^*, \bar{u}_i^*)\) with \(\omega_i^* = (\nu_{i,l}^*, \mu_{i,l}^*)\) such that \(\omega_{i,l}^*\) is contained in the set
\[
\Omega_{i,l} \defeq \left\{ \omega_{i,l} = (\nu_{i,l}^*, \mu_{i,l}^*) : -\bar{\mu}_i \leq \mu_{i,l} \leq 0, \|\nu_{i,l}\| \leq \frac{-\mu_{i,l}}{\epsilon_i}, i \in [m], l \in [s_i-1] \right\},
\]
where \(\epsilon_i\) and \(\bar{\mu}_i\) is given by Assumptions 4 and 5, respectively.

Now that we have shown that we can bound all variables in the saddle point of function \(\tilde{L}(\chi, y)\) over the sets \(\mathcal{X} = \mathcal{X} \times (x_{i=1}^m \times x_{l=1}^{s_i-1} \Omega_{i,l})\) and \(\mathcal{Y} = (x_{i=1}^m \times x_{l=1}^{s_i-1} \Omega_{i,l}) \times W\), we can apply SGSP to the problem with these bounded sets as follows.

**SGSP for \(\tilde{L}\)**

**Input:** \(\tau > 0, \theta_i > 0, \theta_w > 0\) for \(i \in [m]\), and \(N \in \mathbb{N}\)

**Initialization.** Initialize \(\chi^0 \in \mathcal{X}\), and \(y^0 \in \mathcal{Y}\) such that \(\lambda_{i,l}^0 = \lambda_{i,l}^0\) for all \(l \in [s_i], i \in [m]\), and define the diagonal matrix \(\Theta = \text{Diag}(\theta_1; \ldots; \theta_m; \theta_w e)\).

**General step:** For \(k \in [N]\)

- Compute subgradients \(v_{x,k-1}^i = (v_{x,k-1}, v_{\omega,k-1}) \in \partial_{x} \tilde{L}(x_{k-1}^i, y_{k-1}^i), \) and for all \(i \in [m]\)
  \(v_{\omega,k-1}^i = (v_{\omega,k-1}^i, \ldots, v_{\omega,k-1}^i) \in \partial_{\omega,\omega} \tilde{L}(x_{k-1}^i, y_{k-1}^i)\).

- \(x^k = P_{\mathcal{X}}(x_{k-1}^i - \tau v_{x,k-1}^i) \Longleftrightarrow x^k = P_{\mathcal{X}}(x_{k-1}^i - \tau v_{x,k-1}^i), \)
- \(w_{i,l}^k = P_{\Omega_{i,l}}(w_{i,l}^k - \tau (u_{i,l}^k - u_{i,s_l}^k)), i \in [m], l \in [s_i-1] \)
- \(u_{i,s_l}^k = P_{\Omega_{i,s_l}}(u_{i,s_l}^k - \tau v_{i,s_l}^k), i \in [m], l \in [s_i-1] \)
- \(w_{i,s_l}^k = P_{\Omega_{i,s_l}}(w_{i,s_l}^k - \tau v_{i,s_l}^k), i \in [m] \)

- \(y^k = P_{\mathcal{Y}}(y_{k-1}^i - \Theta v_{y,k-1}^i) \Longleftrightarrow y^k = P_{\mathcal{Y}}(y_{k-1}^i - \Theta v_{y,k-1}^i), \)
- \(u_{i,l}^k = P_{\Omega_{i,l}}(u_{i,l}^k + \tau (w_{i,l}^k - w_{i,s_l}^k)), i \in [m], l \in [s_i-1] \)
- \(w_{i,s_l}^k = P_{\Omega_{i,s_l}}(w_{i,s_l}^k + \tau v_{i,s_l}^k), i \in [m] \)
- \(w^k = P_{W}(w_{k-1}^i + \tau_w (Ax_{k-1}^i - b))\).
We will now prove that SGSP is an EB-algorithm and thus, it converges to the optimal solution.

**Proposition 5.** Let Assumptions 1-5 hold. And let \( \{ (x^k, y^k) \}_{k \in \mathbb{N}} \) be the sequence generated by SGSP algorithm with parameters \( \tau = \frac{\epsilon}{\sqrt{N}}, \theta_i = \frac{\delta_i}{\sqrt{N}}, \theta_w = \frac{\delta_w}{\sqrt{N}} \), where \( x^k = (x^k, \omega^k) \) and \( y^k = (\tilde{u}^k, w^k) \). Then,

\[
L(\tilde{x}^N, y) - L(x^*, \tilde{y}^N) \leq \frac{1}{2\sqrt{N}} \left( \frac{\|x^0 - x^*\|}{\tau} + \bar{\tau}G_x^2 + \sum_{i=1}^m \left( \frac{\|\tilde{u}_i^0 - \tilde{u}_i\|}{\theta_i} + \tilde{\theta}_i G_{\tilde{u}_i}^2 \right) + \frac{\|w^0 - w\|}{\theta_w} + \tilde{\theta}_w G_w^2 \right)
\]

where

\[
G_x := \|c\| + \|A\| R_w + \sum_{i \in [m]} \bar{\lambda} G_{x,i} + \bar{\lambda} \sum_{i \in [m]} \left( (s_i - 1)(2 + R_{i,s_i}) + \sum_{l \in [s_i - 1]} R_{i,l} \right)
\]

\[
G_{\tilde{u}_i} := (\bar{\lambda} + R_{i,s_i}) g_{x,i} + \bar{g}_i + 2\tilde{\mu}_i(s_i - 1) \left( 1 + \frac{1}{\epsilon_i} \right)
\]

\[
G_w := \|A\| R_x + \|b\|
\]

Moreover, SGSP is an EB-algorithm. More precisely, let \( x^* \) an optimal solution of (27), \( (\lambda^*, w^*) \) be the optimal dual variables associated with its constraints, and let \( \lambda \in \mathbb{R}_m^m \), and \( w \in \mathbb{R}^r \) such that \( \lambda_i \equiv \lambda_{i,s_i} \) for all \( i \in [m] \). Then,

\[
L(\tilde{x}^N, (\lambda, w)) - L(x^*, (\lambda^*, w^*)) \leq \frac{1}{2\sqrt{N}} \left( \bar{\tau}^{-1}\|x^* - x^0\|^2 + \sum_{i=1}^m \sigma_i \bar{\theta}_i^{-1} \max\{\lambda_i, \lambda^0_i\}^2 + 2\tilde{\theta}_w^{-1} \max\{\|w\|, \|w_0\|\}^2 + \phi \right),
\]

and

\[
L(\tilde{x}^N; (\lambda^*, w^*)) \geq c^T x^*.
\]

where

\[
\sigma_i = \sum_{l=1}^{s_i} (1 + 4R_{i,l}^2), \quad \phi = \bar{\tau} G_x^2 + \sum_{i=1}^m \bar{\theta}_i G_{\tilde{u}_i}^2 + 4 \left( \sum_{i=1}^m \tilde{\mu}_i(s_i - 1) \left( 1 + \frac{1}{\epsilon_i} \right) \right)^2 \bar{\tau} + \tilde{\theta}_w G_w^2
\]

where \( R_x \) is the radius of \( X \), \( R_w \) the radius of \( W \), \( R_{i,l} \) is the radius of \( Z_{i,l} \), \( \bar{g}_i = \max_{x \in X, z_i \in Z_i} |g_i(x, z_i)| \), \( \bar{\lambda}, \bar{\mu}_i \) and \( \epsilon_i \) are defined in Proposition 2, Assumption 5 and Assumption 4, respectively, and \( G_{x,i}, G_{z,i} \) are the bounds on the subgradients \( d^k_{x,i}, d^k_{z,i} \) generated as in Lemma 1.

In the case of CP, we proceed similarly to SGSP with one change – as the CP algorithm presumes bilinearity, we assume \( g_i \) are of the form given in (9), we obtain:

\[
\tilde{L}(\chi, y) = x^T \left( c + \sum_{i \in [m]} \tilde{Q}_i u_{i,s_i} + A^T w \right) - b^T w + \delta_X(x)
\]

\[
+ \sum_{i \in [m]} \left( \tilde{q}_i^T u_{i,s_i} + \sum_{l \in [s_i - 1]} \tilde{w}_{i,l}^T (u_{i,l} - u_{i,s_i}) - \sum_{l \in [s_i]} \delta_{U}^i(u_{i,l}) \right) - b^T w
\]
In this case we can define $B$ as the matrix
\[
B = \begin{bmatrix}
A & 0 & 
\overline{Q}_1 & 0 & 
\overline{Q}_2 & \ldots & 0 & 
\overline{Q}_m \\
0 & I & 0 & \ldots & 0 \ \\
0 & 0 & I & \ldots & 0 \ \\
\vdots & \vdots & \vdots & \ddots & \vdots \ \\
0 & 0 & 0 & \ldots & I & -I \\
\end{bmatrix},
\]
and apply CP to generalized problem as follows.

**CP for $\tilde{L}$**

**Input:** $\tau > 0$, $\theta \in (0, 1/\|B\|^2)$, and $N \in \mathbb{N}$

**Initialization.** Initialize $\omega^0 = 0 \in \mathbb{R}^\Sigma_{i \in [m]}(s_i - 1)$, $x^0 = X$, $\chi^0 = (x^0, \omega^0)$, $\tilde{\chi}^0 = (\tilde{x}^0, \tilde{\omega}^0) := (\chi^0, \tilde{\omega}^0)$, $y^0 = (u^0, w^0) \in \times_{i \in [m]} \times_{t \in [s_i]} U^{x, y} \times \mathbb{R}^r$.

**General step:** For $k \in [N]$

\[
\begin{align*}
    u_{i,l}^k &= P_{U_{i,l}} \left( u_{i,l}^{k-1} + \theta \tilde{\omega}_{i,l}^{k-1} \right) \quad i \in [m], l \in [s_i - 1] \\
    u_{i,s_i}^k &= P_{U_{i,s_i}} \left( u_{i,s_i}^{k-1} + \theta \left( \tilde{q}_0 - \sum_{l=1}^{s_i-1} \tilde{\omega}_{i,l}^{k-1} \right) \right) \quad i \in [m] \\
    w^k &= w^{k-1} + \theta (Ax^{k-1} - b) \\
    x^k &= P_X \left( x^{k-1} - \tau \left( c + A^\top w^k - \sum_{l=1}^{s_i-1} \tilde{Q}_l u_{i,s_i}^k \right) \right) \\
    \omega_{i,l}^k &= \omega_{i,l}^{k-1} - \tau \left( u_{i,l}^k - u_{i,s_i}^k \right) \quad i \in [m], l \in [s_i - 1] \\
    \tilde{x}^k &= 2x^k - x^{k-1} \\
    \tilde{\omega}^k &= 2\omega^k - \omega^{k-1}
\end{align*}
\]

We are now ready to state the convergence of CP for the general formulation via showing that it is an EB-algorithm.

**Proposition 6.** Let Assumptions 1-5 hold true. Then, applying the CP algorithm to $\tilde{L}(\chi, y)$ with parameters $\tau, \sigma > 0$ such that $\tau \sigma \|B\|^2 < 1$ results in
\[
\max_{\chi \in B_1, y \in B_2} \tilde{L}(\tilde{x}^N, y) - \tilde{L}(\chi, \tilde{y}^N) \leq \max_{\chi \in B_1, y \in B_2} \frac{1}{2N} \left( \tau^{-1} \|\chi - \chi^0\|^2 + \theta^{-1} \|y - y^0\|^2 \right).
\] (35)

for any compact sets $B_1$ and $B_2$. Furthermore, CP is an EB algorithm. More precisely, let $x^*$ be an optimal solution of (27) and let $(\lambda^*, \omega^*)$ be the optimal dual variables associated with its constraints. Then, $\lambda \in \mathbb{R}_+^m$ and $w$ such that $\lambda_i \equiv \lambda_{i,s_i}$ for all $i \in [m]$ satisfy
\[
L(\tilde{x}^N, (w, \lambda)) - L(x^*, (w^*, \lambda^*))
\]
where \( \sigma \) and \( \Xi \) defined in Assumptions 5 and 4, respectively.

26

g(x, \z_i)\] the respective variables. To decouple the constraint in vectors \( \varpi \) we will show that we can transform such problems to the biaffine form. For this, we introduce function, \( h(x, \z_i)\] and \( k(x, \z_i)\] as \( \mathbf{g} \).

Remark 1 (CP: the non-biaffine case). The CP algorithm can be extends to the case where \( g_i(x, \z_i)\] is not bilinear, but rather has the following general form:

\[
g_i(x, \z_i) := h_i(x)^\top \ell_i(z_i),
\]

where \( h_i(\cdot) : X \to \mathbb{R}^{k_i} \) and \( \ell_i(\cdot) : Z_i \to \mathbb{R}^{k_i} \), where each element of the mapping \( h_i(\cdot) \) is a convex function, \( i.e., h_{ij}(\cdot) \) is convex for all \( j \in [l_i] \), and each element of \( \ell_i(\cdot) \) is a concave function, \( i.e., k_{ij}(\cdot) \) is concave for all \( j \in [l_i] \). To maintain the convex-concave structure, we assume that

\[
\min_{j \in [l_i]} \ell_{ij}(z_i) \geq 0, \quad \min_{j \in [l_i]} h_{ij}(x) \geq 0 \quad \forall i \in [m].
\]

We will show that we can transform such problems to the biaffine form. For this, we introduce vectors \( \varpi_i \) (\( \zeta_i \) respectively) whose entries upper (lower) bound the entries of \( \ell_i(h_i) \). With these extra variables, the constraint \( g_i(x, \z_i) \leq 0, \forall z_i \in Z^i \) from problem (3) can be reformulated as

\[
\varpi_i^\top \zeta_i \leq 0, \quad \forall (z_i, \zeta_i) \in \Xi^i
\]

\[
h_i(x) \leq \varpi_i
\]

where \( \Xi^i = \{ \z_i = (z_i, \zeta_i) : z_i \in Z^i, \ell_i(z_i) \geq \zeta_i \} \). As we see, the first constraint becomes biaffine in the respective variables. To decouple the constraint in \( x \), one can also duplicate \( x \) to \( x_0, x_1, \ldots, x_m \), and add equality constraints \( x_i = x_0 \) for all \( i \in [m] \).

Accordingly, we can define an extended primal variable vector \( \chi = (x_0, \ldots, x_m, \varpi_1, \ldots, \varpi_m) \in \mathbb{R}^{n + \sum_{i=1}^{m} l_i} \) with feasible set

\[
\mathcal{X} = \{ \chi = (x_0, \ldots, x_m, \varpi_1, \ldots, \varpi_m) : x_0 \in X, h_i(x) \leq \varpi_i \}
\]

and an extended new uncertain parameter \( \xi_i = (z_i, \zeta_i) \) for constraint \( i \) such that \( \xi_i \in \Xi^i \). The \( i \)-th constraint of problem (3) becomes then:

\[
\tilde{g}_i(\chi, \xi) = \chi^\top Q_i \xi,
\]
where

\[
Q_i = \begin{bmatrix}
0_{(n(m+1)+\sum_{j=1}^{i-1}kJ_j)\times d_i} & 0_{(n(m+1)+\sum_{j=1}^{i-1}kJ_j)\times k_i} \\
0_{k_i\times d_i} & I_{k_i} \\
0_{\sum_{j=i+1}^{m}kJ_j\times d_i} & 0_{\sum_{j=i+1}^{m}l_j\times k_i}
\end{bmatrix}
\]

In the end, problem (3) can be shortly written as

\[
\min_{\chi \in \mathcal{X}} \tilde{c}^T \chi \\
\text{s.t.} \sup_{\xi_i \in \Xi_i} \tilde{g}_i(\chi, \xi_i) \leq 0, \quad i \in [m], \\
A\chi = 0
\]

where \(\tilde{c} = (c, 0)\) and

\[
A = \begin{bmatrix}
I_{(n)} & -I_{(n)} & 0_{(n\times n)} & \ldots & 0_{(n\times n)} & 0_{(n\times \sum_{i\in[m]} k_i)} \\
I_{(n)} & 0_{(n\times n)} & -I_{(n)} & \ldots & 0_{(n\times n)} & 0_{(n\times \sum_{i\in[m]} k_i)} \\
\vdots & \vdots & \ddots & \ddots & \vdots & \vdots \\
I_{(n)} & 0_{(n\times n)} & \ldots & 0_{(n\times n)} & -I_{(n)} & 0_{(n\times \sum_{i\in[m]} k_i)}
\end{bmatrix}
\]

Thus, we are back at the biaffine case for the lifted variables \((\chi, \xi)\), and CP can be applied as long as the projections over \(\Xi_i\) and \(\mathcal{X}\) are easily attainable.

5. Numerical experiment: robust quadratic programming

5.1. Introduction

In this section, we compare the numerical performance of our SGSP algorithm to the various approaches of Ho-Nguyen and Kılınç-Karzan (2018) and the standard cutting-plane algorithm. To do this, we consider an extension of the experiment in Ho-Nguyen and Kılınç-Karzan (2018), solving problems

\[
\min_{x \in \mathcal{X}} \sup_{z \in Z} g_0(x, z) \\
\text{s.t.} \ g_i(x, z) \leq 0, \quad \forall z \in Z, \ i \in [m]
\]

where the objective and constraint functions are:

\[
g_i(x, z) = \left\| P_{ik} z_k + P_{ik} x \right\|^2 + b_i^T x + c_i
\]

with \(P_{ik} \in \mathbb{R}^{L \times n}, \ b_i \in \mathbb{R} \) and \(c_i \in \mathbb{R}\). We assume that \(Z = \{z \in \mathbb{R}^K : \|z\|_2 \leq 1\} \) and \(X = \{x \in \mathbb{R}^n : \|x\|_2 \leq 1\}\). The experiment is an extension of the one performed in Ho-Nguyen and Kılınç-Karzan (2018), since the original considers problem with uncertainty only in the objective, while ours captures the more general setting with uncertain constraints.
Note that in (37), the functions \( g_i(x, \cdot) \) are convex, and therefore the problem does not readily fit our framework. However, in the pessimization problem

\[
\sup_{\|z\| \leq 1} g_i(x, z),
\]

known as the trust region problem, the function \( g_i(x, \cdot) \) can be replaced by equivalent concave function, where the equivalence is in the sense of the same maximal value. For this, \( g_i \) is first rewritten as

\[
g_i(x, z) = z^\top Q_i(x)z + 2r_i(x)^\top z + s_i(x)
\]

where \( Q_i(x) = P_i(x)^\top P_i(x) \) with \( P_i(x) \in \mathbb{R}^{n \times K} \) being a matrix whose columns are the vectors \( P_{ik}x \) for \( k \in [K] \), \( r_i(x) = P_i(x)^\top P_{00}x \) and \( s_i(x) = \|P_{00}x\|^2_2 - b_i^\top x - c_i \). For this function, by result of Jeyakumar and Li (2014) problem (38) can be reformulated as

\[
\sup_{\|z\| \leq 1} g_i(x, z) = \sup_{\|z\| \leq 1} z^\top (Q_i(x) - \lambda_{\max}(Q_i(x))I)z + 2r_i(x)^\top z + s_i(x) + \lambda_{\max}(Q_i(x))
\]

where \( \lambda_{\max}(\cdot) \) denotes the largest eigenvalue of the argument matrix. Since

\[
gbar_i(x, z) := z^\top (Q_i(x) - \lambda_{\max}(Q_i(x))I)z + 2r_i(x)^\top z + s_i(x) + \lambda_{\max}(Q_i(x))
\]

is convex-concave, using \( gbar_i(x, z) \) instead of \( g_i(x, z) \) in the robust formulation is equivalent and our setting applies. Therefore, in this set of experiments, whenever using first-order methods, we solve the modified problem:

\[
\begin{align*}
\min_{x \in X} & \quad \sup_{z \in Z} gbar_0(x, z) \\
\text{s.t.} & \quad gbar_i(x, z) \leq 0 \\
& \quad \forall z \in Z, \ i \in [m]
\end{align*}
\]

We note that with respect \( x \), problem (40) is semidefinite optimization problem. We are ready to state the four algorithms we compare:

- **Cutting planes algorithm** applied directly to (37), where pessimization subproblem (38) is solved using a generic solver to find \( z \) violating the constraints.
- **The SGSP algorithm** applied to (40). As part of this method, we first apply SGSP to find a Slater point for (40), and given the obtained Slater point we run SGSP to solve problem (40).
- **The OCO algorithm** of Ho-Nguyen and Kilinc-Karzan (2018), applied to (40), where both the variables \( x \) and \( z \) are solved using online gradient descent (OGD).
- **The FO-pessimization approach** of Ho-Nguyen and Kilinc-Karzan (2018) applied to (40), where the primal problem is solved using OGD and for each constraint the worst-case \( z \) is found by solving (39) using a generic solver.
We will compare the algorithms on their speed of reducing the feasibility and optimality gaps in the sense of Theorem 1. In the following Section, we describe the exact numerical setup and the results.

Remark 2. In the presented experiments we do not compare to the ‘nominal’ approach suggested in Ben-Tal et al. (2015b). Indeed, in the implementation of Ben-Tal et al. (2015b) presented in Ho-Nguyen and Kılınç-Karzan (2018), the authors suggests that in the \(k\)-th iteration, the following problem will be solved:

\[
\begin{align*}
\min_{x} & \quad g_0(x, z^k_i) \\
\text{s.t.} & \quad g_i(x, z^k_i) \leq 0 \quad i \in [m].
\end{align*}
\]

However, convergence of such a method requires for there to be a saddle point

\[
\inf_{x \in X} \sup_{z_i \in Z} g_i(x, z_i) = \sup_{z_i \in Z} \inf_{x \in X} g_i(x, z_i),
\]

which does not necessarily exist due to the convexity of \(g_i(x, \cdot)\). Thus, to solve the (37) problem using some kind of a ‘nominal’ approach with updated \(z^k_i\), one would need to either (i) use the semidefinite program (40) as the nominal oracle which contradicts the idea of solving ‘simple’ problems per iteration, or (ii) use the dual-subgradient meta-algorithm of Ben-Tal et al. (2015b) where \(z_i\) is lifted to a semidefinite matrix and the original nominal oracle (37) is used w.r.t. \(x\) – however, running the dual step would require projections on the spectahedron which, again, contradicts the idea of solving ‘simple’ problems, or (iii) use the dual-perturbation meta-algorithm of Ben-Tal et al. (2015b) – however, as we focus on deterministic algorithms, we do not include an implementation of Ben-Tal et al. (2015b).

5.2. Experiment setting

We explored different problem sizes, with respect to \(n\) the dimension of \(x\), \(m\) the number of constraints, \(K\) the dimension of uncertainty \(z_i\), and \(\ell\) the dimension of the vector in the norm. For each problem size, we sampled the problem data for 50 problem instances as follows. First, each entry of \(P_{ik}\) and \(b_i\) is sampled uniformly from interval \([-1, 1]\). Fixed value \(c_i = -0.05\) is chosen deterministically to ensure Slater feasibility of the problem. Next, \(P_{ik}\) and \(b_i\) are normalized as

\[
P_{ik} = \frac{P_{ik}}{S_{11}} \quad \text{where} \quad S_{11} = \left\| [P_{i0}^\top \ldots P_{iK}^\top]^\top \right\|_{2,2}
\]

\[
b_i = \frac{b_i}{S_{i2}} \quad \text{where} \quad S_{i2} = \|b_i\|_2
\]

To compare the algorithms in a fair way, we will use the same starting point for all of them. This point will be the optimal solution to the nominal problem

\[
\begin{align*}
\min_{x} & \quad g_0(x, 0) \\
\text{s.t.} & \quad g_i(x, 0) \leq 0 \quad i \in [m].
\end{align*}
\]
Table 4 Sizes of tested problems.

| Name   | n     | K     | L     | m     | maximum allowed time (seconds) | Output frequency (N iterations) |
|--------|-------|-------|-------|-------|--------------------------------|--------------------------------|
| Small  | 10    | 10    | 10    | 0     | 600                            | 100 †                           |
| Medium | 600   | 25    | 15    | 0     | 1200                           | 100                            |
| Large  | 3600  | 30    | 16    | 0     | 3600                           | 100                            |

† Cutting planes method records every iteration.

The values \( z_i \) are initialized as the zero vectors. We consider an \( \epsilon \) tolerance of 0.001 for both feasibility and optimality.

Note that the first-order algorithms require a choice of step-size. The step-size for OGD is chosen to be \( 2/(\|\nabla_x g_i(x, z_i)\| \sqrt{k}) \) and \( 2/(\|\nabla z_i g_i(x, z_i)\| \sqrt{k}) \) for the primal and dual steps at iteration \( k \), respectively. These stepsizes correspond with the analysis of OGD given in (Hazan 2016, Theorem 3.4). Similarly, the step sizes for SGSP is given by \( r^k = 2/(\|\nabla_x L(x, u)\| \sqrt{k}) \) and \( \theta^k = 2/(\|\nabla u_i L(x, u)\| \sqrt{k}) \) for the primal and dual steps at iteration \( k \), respectively. In both cases, the average solution \( \bar{x}^N \) is computed as a weighted sum of the iterates \( x^k \) with weights corresponding to the step-sizes. We note that these step-sizes were chosen instead of constant step-sizes, since they produced better results for all methods while retaining theoretical convergence guarantees.

Table 4 describes the different settings in which the algorithms were run. We consider small, medium and large problem sizes, with either no constraints or three constraints for each. For each problem size we specify the time limit we gave the methods as well as the sampling frequency for the output (see explanation below).

In order to measure the feasibility and the optimality gap for each method, for each parameter realization, we first define a constant \( \tilde{N} \), such that every \( \tilde{N} \) iterations statistics on the solution are gathered. Specifically, for all \( k \in \mathbb{N} \), let \( x^{k\tilde{N}} \) be the solution obtained after iteration \( k\tilde{N} \) of the algorithm, and let \( T_k \) be the time it took to run these \( k\tilde{N} \) iterations. The feasibility gap at iteration \( k\tilde{N} \) is given by

\[
FG_k := \max_{i \in [m]} \max_{z_i \in Z} g_i(x^{k\tilde{N}}, z_i).
\]

Defining the optimality gap to be infinity if the feasibility gap is larger than the defined \( \epsilon \), the optimality gap ratio at iteration \( k\tilde{N} \) is given by

\[
\text{OGR}_k := \frac{\delta_{\{p:p \leq \epsilon\}}(FG_k) \max_{z_i \in Z} g_0(x^{k\tilde{N}}, z_i) - LB}{LB}.
\]

\(^1\) We note that although the analysis of Nedić and Ozdaglar (2009) was done for a constant step-size, a similar analysis to the one shown in (Hazan 2016, Theorem 3.4) can be done for SGSP, with similar theoretical results.
In the above formula, \( \delta_{\{p \leq \epsilon\}} \) is the indicator function, and LB is the lower bound on the optimal solution obtained at the end of the cutting planes algorithm, by only considering the cuts added during the algorithm. Thus, for each time \( t \geq 0 \) we can record the minimal feasibility gap up to time \( t \) as

\[
\min_{k: T_k \leq t} \text{FG}_k,
\]

and the minimal optimality gap ratio up to time \( t \) as

\[
\min_{k: T_k \leq t} \text{OGR}_k.
\]

All the code, available in the online repository (Postek and Shtern 2024), was run using Python 3.7, with the CasADi package (Andersson et al. 2019) as the optimization interface. The optimization problems were solved using Gurobi 9.0.0 (Optimization 2020), with the exception of the trust-region subproblem (38) that due to numerical difficulties was solved using the IPOPT solver (Wächter and Biegler 2006). The code was run on a PowerEdge R740xd server with two Intel Xeon Cold 6254 3.1GHz processors, each with 18 cores, and a total RAM of 384GB.

### 5.3. Results

In Figure 1, we show the optimality convergence for all methods for the small, medium, and large instances without constraints. In the small instances, the computational and memory requirements of the cutting planes algorithm are negligible, and its performance dominates the other methods. Among the first order methods, however, the SGSP algorithm attains the fastest convergence.

When the problem instances become larger, the memory and computational requirements of the cutting planes algorithm become more significant, making its performance similar (medium instances) and then worse than the performance of the first-order methods. Among the first-order methods, our SGSP algorithm consistently dominates the other methods, although the differences with the OCO are rather small. Interestingly, we observe that for all methods it is the medium instances that keep their optimality gaps rather large for the longest. This might be related to the way we sample the problems where up to a certain point the problem size growth effect outweighs the ‘averaging out’ effect of the large matrices that make it easier to find a high-quality solution.

We now proceed to discuss the constrained problems. In Figures 2-4 we show the results for the small, medium, and large instances, respectively. We start with discussing the small examples. Similarly to the unconstrained case, the cutting planes algorithm is the fastest of all. Among the first-order methods, OFO is better than SGSP at finding feasible solutions fast, but having found them, it gets ‘stuck’ on improving optimality due to the need to run binary search on the objective value. Specifically, the problem is with the bi-section iterations which do not have a
feasible solution, but this infeasibility can only be identified after performing a very large number of iterations. The FO-Pess method is the slowest across all three measures.

For the medium instances in Figure 3, we would expect the first order methods to already perform better than the cutting planes algorithm. Indeed, the cutting planes algorithm becomes worse in decreasing the feasibility gap and the number of infeasible instances compared to all first-order methods. Among the feasible first-order methods, again, it is the SGSP algorithm that manages to reach optimality guarantees within the prescribed amount of time which are equivalent to those of
the cutting planes, while OFO and FO-Pess do get stuck due to the need for running the binary search on the objective function value.

Finally, for the largest instances in Figure 4, predictably, the cutting planes does not only have inferior performance to all first order based methods. We therefore focus on the comparison of computational performance of the three first-order methods. We observe that all first order methods are able to find feasible solutions faster than in the medium instances. We again believe that this is due to the problem generation process. Among the first-order methods, SGSP is relatively the slowest one to find a feasible solution for all the instances. With respect to the optimality gap, we observe that OFO slightly dominates the SGSP. The SGSPs performance on these instances is affected by the fact that at each iteration, it requires gradient computations with respect to all the constraints, whereas for the OFO this is done only for the constraint with largest current value of the left-hand side. Since for large dimensional problems these computations are substantial, the SGSP performs slightly slower than OFO, while dominating over the FO-Pess method.

6. Conclusions

In this paper, we have proposed a first-order optimization approach to robust optimization problems based on a convex-concave saddle-point reformulation of the problem’s Lagrangian. Our approach recovers the $O(1/\epsilon^2)$ convergence rate for general problems considered also by Ben-Tal et al. (2015b), Ho-Nguyen and Kılınç-Karzan (2018), and offers an improved $O(1/\epsilon)$ convergence guarantee for problems with a biaffine function structure. Similar to those algorithms, our method allows for a convenient parallelization of the computations related to different constraint functions and avoids problem size increase typical for the cutting planes and robust counterparts approaches. At the same time, our approach has the numerical benefit of avoiding a binary-search procedure.
for the optimal value of the objective as in Ho-Nguyen and Kilinc-Karzan (2018), while providing a deterministic algorithm which does not have to solve the nominal problem, contrary to Ben-Tal et al. (2015b).

Acknowledgements
We thank Fatma Kilinc-Karzan and Nam Ho-Nguyen for their discussions and sharing with us the implementation of their experiments. We also thank Shoham Sabach for his work at the early stage of this work and valuable suggestions.

Appendix
A. Proofs

Proof of Proposition 1. Proof of \(\Rightarrow\). We first note that from the definition of \(f_i\) and the construction of the lifted uncertainty set \(U^i\) we have that for any \(\lambda \in \mathbb{R}^m_+\) and \(x \in X\) and \(i \in [m]\).

\[
\lambda_i f_i(x) = \lambda_i \sup_{z_i \in \mathbb{Z}^i} g_i(x, z_i) = \sup_{\tilde{z}_i, (\hat{z}_i, \lambda_i) \in U^i} \lambda_i g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right).
\]

Indeed, if \((x^*, (u^*, w^*))\) is a saddle point of \(\bar{L}\) then, defining \(z_i^* = \tilde{z}_i^*/\lambda_i^*\) if \(\lambda_i^* > 0\) and \(z_i^* = 0\) otherwise, gives:

\[
L(x^*, (\lambda^*, w^*)) = \bar{L}(x^*, (u^*, w^*)), \tag{41}
\]

and

\[
\bar{L}(x^*, (u^*, w^*)) = \sup_{u \in U, w \in \mathbb{R}^r} \bar{L}(x^*, (u, w)) = \sup_{\lambda \in \mathbb{R}_+^m, \tilde{z}_i \in \mathbb{Z}^i, i \in [m], w \in \mathbb{R}^r} e^T x^* + \sum_{i=1}^m \lambda_i g_i \left( x^*, \frac{\tilde{z}_i}{\lambda_i} \right) + w^T (Ax^* - b)
\]
\[
\begin{align*}
&= \sup_{\lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r} \left( c^T x + \sum_{i=1}^m \lambda_i \sup_{z_i \in Z} g_i(x^*, z_i) + w^T (Ax^* - b) \right) \\
&= \sup_{\lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r} \left( c^T x + \sum_{i=1}^m \lambda_i f_i(x^*) + w^T (Ax^* - b) \right) \\
&= \sup_{\lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r} L(x^*, (\lambda, w)), \quad (42)
\end{align*}
\]

where the subsequent equalities follow from (i) the definition of \( \hat{L} \), (ii) definition of \( U^i \), (iii) the definition of \( f_i \), and (iv) the definition of \( L \). Moreover,

\[
\begin{align*}
\hat{L}(x^*, (u^*, w^*)) &= \inf_{x \in X} \hat{L}(x, (u^*, w^*)) \\
&= \inf_{x \in X} \left( c^T x + \lambda^*_i g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^T (Ax - b) \right) \\
&\leq \inf_{x \in X} \left( c^T x + \sum_{i=1}^m \sup_{\tilde{z}_i \in \{(\tilde{z}_i), \lambda_i^*) \in U^i} \lambda_i^* g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^T (Ax - b) \right) \\
&= \inf_{x \in X} \left( c^T x + \sum_{i=1}^m \sup_{\tilde{z}_i \in \mathbb{R}^*} \lambda_i^* g_i(x, \tilde{z}_i) + w^T (Ax - b) \right) \\
&= \inf_{x \in X} \left( c^T x + \sum_{i=1}^m \sup_{\tilde{z}_i \in \mathbb{R}^*} \lambda_i^* g_i(x, \tilde{z}_i) + w^* \right) \\
&= \inf_{x \in X} L(x, (\lambda^*, w^*)), \quad (43)
\end{align*}
\]

where the subsequent steps follow from (i) the definition of \( \hat{L} \), (ii) the definition of \( U^i \), and (iii) the definition of \( L \). Combining (41), (42) and (43) we obtain that

\[
\begin{align*}
\inf_{x \in X} \hat{L}(x, (u^*, w^*)) &= \inf_{x \in X} L(x, (\lambda^*, w^*)) \\
&= L(x^*, (\lambda^*, w^*)) \\
&= \hat{L}(x^*, (u^*, w^*)) \\
&= \sup_{u \in U^i, w \in \mathbb{R}^r} \hat{L}(x^*, (u, w)) \\
&= \sup_{\lambda \in \mathbb{R}^m_+, w \in \mathbb{R}^r} L(x^*, (\lambda, w)),
\end{align*}
\]

i.e., \((x^*, (\lambda^*, w^*))\) is a saddle point of \( L \).

**Proof of \( \Leftarrow \).** We shall show that \((x^*, (\lambda^*, w^*))\) can be extended to a saddle point of the lifted Lagrangian \( \hat{L} \). Note that defining \( \tilde{z}_i = \lambda_i z_i \) we have that

\[
\begin{align*}
&= c^T x^* + \sum_{i=1}^m \lambda_i^* f_i(x^*) + w^* (Ax^* - b) \\
&= \inf_{x \in X} c^T x + \sum_{i=1}^m \lambda_i^* \max_{(\tilde{z}_i), \lambda_i^*) \in U^i} g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^* (Ax - b) \\
&= \max_{(\tilde{z}_i), \lambda_i^*) \in U^i} \inf_{x \in X} c^T x + \sum_{i=1}^m \lambda_i^* g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^* (Ax - b) \\
&= \inf_{x \in X} c^T x + \sum_{i=1}^m \lambda_i^* g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^* (Ax - b).
\end{align*}
\]

The first equality is due to \((x^*, (\lambda^*, w^*))\) being a saddle point of \( L \) and the definition of \( f_i \). The third equality follows from Sion’s theorem, applicable due to boundedness of \( \{(\tilde{z}_i), \lambda_i^*) \in U^i : \lambda_i = \lambda_i^* \} \) and where we define \( \tilde{z}_i \) as a (necessarily-existing) maximizer:

\[
(\tilde{z}_i)_{i \in [m]} \in \arg \max_{(\tilde{z}_i), \lambda_i^*) \in U^i} \left\{ \inf_{x \in X} c^T x + \sum_{i=1}^m \lambda_i^* g_i(x, \frac{\tilde{z}_i}{\lambda_i^*}) + w^* (Ax - b) \right\}.
\]
Thus, defining \( u^*_i = (\tilde{z}^*_i, \lambda^*_i) \) we have that
\[
L(x^*, (\lambda^*, w^*)) = \bar{L}(x^*, (u^*, w^*)) = \inf_{x \in X} c^T x + \sum_{i=1}^m \lambda^*_i g_i \left( x, \frac{\tilde{z}^*_i}{\lambda^*_i} \right) + w^T (Ax - b) = \inf_{x \in X} \bar{L}(x, (u^*, w^*))
\]
Moreover,
\[
\bar{L}(x^*, (u^*, w^*)) = L(x^*, (\lambda^*, w^*)) = c^T x^* + \sum_{i=1}^m \lambda^*_i f_i(x^*) + w^T (Ax^* - b)
\]
\[
= c^T x^* + \sup_{\lambda \geq x^*, w \in \mathbb{R}^r} \sum_{i=1}^m \lambda_i \sup_{z_i \in \mathbb{R}^r} g_i(x^*, z) + w^T (Ax^* - b)
\]
\[
= c^T x^* + \max_{u_i \in U^*, i \in [m], w \in \mathbb{R}^r} \sum_{i=1}^m \lambda_i g_i \left( x^*, \frac{\tilde{z}^*_i}{\lambda^*_i} \right) + w^T (Ax^* - b)
\]
where we used the fact that \((x^*, (\lambda^*, w^*))\) is a saddle point of the original Lagrangian \( L \), and the definition of \( f_i \), \( L \), and \( \bar{L} \). Thus, we showed that \((x^*, (u^*, w^*))\) is a saddle point of \( \bar{L} \).

**Proof of Lemma 1.**

(i) Applying (Combettes and Müller 2018, Lemma 2.3) and using the fact that for any \( u_i = (\tilde{z}_i, \lambda_i) \in U^i \) we have that either both \( \lambda_i = 0 \) and \( \tilde{z}_i = 0 \), or \( z_i = \tilde{z}_i/\lambda_i \in Z^i \).

(ii) We use the fact that \( z_i = \tilde{z}_i/\lambda_i \) for all \( i \) such that \( \lambda_i > 0 \), and define
\[
d_z := \sum_{i=1}^m \lambda_i \tilde{d}_{x,i} = \sum_{i: \lambda_i > 0} \lambda_i \tilde{d}_{x,i} = \sum_{i: \lambda_i > 0} \lambda_i \partial_x g_i(x, z_i) = \sum_{i: \lambda_i > 0} \lambda_i \partial_x (x, \frac{\tilde{z}_i}{\lambda_i}) \cdot
\]
Thus, we see that (13) holds, i.e., \( v_z \in \partial_x \bar{L}(x, u) \). For all \( i \) such that \( \lambda_i > 0 \) define
\[
d_i := \tilde{d}_{x,i} \in \partial_{x_i} (-g_i(x, z_i)) = \partial_{x_i} \left( -g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) \right).
\]
For such \( d_i \) we have that
\[
v_i = (d_{x,i}, -g_i(x, z_i) - z^*_i d_{x,i}) = \left( d_i, -g_i \left( x, \frac{\tilde{z}_i}{\lambda_i} \right) \right) \in \partial_{x_i} \left( -\bar{L}(x, u) \right).
\]
i.e., the first case of (14) holds.

Finally, for all \( i \) such that \( \lambda_i = 0 \), using the fact that \( z_i = 0 \) define
\[
d_i := d_{z,i}, \in \partial_{x_i} (-g_i(x, 0)) \subseteq \cup_{\iota \in \mathbb{Z}} \partial_{x_i} (-g_i(x, z_i)) + T_{Z^i} (z_i)^*
\]
where the tangent cone \( T_{Z^i} (z_i)^* \) always includes the zero vector. Moreover, defining \( \phi_i := -g_i(x, 0) \) we have that
\[
(-g_i)^T(x, d_i) = \sup_{\iota \in \mathbb{Z}} \left\{ \iota^T d_i + g(x, \iota) \right\}
\]
\[
\leq g_i(x, 0) = -\phi_i
\]
where the first equality follows from the definition of convex conjugate, and the inequality follows from the convexity of \(-g_i(x, \cdot)\) and the fact that \( d_i, \in \partial_{x_i} (-g_i(x, 0)) \). Thus, meeting the definition (14) we have that for the case where \( \lambda_i = 0 \) we obtain that \( v_i = (d_i, \phi_i) \in \partial_{x_i} \left( -\bar{L}(x, u) \right) \).
Proof of Proposition 3. The projection over set $U^i$ is given by computing the minimizer of the following optimization problem

$$\min \left\{ \frac{1}{2} \| \mathbf{u} - \mathbf{v} \|^2 : \mathbf{v} \in U^i \right\} = \min \left\{ \frac{1}{2} \| \mathbf{z}_i - \zeta \|^2 + \frac{1}{2} (\lambda_i - \mu)^2 : \zeta \in \mu Z^i, \mu \geq 0 \right\}$$

(44)

It is clear that if $\mathbf{u}_i \in U^i$ its projection onto $U^i$ is the vector $\mathbf{u}_i$ itself. Otherwise, we can rewrite the projection problem as follows

$$\min \left\{ \| \mathbf{z}_i - \zeta \|^2 + (\lambda_i - \mu)^2 : \zeta \in \mu Z^i, \mu \geq 0 \right\} = \min \left\{ \| \mathbf{z}_i - \mu \mathbf{z}_i \|^2 + (\lambda_i - \mu)^2 : \mathbf{z}_i \in Z^i, \mu \geq 0 \right\}.$$

Computing this minimum first over $\mathbf{z}_i \in Z^i$ we obtain that if $\mu > 0$ then

$$\arg \min \left\{ \| \mathbf{z}_i - \mu \mathbf{z}_i \|^2 : \mathbf{z}_i \in Z^i \right\} = \arg \min \left\{ \| \mathbf{z}_i - \frac{\mathbf{z}_i}{\mu} \|^2 : \mathbf{z}_i \in Z^i \right\} = P_{Z^i} \left( \frac{\mathbf{z}_i}{\mu} \right)$$

in which case the optimal $\zeta$ is given in by $\mu P_{Z^i} \left( \frac{\mathbf{z}_i}{\mu} \right)$. Otherwise, if $\mu = 0$, then all points $\mathbf{z}_i \in Z^i$ are optimal and the optimal $\zeta$ is 0. Moreover,

$$\min \{ \| \mathbf{u}_i - \mathbf{v} \|^2 : \mathbf{v} \in U^i \} = \min \left\{ \inf_{\mu > 0} \psi_i(\mu), \| \mathbf{u}_i \|^2 \right\}$$

since

$$\psi_i(\mu) = \inf_{\zeta \in \mu Z^i} \frac{1}{2} \| \mathbf{z}_i - \zeta \|^2 + \frac{1}{2} (\lambda_i - \mu)^2$$

$$= \frac{\mu^2}{2} \| \frac{\mathbf{z}_i}{\mu} - P_{Z^i} \left( \frac{\mathbf{z}_i}{\mu} \right) \|^2 + \frac{1}{2} (\lambda_i - \mu)^2$$

$$= \frac{1}{2} \mu^2 \text{Dist} \left( \frac{\mathbf{z}_i}{\mu}, Z^i \right)^2 + \frac{1}{2} (\lambda_i - \mu)^2.$$
Proof of Proposition 4. We look at the saddle point problem (30) where the optimization problem is done over the nonrestricted sets \( U^{i,l} \), that is

\[
\min_{\omega} \max_{u_i \in U^{i,l}, \lambda \leq \lambda_i, s_i \leq \tilde{\lambda}} \tilde{g}_i(x^*, u_i) + \sum_{l=1}^{s_i-1} \omega_{i,l}^T(u_{i,l} - u_{i,s_i}).
\]  

(45)

We will run the proof in three steps:

- Proving that (30) has a saddle point.
- Proving that any saddle point of (45) is also a saddle point of (30).
- Proving the boundedness of \( \omega^* \) for the problem without the restriction.

It will therefore follow that after restricting \( \omega \) problem (30) still has saddle points. We begin with the first step. Indeed, since

\[
\sup_{u_i \in U^i, \lambda \leq \lambda_i \leq \tilde{\lambda}} \tilde{g}_i(x^*, u_i) \leq \sup_{u_i \in U^i} \tilde{g}_i(x^*, u_i) < \infty
\]

by the same logic as (28), (45) must have a saddle point.

Moving to the second step, we use the necessary and sufficient optimality conditions of the saddle point formulation to obtain that \((\omega^*, \tilde{u}^*)\) is a saddle point of (45) where \( u_{i,l} \in U^{i,l} \) if and only if

\[
\begin{align*}
\tilde{u}_{i,l}^T = u_{i,s_i}^T & \quad l \in [s_i - 1] \\
0 \in \omega_{i,l}^T - \partial \delta_{U^{l,i}}(\tilde{u}_{i,l}^T), & \quad l \in [s_i - 1] \\
0 \in -\partial \tilde{g}_i(-\tilde{g}_i(x^*, u_{i,s_i}^T)) - \sum_{l \in [s_i - 1]} \omega_{i,l}^T - \partial \delta_{U^{i,l}}(u_{i,s_i}^T) & \quad \text{for } l \in [s_i - 1],
\end{align*}
\]

where \( \tilde{U}^{i,s_i} = \{u_{i,s_i} \in U_{i,s_i} : \lambda \leq \lambda_{i,s_i} \leq \tilde{\lambda}\} \subseteq \tilde{U}^{i,s_i} \). Since \( 0 \leq \lambda \leq \tilde{\lambda} \leq \tilde{\lambda} \) it follows that \( u_{i,l}^T \in \tilde{U}^{i,l} \subseteq U^{i,l} \) and thus, \( \omega_{i,l}^T \in \partial \delta_{U^{l,i}}(u_{i,l}^T) \subseteq \partial \tilde{U}^{i,l}(u_{i,l}) \) and \( -\sum_{l \in [s_i - 1]} \omega_{i,l}^T \in \partial \delta_{U^{i,s_i}}(u_{i,s_i}^T) + \partial \tilde{g}_i(u_{i,s_i}^T) \). Therefore,

\[
\begin{align*}
\tilde{u}_{i,l}^T = u_{i,s_i}^T & \quad l \in [s_i - 1] \\
0 \in \omega_{i,l}^T - \partial \delta_{U^{l,i}}(u_{i,l}^T), & \quad l \in [s_i - 1] \\
0 \in -\partial \tilde{g}_i(-\tilde{g}_i(x^*, u_{i,s_i}^T)) - \sum_{l \in [s_i - 1]} \omega_{i,l}^T - \partial \delta_{U^{i,s_i}}(u_{i,s_i}^T) & \quad \text{for } l \in [s_i - 1],
\end{align*}
\]

which are exactly the optimality conditions when using \( \tilde{U}^{i,l} \) instead of \( U^{i,l} \), and so \((\omega^*, \tilde{u}^*)\) is also a saddle point of the restricted problem.

We now move to the last step, showing that \( \omega^* \) must be bounded. Note that if for some \( i \in [m] \) we have that \( \lambda_{i,s_i}^l = 0 \), then the pair \( (\omega^*, \tilde{u}^*) = (0, 0) \) is a saddle point for the \( i \)th element of the sum, and so restricting the norm of \( \omega_i \) is possible. We therefore continue with bounding \( \omega_i \) for the case where \( \tilde{\lambda} > 0 \). By definition, we have that \( u_{i,s_i}^l = (z_{i,s_i}^l, \alpha_i^l, \lambda_i^l) \) where \( 0 \leq \lambda \leq \lambda_{i,s_i}^l \leq \tilde{\lambda} \). Also note that it must be that \( u_{i,s_i}^l = u_{i,l} \), otherwise the minimization over \( \omega_i \) would yield minus infinity. Thus, we have that \( z_{i,s_i}^l \in \cap_{l=1}^{s_i - 1} Z^{i,l} \). Finally, we have that

\[
0 \geq \lambda_{i,s_i}^l, \tilde{g}_i(x^*, z_{i,s_i}^l) = \tilde{g}_i(x^*, u_{i,s_i}^l) + \sum_{l=1}^{s_i-1} (\omega_{i,l}^T - u_{i,s_i}^l)
\]
where the first inequality comes from the feasibility of \( x^* \) for all \( z_i \in Z^i = \cap_{i=1}^s Z^{i,l} \) and the non-negativity of \( \lambda^i_{l,s} \), the first equality follows from \( u^i_{l,s} = u^i_{l,l} \), the second equality follows from the optimality of \( u^l \) and \( \omega^l \) for the saddle point problem, the second inequality follows from choosing \( u_{i,l} = (z_i, \lambda^i_{l,s}, \lambda^i_{l,s}) \), and the third inequality follows from choosing \( z_{i,l} = 0 \) for all \( l \in [s_i]/\{l'\} \) and \( z_{i,l'} = \epsilon, \nu^i_{l,l'} \). Therefore, if \( \lambda^i_{l,s} > 0 \) it follows from (47) that \( \| \nu^i_{l,l'} \| \leq -g_i(x^*, 0)/\epsilon_i \leq \bar{\mu}/\epsilon_i \).

Similarly, taking an arbitrary \( l' \in [s_i] \setminus \{l'\} \), we can bound (46) from below by choosing \( u_{i,l} = (0, \lambda^i_{l,s}) \) for all \( l \in [s_i] \setminus \{l'\} \) and \( u_{i,l'} = (0, \lambda^i_{l,s}) \).

\[
\begin{align*}
0 \geq & \max_{u_{i,l} \in U^{i,l}, \lambda \leq \lambda_{i,s} \leq \bar{\lambda}} g_i(x^*, 0) + \sum_{l=1}^{s_i-1} \mu^i_{l,l}(\lambda_{i,l} - \lambda^i_{l,s}) \\
&= \lambda^i_{l,s}(g_i(x^*, 0) - \mu^i_{l,l} + \mu^i_{l,l'} \lambda^i_{l,s}).
\end{align*}
\]

Since \( \lambda^i_{l,l'} \) can be taken to infinity, it follows that the equality holds only if \( \mu^i_{l,l'} \leq 0 \). Moreover, choosing \( \lambda^i_{l,l'} = 0 \) implies that \( -\mu^i_{l,l'} \leq -g_i(x^*, 0) \leq \bar{\mu} \). Since \( l' \) was arbitrarily chosen the proof is complete.

**Proof of Proposition 5.** In the proof the the proposition, claim (31) plays the key role, from which (32) and (33) follow.

**Proof of (31)** Denoting \( G_\chi, G_{u,i}, G_w \) as bounds on the subgradients \( v^k_\chi \in \partial_\chi \bar{L}(x^{k-1}, y^{k-1}), v^k_u \in \partial_u (-\bar{L}(x^{k-1}, y^{k-1})), v^k_w \in \partial_w \bar{L}(x^{k-1}, y^{k-1}) \) used throughout the algorithm, the first of the corollary follows directly from (Nedić and Ozdaglar 2009, Lemmas 3.1 and 3.2). It is left to prove that under the chosen assumptions these bounds exist and are equal to the stated values.

We begin with the primal variables \( \chi \). Denoting \( u^{k-1}_{l,s} = (z^{k-1}_{l,s}, \lambda^{k-1}_{l,s}) \) and defining \( z^{k-1}_{l,s} \) as

\[
\begin{align*}
\begin{cases}
z^{k}_{l,s} = \frac{z^{k}_{l,s}}{\lambda_{l,s}}, & \lambda_{l,s} > 0 \\
0, & \lambda_{l,s} = 0.
\end{cases}
\end{align*}
\]

we have that

\[
v^k_\chi = \begin{bmatrix}
c + A^T w^{k-1} + \sum_{i=1}^m \lambda^{k-1}_{i,s} d^{k}_{i,s} \\
(u^{k-1}_{l,s} - u^{k-1}_{l,s-1}) \\
\vdots \\
(u^{k-1}_{m,s} - u^{k-1}_{m,s-1})
\end{bmatrix} \in \partial_\chi \bar{L}(x^{k-1}, u^{k-1}),
\]

where \( d^{k}_{i,s} \in \partial g_i(x^{k-1}, z^{k-1}_{i,s}) \). By boundedness of the subgradients, \( \lambda_i \) and \( w \) we can bound the first component

\[
\left\| c + A^T w^{k-1} + \sum_{i=1}^m \lambda^{k-1}_{i,s} d^{k}_{i,s} \right\| \leq \|c\| + \|A\| R_w + \sum_{i=1}^m \bar{\lambda} G_{x,i}.
\]

**Proof of (32)** Denoting \( G_\lambda, G_{u,l}, G_w \) as bounds on the subgradients in the dual problem

\[
v^k_\lambda \in \partial_\lambda \bar{L}(y^{k-1}, x^{k-1}).
\]
By the definition of $\tilde{U}^{i,l}$ we have that for all $i \in [m]$ and $l \in [s_i - 1]$
\[
\|u^{k-1}_{i,s_i} - u^{k-1}_{i,l}\| \leq (\lambda^{k-1}_{i,s_i} - \lambda^{k-1}_{i,l}) + \|\tilde{z}_{i,s_i} - \tilde{z}_{i,l}\|
\leq 2\lambda + \tilde{\lambda} \max_{z_{i,l} \in Z^{i,l}, z_{i,l} \in Z^{i,l}} \|z_{i,s_i} - z_{i,l}\| \leq \tilde{\lambda}(2 + R_{i,s_i} + R_{i,l}).
\]
Adding these two bounds, we obtain the following bound:
\[
\|v^k_x\| \leq \|c\| + \|A\|R_w + \sum_{i \in [m]} \tilde{\lambda}G_{i,s_i} + \tilde{\lambda} \sum_{i \in [m]} (s_i - 1)(2 + R_{i,s_i}) + \sum_{l \in [s_i - 1]} R_{i,l}) = G_x
\]
Now, we will bound the norms of the subgradients corresponding to the dual variables $u_{i,l}$, $w$:
\[
v^k_{i,s_i} = \left[ \begin{array}{c} v^k_{i,1} \\ \vdots \\ v^k_{i,s_i} \end{array} \right]
\]
First, consider the subgradients $v^k_{i,s_i}$ with respect to $u_{i,s_i}$. According to Lemma 1
\[
v^k_{i,s_i} = \left[ (\tilde{g}_i) \tilde{d}^k_{i,s_i} \right] - \sum_{l \in [s_i - 1]} \omega^k_{i,l} \in \partial_{u_{i,s_i}}(-\tilde{L})(\chi^{k-1}, u^{k-1}),
\]
where $\tilde{d}^k_{i,s_i} = \partial_{u_{i,l}}(-g_i)(\chi^{k-1}, z^{k-1}_{i,l})$. From the definitions of the sets $\tilde{U}^{i,l}$, $\Omega^{i,l}$, and $W$ we get:
\[
\|v^k_{i,s_i}\| \leq \|\tilde{d}^k_{i,s_i}\| + \|(-g_i)(\chi^{k-1}, z^{k-1}_{i,s_i})\| + \|u^{k-1}\| \|z^{k-1}_{i,s_i}\| + \sum_{l \in [s_i - 1]} \|\omega^k_{i,l}\|
\leq (\tilde{\lambda} + R_{i,s_i})G_{i,s_i} + \tilde{g}_i + \tilde{\mu}_i(s_i - 1) \left( 1 + \frac{1}{\epsilon_i} \right).
\]
Moreover, for any $l \in [s_i - 1]$ it follows from the definition $\Omega^{i,l}$ that
\[
v^k_{i,l} = -\omega^k_{i,l} \in \partial_{u_{i,l}}(-\tilde{L})(\chi^{k-1}, y^{k-1}) \Rightarrow \|v^k_{i,l}\| \leq \tilde{\mu}_i \left( 1 + \frac{1}{\epsilon_i} \right).
\]
In the end, for the subgradients w.r.t. $w$, we have $v^k_w = -Ax^{k-1} + b$ so by boundedness of $X$ we obtain the bound $\|v^k_w\| \leq \|A\|R_x + \|b\|$. Using the defined $v^k_x$, $v^k_{i,l}$ and $v^k_w$ in the algorithm we obtain the desired result.

Proof of (32) To prove this claim, we will use (31). First, define $u^*_i = \lambda_i z^*_i$ where $z^*_i = \arg \max_{x_i \in Z} g(\tilde{x}^N, z_i)$. Following (28) we have that
\[
L(\tilde{x}^N; (\lambda, w)) = c^T \tilde{x}^N + \sum_{i \in [m]} \tilde{g}_i(\tilde{x}^N, u^*_i) + w^T (Ax^N - b)
\]
\[
= \min_u \max_{u_{i,l} \in U^{i,l}, \lambda_{i,l} = \lambda_i} c^T x^N + \sum_{i \in [m]} \tilde{g}_i(\tilde{x}^N, u_{i,s_i}) + w^T (Ax^N - b) + \sum_{i = 1}^m \sum_{l = 1}^{s_i - 1} \omega_{i,l}^T (u_{i,l} - u_{i,s_i})
\]
\[
\leq \max_{u_{i,l} \in U^{i,l}, \lambda_{i,l} = \lambda_i} c^T \tilde{x}^N + \sum_{i \in [m]} \tilde{g}_i(\tilde{x}^N, u_{i,s_i}) + w^T (Ax^N - b) + \sum_{i = 1}^m \sum_{l = 1}^{s_i - 1} (\omega_{i,l}^\top) (u_{i,l} - u_{i,s_i})
\]
\[
= \max_{u_{i,l} \in U^{i,l}, \lambda_{i,l} = \lambda_i} \tilde{L}((\tilde{x}^N, \omega^N), (\bar{u}, \bar{w})).
\]
We move on to lower bounding $L(x^*, \tilde{\Lambda}^N, \tilde{w}^N)$ with a term of the form $\tilde{L}((x^*, \omega), \tilde{y}^N)$:

$$
L(x^*, (\tilde{\Lambda}^N, \tilde{w}^N)) = \max_{\omega_i \in D_i, \lambda_i = \lambda^N_{i, s_i}} c^T x^* + \sum_{i \in [m]} b_i(x^*, u_i)
$$

$$
= \min_{\omega} \max_{u_{i,i} \in D_i, k \in [s_i], \lambda_i = \lambda^N_{i, s_i}} c^T x^* + \sum_{i \in [m]} b_i(x^*, u_i) + \sum_{i=1}^{m} \sum_{l=1}^{s_i - 1} \omega^T_{i,l}(u_{i,l} - u_{i,s_i})
$$

$$
= \min_{\omega} \max_{u_{i,i} \in D_i, k \in [s_i], \lambda_i = \lambda^N_{i, s_i}} c^T x^* + \sum_{i \in [m]} b_i(x^*, u_i) + \sum_{i=1}^{m} \sum_{l=1}^{s_i - 1} \omega^T_{i,l}(u_{i,l} - u_{i,s_i})
$$

$$
\geq \min_{\omega \in \Omega} \max_{u_{i,i} \in D_i, k \in [s_i], \lambda_i = \lambda^N_{i, s_i}} c^T x^* + \sum_{i \in [m]} b_i(x^*, u_i) + \sum_{i=1}^{m} \sum_{l=1}^{s_i - 1} \omega^T_{i,l}(\tilde{u}_{i,l} - \tilde{u}_{i,s_i})
$$

$$
= \min_{\omega \in \Omega} \tilde{L}((x^*, \omega), \tilde{y}^N)
$$

where the first equality follows (28), the second equality follows from Proposition 4, the third equality follows from the fact that existence of a saddle point, established in (28), and the final inequality follows from the definition of $\tilde{L}$. Moreover, for any $\tilde{\Lambda}^N_i \equiv \lambda^N_{i, s_i}$ and $\tilde{w}^N$, it follows from $(x^*, (\lambda^*, w^*))$ being a saddle point of $L$ that

$$
L(x^*, (\tilde{\Lambda}^N, \tilde{w}^N)) \leq L(x^*, (\lambda^*, w^*))
$$

Combining (48), (50), and (51) we have that for any $\lambda_i \in [0, \tilde{\lambda}]$ and any $w \in W$ the following holds

$$
L(x^*, (\lambda, w)) - L(x^*, (\lambda^*, w^*)) \leq L(x^*, (\tilde{\lambda}, w^*)) - L(x^*, (\tilde{\lambda}_N, \tilde{w}^N)) \leq \max_{\omega \in \Omega} \max_{u_{i,i} \in D_i, k \in [s_i], \lambda_i = \lambda} \left( \tilde{L}((x^*, \omega^N), y) - \tilde{L}((x^*, \omega), \tilde{y}^N) \right)
$$

$$
\leq \max_{\omega \in \Omega} \max_{u_{i,i} \in D_i, k \in [s_i], \lambda_i = \lambda} \frac{1}{2\sqrt{N}} \left( \|x^0 - x^*\|^2 + \|\omega^0 - \omega\|^2 + \tilde{\gamma} G^2 \right)
$$

$$
+ \sum_{i=1}^{m} \left( \sum_{l=1}^{s_i} \|u_{i,l}^0 - u_{i,l}\|^2 + \tilde{\gamma} G^2 \right) + \|w^0 - w\|^2 + \tilde{\gamma} G^2 ,
$$

where the last inequality follows from (31). Now we can use the fact that for any $\omega, \omega^0 \in \Omega$

$$
\|\omega - \omega^0\| \leq 2 \sum_{i=1}^{m} \mu_i (s_i - 1) (1 + 1/\epsilon_i)
$$

and for any $u_{i,l}^0, u_{i,l} \in \tilde{U}_i$ such that $u_{i,l} = (\tilde{z}_{i,l}, \lambda_i), \lambda_i = \lambda_i$

$$
\|u_{i,l}^0 - u_{i,l}\|^2 \leq \max\{\lambda^0_i, \lambda_i\}^2 (1 + 4R^2_i).$$
In the end we have
\[ \|w^0 - w\|^2 \leq 2\max\{\|w\|, \|w_0\|\}^2 \]
We can use these inequalities and the bounds on variables to bound (52) and the definition of \( \phi \) and \( \sigma_i \) we obtain the desired result.

**Proof of (33)** For the last claim we use the fact that if \(((x^*, \omega^*), (\tilde{u}, w^*))\) is a saddle point of \( \tilde{L} \) then \((x^*, (\lambda^*, w^*))\) where \( \lambda_i^* = \lambda_{i,s_i} \) is a saddle point of \( L \). Thus,
\[ L(\tilde{x}^N, (\lambda^*, w^*)) \geq L(x^*, (\lambda^*, w^*)) = c^\top x^* \]
where the inequality follows from \((x^*, (\lambda^*, w^*))\) being a saddle point, and the equality follows from the saddle point value for the Lagrangian being equal to the optimal primal objective value.

**Proof of Proposition 6.** The first part of the proposition is a direct result of (Chambolle and Pock 2011, Theorem 1). To prove the second part, the strategy is the same as in the SGSP case: we upper (lower) bound the first (second) terms in the left-hand side of (36) with terms that are like the ones in the LHS of (35) albeit over a bounded domain, and then suprime the RHS of (35) over that domain. Defining \( u_{i,l}^* = \lambda_i(z_i^*), 1 \) for all \( l = 1, \ldots, s_i \), where \( z_i^* := \arg\max_{z_i \in \mathbb{Z}} g_i(\tilde{x}^N, z_i) \), we then have:
\[
L(\tilde{x}^N, (\lambda, w)) = c^\top \tilde{x}^N + w^\top (A\tilde{x}^N - b) + \sum_{l \in [m]} \tilde{g}_i(\tilde{x}^N, u_i^*)
= c^\top \tilde{x}^N + w^\top (A\tilde{x}^N - b) + \sum_{l \in [m]} \tilde{g}_i(\tilde{x}^N, u_i^*) + \sum_{i \in [s_i - 1]} (\omega_i^*)^\top (u_{i,l}^* - u_{i,s_i}^*)
= \tilde{L}(\tilde{x}^N, (u^*, w)).
\]
where the first equality follows from the definition of \( L \), the second equality follows from \( u_{i,l}^* = u_{i,s_i}^* \) for all \( l \in [s_i - 1] \) and \( i \in [m] \), and the third equality follows from definition of \( \tilde{L} \). Since \((x^*, (\lambda^*, w^*))\) is a saddle point of \( L \), it can be extended to \((x^*, y^*)\) which is a saddle point of \( \tilde{L} \), and by definition
\[
L(x^*, (\lambda^*, w^*)) = \tilde{L}(x^*, y^*) \geq \tilde{L}(x^*, \tilde{y}^N),
\]
which combined with inequalities (35) (with \( B_1 = \{x^*\} \) and \( B_2 = \{y = (u^*, w) : u_{i,l}^* = u_{i,s_i}^* \} \)) and (53) gives
\[
L(\tilde{x}^N, (\lambda, w)) - L(x^*, (\lambda^*, w^*)) \leq \max_{y \in B_2} \tilde{L}(\tilde{x}^N, y) - \tilde{L}(x^*, \tilde{y}^N)
\leq \max_{y \in B_2} \frac{\tau - 1}{2N} ||x^* - \chi^0||^2 + \sigma - 1 ||y - y^0||^2.
\]
From Proposition 4 and \( \omega^0 = 0 \) we have that
\[
||x^* - x^0||^2 = ||x^* - x^0||^2 + ||\omega^* - \omega^0||^2 \leq ||x^* - x^0||^2 + \sum_{i=1}^m (s_i - 1) \mu_i^2 \left( 1 + \frac{1}{\epsilon_i} \right)^2.
\]
The boundedness of \( Z^i \) implies that
\[
\sup_{y = (\tilde{u}, w) : u_{i,l}^* = u_{i,s_i}^*, z_i^* = \lambda_i^*} ||y - y^0||^2 \leq \sum_{i \in [m]} \max_{l \in [s_i]} \sum_{i \in [s_i]} ||u_{i,l}^* - u_{i,s_i}^*||^2 + ||w - w^0||^2
\leq \sum_{i \in [m]} \sum_{l \in [s_i]} (1 + 4R_i^2) \max_{l \in [s_i]} \lambda_{i,l}^0 + \max{||w||, ||w^0||}^2
= \sum_{i \in [m]} \sigma_i \max{||\lambda_i^0||^2, \sigma_i^2} + \max{||w||, ||w^0||}^2.
\]
Combining (54), (55), and (56) we obtain (36). The last part follows by the same argument as in the case of SGSP.

References

Alacaoglu A, Malitsky Y, Cevher V (2021) Forward-reflected-backward method with variance reduction. *Computational optimization and applications* 80(2):321–346.

Andersson JAE, Gillis J, Horn G, Rawlings JB, Diehl M (2019) CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation* 11(1):1–36, URL http://dx.doi.org/10.1007/s12532-018-0139-4.

Auslender A, Teboulle M (2009) Projected subgradient methods with non-euclidean distances for non-differentiable convex minimization and variational inequalities. *Mathematical Programming* 120(1):27–48.

Bauschke HH, Combettes PL (2011) *Convex analysis and monotone operator theory in Hilbert spaces* (Springer).

Ben-Tal A, den Hertog D, Vial JP (2015a) Deriving robust counterparts of nonlinear uncertain inequalities. *Mathematical Programming* 149(1):265–299, ISSN 1436-4646, URL http://dx.doi.org/10.1007/s10107-014-0750-8.

Ben-Tal A, El Ghaoui L, Nemirovski A (2009) *Robust optimization* (Princeton University Press).

Ben-Tal A, Hazan E, Koren T, Mannor S (2015b) Oracle-based robust optimization via online learning. *Operations Research* 63(3):628–638.

Bertsimas D, Brown DB, Caramanis C (2011) Theory and applications of robust optimization. *SIAM Review* 53(3):464–501.

Bienstock D (2007) Histogram models for robust portfolio optimization. *Journal of Computational Finance* 11(1):1.

Chambolle A, Pock T (2011) A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision* 40(1):120–145.

Combettes PL, Müller CL (2018) Perspective functions: Proximal calculus and applications in high-dimensional statistics. *Journal of Mathematical Analysis and Applications* 457(2):1283–1306.

Gabriel V, Murat C, Thiele A (2014) Recent advances in robust optimization: An overview. *European Journal of Operational Research* 235(3):471–483.

Gidel G, Jebara T, Lacoste-Julien S (2017) Frank-wolfe algorithms for saddle point problems. *Artificial Intelligence and Statistics*, 362–371 (PMLR).

Hazan E (2016) Introduction to online convex optimization. *Foundations and Trends in Optimization* 2(3-4):157–325, ISSN 2167-8888, URL http://dx.doi.org/10.1561/2400000013.

Ho-Nguyen N, Kılıç-Karzan F (2018) Online first-order framework for robust convex optimization. *Operations Research* 66(6):1670–1692.

Ho-Nguyen N, Kılıç-Karzan F (2019) Exploiting problem structure in optimization under uncertainty via online convex optimization. *Mathematical Programming* 177(1):113–147.

Jeyakumar V, Li G (2014) Trust-region problems with linear inequality constraints: exact sdp relaxation, global optimality and robust optimization. *Mathematical Programming* 147(1-2):171–206.

Juditsky A, Nemirovski A (2011) First order methods for nonsmooth convex large-scale optimization, ii: utilizing problems structure. S SN Sra, Wright SJ, eds., *Optimization for Machine Learning*, chapter 6, 149–181 (MIT press Cambridge, MA).

Korpelevich G (1976) Extragradient method for finding saddle points and other problems. *Ekonomiya i Matematicheskie Metody* 12(4):747–756.

Malitsky Y, Tam MK (2020) A forward-backward splitting method for monotone inclusions without cocoercivity. *SIAM Journal on Optimization* 30(2):1451–1472.

Nemirovski A, Malitsky Y, Cevher V (2021) Exploiting problem structure in optimization under uncertainty via online convex optimization. *SIAM Journal on Optimization* 30(2):1451–1472.

Malitsky Y, Nemirovski A (2004) Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization* 15(1):229–251.

Nesterov Y (2007) Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming Computation* 10(2):277–329.

Optimization G (2020) Gurobi optimizer reference manual. URL http://www.gurobi.com.

Ouyang Y, Xu Y (2021) Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming* 185(1):1–35.

Postek K, Shtern S (2024) First-order algorithms for robust optimization problems via convex-concave saddle-point lagrangian reformulation. URL http://dx.doi.org/10.1287/ijoc.2022.0200, available for download at http://github.com/INFORMSJoC/2022.0200.

Tseng P (1991) Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM Journal on Control and Optimization* 29(1):119–138.

Wächter A, Biegler LT (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming* 106(1):25–57.

Zhang J, Hong M, Zhang S (2022) On lower iteration complexity bounds for the convex concave saddle point problems. *Mathematical Programming* 194(1-2):901–935.