



HAL
open science

Solving a class of bilevel programs with quadratic lower level

Martina Cerulli, Antoine Oustry, Claudia d'Ambrosio, Leo Liberti

► **To cite this version:**

Martina Cerulli, Antoine Oustry, Claudia d'Ambrosio, Leo Liberti. Solving a class of bilevel programs with quadratic lower level. 2021. hal-03339887v1

HAL Id: hal-03339887

<https://hal.science/hal-03339887v1>

Preprint submitted on 9 Sep 2021 (v1), last revised 22 Feb 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 **SOLVING A CLASS OF BILEVEL PROGRAMS WITH QUADRATIC LOWER**
2 **LEVEL** *

3 MARTINA CERULLI[†], ANTOINE OUSTRY^{†‡}, CLAUDIA D'AMBROSIO[†], AND LEO LIBERTI[†]

4 **Abstract.** We focus on a particular class of bilevel programs with a quadratic lower-level problem, which can be
5 obtained by reformulating semi-infinite problems with an infinite number of quadratically parametrized constraints.
6 We propose a new approach to solve this class of bilevel programs, based on the dual of the lower-level problem, which
7 can lead to a convex or a semidefinite programming problem, depending on the parametrization of the lower level
8 with respect to the upper-level variables. This approach is compared with a new tailored cutting plane algorithm,
9 which is proved to be convergent. The rate of convergence of this cutting plane algorithm, directly related to the
10 iteration index, is derived when the upper-level objective function is strongly convex, and under a strict feasibility
11 assumption. We successfully test the two proposed methods on two applications: the constrained quadratic regression
12 and a zero-sum game with cubic payoff.

13 **Key words.** Bilevel programming, Semi-infinite programming, Semidefinite programming, Cutting Plane

14 **AMS subject classifications.** 90C34, 90C22, 90C46

15 **1. Introduction.** A bilevel programming (BP) problem is an optimization problem where a
16 subset of the variables is constrained to take the value of an optimal solution of another given
17 optimization problem parameterized by the remaining variables. The former optimization problem
18 is defined as the *upper-level problem*, and the latter as the *lower-level problem*. Many real situations
19 can be modeled as BP programs, in particular when they involve a hierarchical relationship between
20 two decision levels.

21 Since BP problems are extremely challenging (both theoretically [32, §6] and practically), it is
22 not surprising that much of the research in this field has focused on the simplest cases with linear,
23 convex quadratic, or general convex objective and feasible region. In this paper, we propose a new
24 analysis, and two approaches to solve a special class of bilevel problems, with a possibly non-convex
25 quadratic programming (QP) lower-level problem and convex upper-level constraints and objective.

26 We assume that the upper-level problem has a continuous convex objective function $F(x)$
27 (where x is an array of upper-level decision variables), and a convex feasible set $\mathcal{X} \subset \mathbb{R}^m$ depending
28 only on x . The lower-level problem is a QP in the lower-level decision variables y , with a possibly
29 non-convex objective function, but with a feasible set consisting of the polytope

30
$$\mathcal{F} = \{y \in \mathbb{R}^n : Ay \leq b\} = \{y \in \mathbb{R}^n : \forall j \leq r (a_j^\top y \leq b_j)\},$$

31 where a_j is the j -th row of the matrix A , and r is an integer.

32 We make two overarching assumptions on the BP class of interest: (i) \mathcal{F} does not depend on x ;
33 (ii) the upper-level problem depends only on the optimal value of the lower-level problem, rather
34 than its optimal solutions.

*This research was partly funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement n. 764759 ETN "MINOA".

[†]LIX - CNRS, École Polytechnique, Institut Polytechnique de Paris, 91120, Palaiseau, France (mcerulli@lix.polytechnique.fr, oustry@lix.polytechnique.fr, dambrosio@lix.polytechnique.fr, liberti@lix.polytechnique.fr).

[‡]École des Ponts, 77455, Marne-la-Vallée, France.

35 Thus, the Mathematical Programming (MP) formulation we study is as follows:

$$\begin{aligned}
 & \left\{ \begin{array}{l} \min_{x \in \mathbb{R}^m} F(x) \\ \text{s.t. } x \in \mathcal{X} \\ h(x) \leq \min_{y \in \mathbb{R}^n} \{ \frac{1}{2} y^\top Q(x) y + q(x)^\top y \mid Ay \leq b \}, \end{array} \right. \\
 36 \quad (\text{BP}) & \\
 37 &
 \end{aligned}$$

38 where F , and h , are continuous convex functions in the upper-level variables x , both the $n \times n$
 39 matrix $Q(x)$ and the n -dimensional vector $q(x)$ depend linearly on x , A a $r \times n$ matrix, and b a
 40 r -dimensional vector.

41 Here are the technical assumptions we make on (BP).

42 ASSUMPTION 1. \mathcal{X} is convex.

43 ASSUMPTION 2. The functions $x \mapsto q(x)$ and $x \mapsto Q(x)$ are linear.

44 ASSUMPTION 3. The function $x \mapsto h(x)$ is convex and Lipschitz continuous.

45 ASSUMPTION 4. The set \mathcal{F} is compact, and a scalar $\rho > 0$ is known such that (s.t.) the set \mathcal{F}
 46 is included in the centered l_2 -ball with radius ρ .

47 In the following, given a formulation (P) of an optimization problem, we will use the term
 48 *reformulation* to describe a formulation having the same set of optima of (P), i.e., what is defined
 49 as *exact reformulation* in [18, Definition 10]. With the term *relaxation*, we will refer to a formulation
 50 having a feasible set which contains the feasible set of (P) [18, Definition 13]. Finally, we will use
 51 the term *restriction* when referring to a formulation having a feasible set which is included in the
 52 feasible set of (P).

As mentioned above, (BP) does not consider the optimal solutions of the lower-level problem, but only its optimal objective function value. This renders “pessimistic” or “optimistic” interpretations of (BP) meaningless. The BP class (BP) arises in many applications requiring semi-infinite programming (SIP) problems, i.e. optimization problems with a finite number of variables, and an infinite number of parametrized constraints of the type $\forall y \in Y, g(x, y) \geq 0$. Indeed, this is equivalent to:

$$0 \leq \min_{y \in Y} g(x, y),$$

53 which allows the reformulation of the SIP constraints into a lower-level problem of a BP in the class
 54 (BP), as long as $g(x, y) = \frac{1}{2} y^\top Q(x) y + q(x)^\top y - h(x)$ and $Y = \mathcal{F}$. We remark that, in a bilevel
 55 context, the function $\phi(x) = \min_{y \in Y} g(x, y)$ is called *optimal value function*.

56 Our first contribution is an analysis of (BP) which yields a single-level formulation with a finite
 57 number of constraints. This single-level formulation is obtained by dualizing, using Semidefinite
 58 Programming (SDP), the problem $\min_{y \in Y} g(x, y)$, i.e. the problem of finding the most violated con-
 59 straint among the infinite number of constraints of the corresponding SIP problem. If $g(x, y)$ is
 60 convex in y , i.e. if $Q(x)$ is positive semidefinite (PSD), our single-level is a reformulation of (BP).
 61 This analysis yields a new solution approach, consisting in solving the single-level formulation. We
 62 note that, if $g(x, y)$ were linear in y , our reformulation would be the same as the one mentioned in
 63 [6, Section 1.3]. Although an extension to nonlinear perturbations is briefly outlined in [6, Section
 64 1.4], the specific case of quadratic perturbations over an uncertainty polytope is not considered.

65 Our second contribution is a tailored cutting plane (CP) algorithm. While such algorithms are
 66 well known in SIP, we prove its convergence and derive a new convergence rate in terms of the

67 number of iterations, under the additional assumptions that F is strongly convex and that there
68 exists an upper-level solution strictly satisfying the constraint involving the lower-level problem.

69 The rest of the paper is organized as follows. We review the relevant literature in Section 2. A
70 single-level restriction/reformulation of problem (BP) is introduced and discussed in Section 3. A
71 tailored CP algorithm for solving formulation (BP) directly is presented in Section 4. Applications
72 are introduced in Section 5. Numerical results, obtained by applying both solution approaches to
73 these applications, are presented in Section 6: our results illustrate the interest of the proposed
74 method. Finally, Section 7 concludes the paper.

75 **2. Literature review.** Bilevel quadratic problems (BQPs) are bilevel problems having either
76 one or both the objective functions which can be expressed as quadratic functions. In [4] a BQP
77 having a linear upper-level problem and a convex quadratic lower level is considered, and a branch-
78 and-bound algorithm to solve it is presented. In [33], an ergodic branch-and-bound method is
79 introduced to solve mixed-integer BQPs, having a convex lower-level problem, which is thus replaced
80 by its KKT optimality conditions. In [27], a more general class of BQPs is considered, by allowing
81 some (not necessarily convex) quadratic upper-level constraints and some convex quadratic functions
82 in lower-level constraints. After the reformulation of the problem into a non-convex quadratic single-
83 level problem by replacing its lower level by its KKT conditions (which is possible as they assume
84 to know a sufficiently large number that bounds the Lagrange multipliers) the authors adopt the
85 successive convex relaxation method given by Kojima and Tunçel in [16] for approximating the
86 nonconvex feasible region. Then, they present two types of techniques to enhance the efficiency of
87 the method used.

88 A part of the literature focuses on general nonlinear bilevel problems. For example, in [21], the
89 authors aim at solving bilevel mixed-integer optimization problems with lower-level integer variables
90 and including nonlinear terms. They assume that, for any fixed upper-level variables, and lower-
91 level integer variables, the lower-level problem is convex and satisfies Slater condition. In order to
92 solve these bilevel problems, the authors consider an approximate projection-based algorithm for
93 mixed-integer linear bilevel programming problems introduced by Yue et al. [34] and propose a
94 way of making it exact under the additional assumption that continuous upper-level variables do
95 not appear in lower-level constraints.

96 A nonconvex lower-level problem is considered in both [19, 22], as well as in [3]. In particular,
97 in [19] a BP problem having closed convex feasible sets both in the upper and in the lower level
98 (the lower-level one assumed not dependent on the upper-level variables), but eventually non-
99 convex objective functions in both levels is reformulated into a single-level problem, using the
100 so-called optimal value function transformation. To deal with the non-smoothness introduced by
101 the optimal value function, a smoothing projected gradient algorithm is proposed and used to solve
102 the bilevel problem if a calmness condition holds, which is a strong assumption, and an approximate
103 bilevel program otherwise. In [22], a bounding algorithm for the global solution of nonlinear bilevel
104 programs involving non-convex functions in both the upper and lower levels is presented. The
105 algorithm is rigorous and terminates finitely to a point that satisfies ϵ -optimality in both upper and
106 lower-level problems. This is possible using the optimal value function of the lower-level problem
107 and a piecewise, yet discontinuous, approximation of it. Previously, Bard [3] proposed an algorithm
108 (not guaranteed to be convergent) based on a grid search between a lower and an upper bound
109 of the optimal value of a bilevel problem (max-max) without upper-level constraints. The upper
110 bound is found by solving a relaxation obtained replacing the lower level with its KKT conditions.
111 The lower bound is obtained solving the lower level for a fixed value of the upper-level variables
112 (i.e. $x = x_0$), and then computing the value of the upper-level function in the point $(x_0, \phi(x_0))$.

113 This paper focuses on a particular class of BP problems, where there is no *argmin* operator,
 114 but a constraint in the upper level involving the lower-level problem's value. As mentioned before,
 115 such bilevel programs can be obtained by reformulating SIP problems having an infinite number
 116 of quadratically parametrized constraints. To solve SIP problems, discretization methods, CP
 117 methods, and other hybrid methods are used in the literature. The discretization approach [13, 26]
 118 consists in replacing the infinite constraint parameter set by a finite subset which samples it finely:
 119 this leads to a relaxation of the original problem, the value of which converges towards the value
 120 of the original problem when the mesh gets finer. This method is commonly used for parameters
 121 sets of low dimensions, but deals with the curse of dimensionality when the number of parameters
 122 increases. Instead of using a fixed subset of constraints, the CP approach [15] consists in iteratively
 123 generating and adding constraints. The CP algorithm and its refined variants, as the accelerated
 124 central CP algorithm for instance, are major techniques used for solving linear, quadratic, and
 125 convex SIP problems [17, 10, 8].

126 In this paper, we introduce a tailored CP algorithm which directly solve formulation (BP), and
 127 we prove that it is convergent. We also do a step further, by proving a rate of convergence for CP
 128 valid for a specific setting. Our convergence rate is directly related to the iteration index k , which is
 129 something new w.r.t. what is usually proved in SIP literature, where the linear rate of convergence
 130 is related to an index which is not *controlled* by the index k (see [23, Theorem 4.3]).

131 Another class of algorithms for SIP is based on Lagrangian penalty functions and Trust-Region
 132 methods [9, 28]. However, in the context of problem (BP), they would require to compute the set of
 133 all local minima of problem $\min_{y \in Y} g(x, y)$. In the case where g is not convex with respect to variables
 134 y , the enumeration of all local minima is intractable even for medium-scale instances.

135 **3. Single-level restriction/reformulation via dual approach.** A possible way to deal
 136 with the bilevel problem (BP) is what we call *dual approach*, which consists in replacing the con-
 137 straint involving the quadratic lower-level problem with one involving its dual. We obtain a strong
 138 dual from an SDP relaxation of the lower-level problem (or a reformulation if the latter is convex).
 139 We recall that the lower-level problem of (BP), for any $x \in \mathcal{X}$, reads:

$$140 \quad (\mathbf{P}_x) \quad \begin{cases} \min_{y \in \mathbb{R}^n} & \frac{1}{2}y^\top Q(x)y + q(x)^\top y \\ \text{s.t.} & a_j^\top y \leq b_j, \quad \forall j \in \{1, \dots, r\}, \end{cases}$$

141 where the objective function $f(x, y) = \frac{1}{2}y^\top Q(x)y + q(x)^\top y$ is convex if $Q(x)$ is PSD. In Section 3.1,
 142 we introduce the classical SDP relaxation (reformulation, if the lower level is convex) of the lower-
 143 level problem regularized by a ball constraint and then, in Section 3.2, we introduce the SDP dual
 144 of this relaxation (reformulation resp.). Finally, in Section 3.3 we present a single-level formulation
 145 obtained applying the so-called dual approach to the bilevel problem (BP). This formulation is a
 146 reformulation of (BP) if $Q(x)$ is PSD for any $x \in \mathcal{X}$. Otherwise, it is a restriction.

147 **3.1. SDP relaxation/reformulation of the lower-level problem.** In this section, we
 148 reason for any fixed value of the upper-level decision vector $x \in \mathcal{X}$. Let us define the following
 149 matrices:

$$150 \quad \begin{aligned} & \bullet \mathcal{Q}(x) = \frac{1}{2} \begin{pmatrix} Q(x) & q(x) \\ q(x)^\top & 0 \end{pmatrix}, \\ 151 & \bullet \mathcal{A}_j = \frac{1}{2} \begin{pmatrix} 0_n & a_j \\ a_j^\top & 0 \end{pmatrix}, \quad \forall j \in \{1, \dots, r\}, \end{aligned}$$

152 where 0_n is the $n \times n$ null matrix. We denote by $\langle A, B \rangle = \text{Tr}(A^\top B)$ the Froebenius product of two
 153 square matrices A and B with same size. With this notation, under Assumption 4, the problem

$$154 \quad (3.1) \quad \left\{ \begin{array}{ll} \min_{Y \in \mathbb{R}^{(n+1) \times (n+1)}} & \langle \mathcal{Q}(x), Y \rangle \\ \text{s.t.} & \langle \mathcal{A}_j, Y \rangle \leq b_j \quad \forall j \in \{1, \dots, r\} \\ & \text{Tr}(Y) \leq 1 + \rho^2 \\ & Y_{n+1, n+1} = 1 \\ & Y \succeq 0 \\ & \text{rank}(Y) = 1, \end{array} \right.$$

is a reformulation of (\mathbf{P}_x) , because any feasible matrix Y has the form $Y = \begin{pmatrix} y \\ 1 \end{pmatrix} \begin{pmatrix} y \\ 1 \end{pmatrix}^\top$ with $y \in \mathcal{F}$,
 and, therefore, $\langle \mathcal{Q}(x), Y \rangle = f(x, y)$. The constraint $\text{Tr}(Y) \leq 1 + \rho^2$, derives from Assumption 4 as
 follows:

$$\|y\|_2^2 \leq \rho^2 \Leftrightarrow \text{Tr}(yy^\top) \leq \rho^2 \Leftrightarrow \text{Tr}(Y) \leq \rho^2 + 1,$$

155 being $\text{Tr}(Y) = \text{Tr}(yy^\top) + 1$. This constraint does not play any role at this point, but will be useful
 156 thereafter to come up with a dual SDP problem with no duality gap (see Section 3.2). If we relax
 157 the non-convex constraint $\text{rank}(Y) = 1$ in (3.1), we obtain:

$$158 \quad (\mathbf{SDP}_x) \quad \left\{ \begin{array}{ll} \min_{Y \in \mathbb{R}^{(n+1) \times (n+1)}} & \langle \mathcal{Q}(x), Y \rangle \\ \text{s.t.} & \langle \mathcal{A}_j, Y \rangle \leq b_j \quad \forall j \in \{1, \dots, r\} \\ & \text{Tr}(Y) \leq 1 + \rho^2 \\ & Y_{n+1, n+1} = 1 \\ & Y \succeq 0, \end{array} \right.$$

159 which is a SDP relaxation of (\mathbf{P}_x) , as proved in the following Lemma 3.1. If $Q(x)$ is PSD, Lemma 3.1
 160 states that (\mathbf{SDP}_x) is a reformulation of (\mathbf{P}_x) , the rank-constraint notwithstanding.

161 LEMMA 3.1. *Under Assumption 4, $\text{val}(\mathbf{SDP}_x) \leq \text{val}(\mathbf{P}_x)$. If $Q(x)$ is PSD, then $\text{val}(\mathbf{SDP}_x) =$
 162 $\text{val}(\mathbf{P}_x)$.*

163 For a sake of completeness, we give a proof of this standard lemma.

Proof. The inequality $\text{val}(\mathbf{SDP}_x) \leq \text{val}(\mathbf{P}_x)$ follows from the relaxation of the rank-constraint.
 We now assume that $Q(x)$ is PSD and prove that $\text{val}(\mathbf{SDP}_x) \geq \text{val}(\mathbf{P}_x)$ holds. Given a matrix Y
 feasible for (\mathbf{SDP}_x) , we denote by $u_1, \dots, u_{n+1} \in \mathbb{R}^{n+1}$ a basis of eigenvectors of Y (which is PSD)
 and their respective eigenvalues $v_1, \dots, v_{n+1} \in \mathbb{R}_+$. Let us introduce the two following index sets:

$$I = \{i \in \{1, \dots, n+1\} : (u_i)_{n+1} \neq 0\} \text{ and } J = \{i \in \{1, \dots, n+1\} : (u_i)_{n+1} = 0\}.$$

164 We have then: $I \cup J = \{1, \dots, n+1\}$. Moreover,

- 165 • if $i \in I$: we define the nonnegative scalar $\mu_i = v_i (u_i)_{n+1}^2$ and $y_i \in \mathbb{R}^n$ s.t. $u_i = (u_i)_{n+1} \begin{pmatrix} y_i \\ 1 \end{pmatrix}$
- 166 • if $i \in J$: we define the nonnegative scalar $\nu_i = v_i$ and $z_i \in \mathbb{R}^n$ s.t. $u_i = \begin{pmatrix} z_i \\ 0 \end{pmatrix}$.

With this notation, we have that

$$Y = \sum_{i=1}^{n+1} v_i u_i u_i^\top = \sum_{i \in I} v_i (u_i)_{n+1}^2 \begin{pmatrix} y_i \\ 1 \end{pmatrix} \begin{pmatrix} y_i \\ 1 \end{pmatrix}^\top + \sum_{i \in J} v_i \begin{pmatrix} z_i \\ 0 \end{pmatrix} \begin{pmatrix} z_i \\ 0 \end{pmatrix}^\top$$

$$= \sum_{i \in I} \mu_i \begin{pmatrix} y_i y_i^\top & y_i \\ y_i^\top & 1 \end{pmatrix} + \sum_{i \in J} \nu_i \begin{pmatrix} z_i z_i^\top & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{pmatrix},$$

167 where $\mathbf{0}$ is the null n -dimensional vector, not to be confused with 0_n , the $n \times n$ null matrix. Let
 168 us define the vector $\bar{y} = \sum_{i \in I} \mu_i y_i$. Its objective value in (\mathbf{P}_x) is smaller than the objective value of
 169 Y in (\mathbf{SDP}_x) . In fact:

$$170 \quad (3.2) \quad \langle \mathcal{Q}(x), Y \rangle = \sum_{i \in I} \mu_i f(x, y_i) + \frac{1}{2} \sum_{i \in J} \nu_i z_i^\top Q(x) z_i \geq \sum_{i \in I} \mu_i f(x, y_i) \geq f(x, \sum_{i \in I} \mu_i y_i) = f(x, \bar{y}).$$

171 The first inequality is due to $Q(x) \succeq 0$ and $\nu_i \geq 0$. The second inequality derives from $\sum_{i \in I} \mu_i =$
 172 $Y_{n+1, n+1} = 1$, and from the convexity of function f_x (Jensen inequality). Moreover, since Y is
 173 feasible in (\mathbf{SDP}_x) , for each $j \in \{1, \dots, r\}$ we have $b_j \geq \langle \mathcal{A}_j, Y \rangle = \sum_{i \in I} \mu_i a_j^\top y_i = a_j^\top \bar{y}$, which
 174 means that \bar{y} is feasible in (\mathbf{P}_x) too. This implies that $f(x, \bar{y}) \geq \text{val}(\mathbf{P}_x)$ and together with (3.2),
 175 that $\langle \mathcal{Q}(x), Y \rangle \geq \text{val}(\mathbf{P}_x)$. This being true for any matrix Y feasible in (\mathbf{SDP}_x) , we conclude that
 176 $\text{val}(\mathbf{SDP}_x) \geq \text{val}(\mathbf{P}_x)$. This proves that $\text{val}(\mathbf{SDP}_x) = \text{val}(\mathbf{P}_x)$. \square

177 **3.2. Dual SDP problem.** As already done in Section 3.1, also in this section we reason for
 178 any fixed value of $x \in \mathcal{X}$. Let E be a $(n+1) \times (n+1)$ matrix s.t. $E_{n+1, n+1} = 1$ and $E_{ij} = 0$
 179 everywhere else. Let I_{n+1} be the $(n+1) \times (n+1)$ identity matrix. The following SDP problem

$$180 \quad (\mathbf{DSDP}_x) \quad \begin{cases} \max_{\lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}} & -b^\top \lambda - \alpha(1 + \rho^2) - \beta \\ \text{s.t.} & \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \succeq 0, \end{cases}$$

181 is the dual of problem (\mathbf{SDP}_x) , as the following proposition states.

182 **PROPOSITION 3.2.** *Formulations (\mathbf{SDP}_x) and (\mathbf{DSDP}_x) are a primal-dual pair of SDP problems*
 183 *and strong duality holds, i.e., $\text{val}(\mathbf{SDP}_x) = \text{val}(\mathbf{DSDP}_x)$.*

184 *Proof.* The Lagrangian of problem (\mathbf{SDP}_x) is defined over $Y \in S_{n+1}^+(\mathbb{R})$, $\lambda \in \mathbb{R}_+^r$, $\alpha \in \mathbb{R}_+$, $\beta \in$
 185 \mathbb{R} and reads

$$186 \quad \begin{aligned} L_x(Y, \lambda, \alpha, \beta) &= \langle \mathcal{Q}(x), Y \rangle + \sum_{j=1}^r [\lambda_j (\langle \mathcal{A}_j, Y \rangle - b_j)] + \alpha(\text{Tr}(Y) - 1 - \rho^2) + \beta(Y_{n+1, n+1} - 1) \\ &= - \sum_{j=1}^r \lambda_j b_j - \alpha(1 + \rho^2) - \beta + \left\langle \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E, Y \right\rangle. \end{aligned}$$

187 The Lagrangian dual problem of (\mathbf{SDP}_x) is:

$$188 \quad \max_{\substack{\lambda \in \mathbb{R}_+^r \\ \alpha \in \mathbb{R}_+ \\ \beta \in \mathbb{R}}} \min_{Y \in S_{n+1}^+(\mathbb{R})} L_x(Y, \lambda, \alpha, \beta).$$

189 According to equality above, it can thus be written as

$$190 \quad \max_{\substack{\lambda \in \mathbb{R}_+^r \\ \alpha \in \mathbb{R}_+ \\ \beta \in \mathbb{R}}} \left(- \left(\sum_{j=1}^r \lambda_j b_j + \alpha(1 + \rho^2) + \beta \right) + \min_{Y \in S_{n+1}^+(\mathbb{R})} \left\langle \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E, Y \right\rangle \right).$$

191 We notice that

$$192 \quad \min_{Y \in S_{n+1}^+(\mathbb{R})} \left\langle \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E, Y \right\rangle = \begin{cases} 0 & \text{if } \left(\mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \right) \succeq 0 \\ -\infty & \text{otherwise.} \end{cases}$$

193 This proves that the dual problem of (SDP_x) reads

$$194 \quad \begin{cases} \max_{\lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}} & -b^\top \lambda - \alpha(1 + \rho^2) - \beta \\ \text{s.t.} & \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \succeq 0, \end{cases}$$

195 which is the formulation (DSDP_x) . To prove that $\text{val}(\text{SDP}_x) = \text{val}(\text{DSDP}_x)$, we prove that Slater
196 condition holds for the dual problem (DSDP_x) , exploiting the Lagrangian multiplier associated
197 to the constraint $\text{Tr}(Y) \leq 1 + \rho^2$. In fact, Slater condition is a sufficient condition for strong
198 duality [31]. We denote by m_x the minimum eigenvalue of $\mathcal{Q}(x)$. By definition of m_x , matrix
199 $\mathcal{Q}(x) + (1 - m_x)I_{n+1}$ is positive definite. This is why $(\lambda, \alpha, \beta) = (0, \dots, 0, 1 - m_x, 0)$ is a strictly
200 feasible point of (DSDP_x) . Hence, Slater condition holds. \square

201 **3.3. SDP restriction/reformulation of the bilevel problem.** Leveraging on Section 3.1
202 and Section 3.2, which focus on the lower-level problem (P_x) , its SDP relaxation (SDP_x) and the
203 respective dual problem (DSDP_x) , we propose a single-level restriction of the bilevel programming
204 problem (BP) . It is a reformulation of (BP) if $Q(x)$ is PSD for any $x \in \mathcal{X}$.

205 **THEOREM 3.3.** *The single-level formulation*

$$206 \quad (\text{BPR}) \quad \begin{cases} \min_{x, \lambda, \alpha, \beta} & F(x) \\ \text{s.t.} & x \in \mathcal{X} \\ & h(x) \leq -\lambda^\top b - \alpha(1 + \rho^2) - \beta \\ & \mathcal{Q}(x) + \sum_j \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \succeq 0 \\ & x \in \mathbb{R}^m, \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}, \end{cases}$$

208 *is a restriction of the bilevel programming problem (BP) . If $Q(x)$ is PSD for any $x \in \mathcal{X}$, (BPR) is*
209 *a reformulation of (BP) .*

210 *Proof.* Being $\text{Feas}(\text{BP})$ and $\text{Feas}(\text{BPR})$ the feasible sets of (BP) and (BPR) respectively, since
211 (BP) and (BPR) share the same objective function, proving the following implication for any $x \in \mathbb{R}^m$

$$212 \quad (3.3) \quad (\exists \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R} : (x, \lambda, \alpha, \beta) \in \text{Feas}(\text{BPR})) \implies x \in \text{Feas}(\text{BP}),$$

213 will prove the first part of the theorem. For any $x \in \mathcal{X}$, we have:

$$214 \quad (3.4) \quad h(x) \leq \text{val}(\text{SDP}_x) \implies h(x) \leq \text{val}(P_x) \iff x \in \text{Feas}(\text{BP}),$$

215 where the first implication stems from Lemma 3.1, which stipulates that $\text{val}(\text{SDP}_x) \leq \text{val}(P_x)$.
216 Applying Proposition 3.2, we obtain that:

$$217 \quad (3.5) \quad h(x) \leq \text{val}(\text{SDP}_x) \iff h(x) \leq \text{val}(\text{DSDP}_x).$$

218 For any $x \in \mathcal{X}$, we have that

$$219 \quad (3.6) \quad h(x) \leq \text{val}(\text{DSDP}_x) \iff \exists \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R} : \begin{cases} h(x) \leq -\lambda^\top b - \alpha(1 + \rho^2) - \beta \\ \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \succeq 0 \end{cases}$$

220 The equivalence (3.6) just expresses the fact that the maximization problem (DSDP_x) has a value
 221 exceeding $h(x)$ if and only if it has a feasible solution with value exceeding $h(x)$. Hence, from (3.5),
 222 and (3.6), the following equivalences hold:

$$223 \quad (3.7) \quad h(x) \leq \text{val}(\text{SDP}_x) \iff \exists \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R} : \begin{cases} h(x) \leq -\lambda^\top b - \alpha(1 + \rho^2) - \beta \\ \mathcal{Q}(x) + \sum_{j=1}^r \lambda_j \mathcal{A}_j + \alpha I_{n+1} + \beta E \succeq 0 \end{cases}$$

$$224 \quad \iff \exists \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}, (x, \lambda, \alpha, \beta) \in \text{Feas}(\text{BPR}).$$

226 The equivalence (3.7), together with implication (3.4), proves the implication (3.3).

227 If $\mathcal{Q}(x)$ is PSD for any $x \in \mathcal{X}$, we can replace the implication (3.4) by the equivalence

$$228 \quad (3.8) \quad h(x) \leq \text{val}(\text{SDP}_x) \iff h(x) \leq \text{val}(\text{P}_x) \iff x \in \text{Feas}(\text{BP}).$$

229 This, together with equivalence (3.7), proves that

$$\exists \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R} : (x, \lambda, \alpha, \beta) \in \text{Feas}(\text{BPR}) \iff x \in \text{Feas}(\text{BP}),$$

230 meaning that (BPR) is a reformulation of (BP), since the objective function is the same. \square

231 Assumptions 1, 2, and 3 implies that the single-level problem (BPR) is convex. Let us recall
 232 the following definition of *semidefinite representable* (SDr) functions

233 DEFINITION 3.4 ([25]). A convex (resp. concave) function f is SDr if and only if its epigraph,
 234 i.e., $(t, x) : f(x) \leq t$ (resp. the hypograph $(t, x) : t \leq f(x)$), is SDr [7].

235 Thus, we further remark that formulation (BPR) is a SDP problem if set \mathcal{X} is SDr, as well as
 236 functions $F(x)$, and $h(x)$.

237 **4. Cutting plane algorithm.** In order to benchmark the results and the performance of
 238 the single-level approach proposed in Section 3, we introduce in this section a CP algorithm for
 239 solving the bilevel formulation (BP) directly. We also include a proof of convergence for this tailored
 240 algorithm in Section 4.1, as well as a convergence rate in Section 4.2, obtained by introducing a
 241 dual view of the CP algorithm. We make the following further assumption on set \mathcal{X} :

242 ASSUMPTION 5. The set \mathcal{X} is compact.

Algorithm 4.1 CP algorithm for (BP)

-
- 1: Let $k = 0$. Initialize the relaxation R_k of the bilevel problem (BP), obtained by considering the upper-level problem only.
 - 2: **while** true **do**
 - 3: Solve R_k , obtaining an optimal solution x^k .
 - 4: Compute an optimal solution y^k of the lower-level problem for $x = x^k$.
 - 5: **if** $h(x^k) \leq \frac{1}{2}(y^k)^\top Q(x^k)y^k + q(x^k)^\top y^k$ **then**
 - 6: Return (x^k, y^k) .
 - 7: **else**
 - 8: Define R_{k+1} as R_k with the adjoined inequality:

$$(4.1) \quad h(x) \leq \frac{1}{2}(y^k)^\top Q(x)y^k + q(x)^\top y^k.$$

- 9: $k := k + 1$
 - 10: **end if**
 - 11: **end while**
-

243 At the first iteration of Algorithm 4.1, the relaxed problem R_0 is given by:

$$244 \quad (4.2) \quad \min_{x \in \mathcal{X}} F(x),$$

245 which considers minimizing the upper-level objective function subject to the upper-level constraints
 246 only. This problem has a finite value according to the compactness of set \mathcal{X} .

247 At each iteration, Algorithm 4.1 defines the feasible set of the upper-level problem by means
 248 of cuts in the upper-level variables x . The resulting R_k problems are relaxations of (BP), and their
 249 feasible sets are decreasing in the sense of the inclusion, bounded, because included in the feasible
 250 set of R_0 , and closed as intersections of closed sets. Thus, each problem R_k admits a minimum.
 251 Moreover, the sequence $(F(x^k))$ is increasing, and $F(x^k) \leq \text{val}(\text{BP})$ holds for any k . At step 4, the
 252 problem solved to find a new cutting plane is

$$253 \quad (P_{x^k}) \quad \min_{y \in \mathbb{R}^n} \left\{ \frac{1}{2}y^\top Q(x^k)y + q(x^k)^\top y \mid Ay \leq b \right\}.$$

254 This problem is a quadratic program that is either convex or non-convex depending on the positive
 255 semi-definiteness of the constant matrix $Q(x^k)$. In order to find global optima of (P_{x^k}) , regardless
 256 of the definiteness of $Q(x^k)$ (in turn depending on the value of x^k), a global optimization algorithm
 257 should be employed. Step 6 returns the optimal solution of the bilevel formulation (BP).

258 **4.1. Convergence proof.** In this section, a convergence proof for Algorithm 4.1 is given.
 259 First of all, let us define the negative part of a function f as $f^- := \max(0, -f)$. Since $Q(x)$ and
 260 $q(x)$ are linear w.r.t. x , the function $f : (x, y) \mapsto \frac{1}{2}y^\top Q(x)y + q(x)^\top y$ is continuously differentiable,
 261 and therefore Lipschitz-continuous on the compact set $\mathcal{X} \times \mathcal{F}$ (see Assumption 4 and 5), with $L > 0$
 262 an associated Lipschitz constant.

Moreover, $x \mapsto \text{val}(P_x)$ is continuous. To show this, let us consider any $\omega > 0$ and any pair
 $(x, \tilde{x}) \in \mathcal{X}^2$ s.t. $\|x - \tilde{x}\| \leq \frac{\omega}{L}$. We define $y \in \mathcal{F}$ an optimal solution of (P_x) , i.e., $\text{val}(P_x) = f(x, y)$,
 and $\tilde{y} \in \mathcal{F}$ an optimal solution of $(P_{\tilde{x}})$, i.e., $\text{val}(P_{\tilde{x}}) = f(\tilde{x}, \tilde{y})$. By definition of $\text{val}(P_{\tilde{x}})$ and using

the Lipschitz continuity of f , we know that

$$\text{val}(P_{\tilde{x}}) \leq f(\tilde{x}, y) \leq f(x, y) + L \left\| \begin{pmatrix} x - \tilde{x} \\ y - y \end{pmatrix} \right\| \leq \text{val}(P_x) + L \|x - \tilde{x}\| \leq \text{val}(P_x) + \omega,$$

and, symmetrically, that

$$\text{val}(P_x) \leq f(x, \tilde{y}) \leq f(\tilde{x}, \tilde{y}) + L \left\| \begin{pmatrix} x - \tilde{x} \\ \tilde{y} - \tilde{y} \end{pmatrix} \right\| \leq \text{val}(P_{\tilde{x}}) + L \|x - \tilde{x}\| \leq \text{val}(P_{\tilde{x}}) + \omega.$$

263 Thus, $|\text{val}(P_x) - \text{val}(P_{\tilde{x}})| \leq \omega$, which proves that the value function $x \mapsto \text{val}(P_x)$ is continuous at
264 any $x \in \mathcal{X}$. Based on these observations, we prove the convergence of the algorithm.

265 **THEOREM 4.1.** *Under Assumptions 4 and 5 Algorithm 4.1 either terminates in $K \in \mathbb{N}^*$ iter-*
266 *ations, in which case x^K is the solution of (BP), or generates an infinite sequence $(x^k)_{k \in \mathbb{N}^*}$ with*
267 *the following convergence guarantees:*

- 268 • feasibility error: $\epsilon_k = (\text{val}(P_{x^k}) - h(x^k))^- \rightarrow 0$,
- 269 • objective error: $\delta_k = \text{val}(\text{BP}) - F(x^k) \rightarrow 0$.

270 *Proof.* If Algorithm 4.1 terminates at iteration $K \in \mathbb{N}^*$, x^K is feasible in (BP), i.e., $x^K \in \mathcal{X}$ and
271 $\text{val}(P_{x^K}) \geq h(x^K)$, which implies that $F(x^K) \geq \text{val}(\text{BP})$. At the same time $F(x^K) = \text{val}(R_K) \leq$
272 $\text{val}(\text{BP})$, being R_K a relaxation of (BP) by definition. Thus, $F(x^K) = \text{val}(\text{BP})$, and x^K is an optimal
273 solution of (BP).

274 Let us suppose now that the stopping test is never satisfied. In this context, we prove first
275 the convergence of the feasibility error ϵ_k towards 0. For any $k \in \mathbb{N}^*$, we have that $\text{val}(P_{x^k}) =$
276 $\frac{1}{2} y^{k\top} Q(x^k) y^k + q(x^k)^\top y^k = f(x^k, y^k)$, thus $\epsilon_k = (f(x^k, y^k) - h(x^k))^-$. Since f , h and the negative
277 part function are continuous, and since both x^k and y^k are bounded, the sequence ϵ_k is also
278 bounded. According to Bolzano-Weierstrass theorem [1], this bounded sequence has at least a
279 convergent sub-sequence. In the following, we define any convergent sub-sequence extracted from
280 ϵ_k as $\epsilon_{\psi_0(k)}$, where $\psi_0 : \mathbb{N}^* \mapsto \mathbb{N}^*$ is an increasing application. Defining as $\epsilon_* \in \mathbb{R}$ the limit of this
281 convergent sub-sequence, we will show that this limit value is in fact 0.

282 The sequence $(y^{\psi_0(k)}, \epsilon_{\psi_0(k)})$ is a sub-sequence of the bounded sequence (y^k, ϵ_k) , therefore it is
283 bounded. According to the Bolzano-Weierstrass theorem, the sequence $(y^{\psi_0(k)}, \epsilon_{\psi_0(k)})$ has thus a
284 convergent sub-sequence $(y^{\psi(k)}, \epsilon_{\psi(k)})$. Since $\epsilon_{\psi(k)}$ is a convergent sub-sequence of $\epsilon_{\psi_0(k)}$, $\epsilon_{\psi(k)} \rightarrow \epsilon_*$
285 holds. Because $\psi(k-1) < \psi(k)$ by definition of ψ , the cut related to $y^{\psi(k-1)}$ is a constraint of
286 problem $R_{\psi(k)}$ (added by Algorithm 4.1 at iteration $k-1$). Thus, $f(x^{\psi(k)}, y^{\psi(k-1)}) - h(x^{\psi(k)}) \geq 0$,
287 and

$$288 \quad \begin{aligned} f(x^{\psi(k)}, y^{\psi(k)}) - h(x^{\psi(k)}) &= f(x^{\psi(k)}, y^{\psi(k)}) - f(x^{\psi(k)}, y^{\psi(k-1)}) + f(x^{\psi(k)}, y^{\psi(k-1)}) - h(x^{\psi(k)}) \\ &\geq f(x^{\psi(k)}, y^{\psi(k)}) - f(x^{\psi(k)}, y^{\psi(k-1)}). \end{aligned}$$

Being the negative part function decreasing,

$$\epsilon_{\psi(k)} = \left(f(x^{\psi(k)}, y^{\psi(k)}) - h(x^{\psi(k)}) \right)^- \leq \left(f(x^{\psi(k)}, y^{\psi(k)}) - f(x^{\psi(k)}, y^{\psi(k-1)}) \right)^-.$$

289 Therefore

$$290 \quad (4.3) \quad \epsilon_{\psi(k)} \leq \left| f(x^{\psi(k)}, y^{\psi(k)}) - f(x^{\psi(k)}, y^{\psi(k-1)}) \right|.$$

291 From the fact that f is L -Lipschitz continuous, and Eq. (4.3) we deduce that

$$292 \quad (4.4) \quad \epsilon_{\psi(k)} \leq L \left\| \begin{pmatrix} x^{\psi(k)} \\ y^{\psi(k)} \end{pmatrix} - \begin{pmatrix} x^{\psi(k-1)} \\ y^{\psi(k-1)} \end{pmatrix} \right\| = L \|y^{\psi(k)} - y^{\psi(k-1)}\|.$$

293 As $y^{\psi(k)}$ is convergent, we know that $\|y^{\psi(k)} - y^{\psi(k-1)}\| \rightarrow 0$. Being $\epsilon_{\psi(k)}$ nonnegative, we deduce
294 from Eq. (4.4) that $\epsilon_{\psi(k)} \rightarrow 0$, and thus, $\epsilon_* = 0$.

295 We proved that the sequence ϵ_k is bounded, and that any converging sub-sequence converge
296 towards 0, thus we can conclude that ϵ_k converges towards 0 itself, according to a well-known result
297 in analysis [1]. Based on this first result, we are now going to prove the second point, i.e., the
298 convergence of objective error. We know that

$$299 \quad (4.5) \quad \forall k \in \mathbb{N}^* \quad F(x^k) \in [F(x^1), \text{val}(\mathbf{BP})],$$

300 therefore the increasing sequence $F(x^k)$ is bounded, and thus, converging. Since x^k bounded, we
301 can derive a converging sub-sequence $x^{\phi(k)} \rightarrow x^*$ with $\phi : \mathbb{N}^* \mapsto \mathbb{N}^*$ being an increasing function.

302 The associated feasibility error is $\epsilon_{\phi(k)} = (\text{val}(P_{x^{\phi(k)}}) - h(x^{\phi(k)}))^-$. On the one hand, being $\epsilon_{\phi(k)}$ a
303 sub-sequence of ϵ_k which has been proven to converge towards zero, $\epsilon_{\phi(k)} \rightarrow 0$. On the other hand,
304 $\epsilon_{\phi(k)} \rightarrow (\text{val}(P_{x^*}) - h(x^*))^-$ holds by continuity of $x \mapsto \text{val}(P_x)$ and h . By uniqueness of the limit,
305 $(\text{val}(P_{x^*}) - h(x^*))^- = 0$. Therefore, $x^* \in \mathcal{X}$ is feasible in (BP) and $F(x^*) \geq \text{val}(\mathbf{BP})$. From (4.5)
306 we also know that $F(x^*) \leq \text{val}(\mathbf{BP})$, and thus $F(x^*) = \text{val}(\mathbf{BP})$. We can conclude that $F(x^k)$ is
307 bounded and admits a unique limit point which is $\text{val}(\mathbf{BP})$. Hence, $\delta_k \rightarrow 0$. \square

308 **4.2. A convergence rate for the CP algorithm.** In this section, we give a convergence
309 rate of the CP algorithm 4.1, under two additional assumptions on the bilevel problem. First of all,
310 let us reformulate the bilevel problem, by moving the function $h(x)$ within the lower-level problem:

$$311 \quad (\mathbf{BP}) \quad \begin{cases} \min_{x \in \mathcal{X}} & F(x) \\ \text{s.t.} & 0 \leq \min_{y \in \mathbb{R}^n} \{ \frac{1}{2} y^\top Q(x) y + q(x)^\top y - h(x) \mid y \in \mathcal{F} \}. \end{cases}$$

312

We introduce then the matrix $\mathcal{G}(x) = \frac{1}{2} \begin{pmatrix} Q(x) & q(x) \\ q(x)^\top & -2h(x) \end{pmatrix} = \mathcal{Q}(x) - \begin{pmatrix} 0_n & 0 \\ 0 & h(x) \end{pmatrix}$ and we define
the set

$$\mathcal{P} = \left\{ M(y) = \begin{pmatrix} yy^\top & y \\ y^\top & 1 \end{pmatrix} : y \in \mathcal{F} \right\} \subset \mathbb{R}^{(n+1) \times (n+1)}.$$

313 With this notation, we acknowledge that (BP) can be formulated as

$$314 \quad (\mathbf{SIP}) \quad \begin{cases} \min_{x \in \mathcal{X}} & F(x) \\ \text{s.t.} & 0 \leq \langle \mathcal{G}(x), Y \rangle, \forall Y \in \mathcal{P}. \end{cases}$$

315

We define as $\mathcal{K} = \text{cone}(\mathcal{P}) \subset \mathbb{R}^{(n+1) \times (n+1)}$ the convex cone generated by \mathcal{P} , and $\mathcal{L}(x, Y) = F(x) - \langle \mathcal{G}(x), Y \rangle$ the Lagrangian function defined over $\mathcal{X} \times \mathcal{K}$. We remark that for any $x \in \mathcal{X}$, the following
equality holds

$$\sup_{Y \in \mathcal{K}} \mathcal{L}(x, Y) = \begin{cases} F(x) & \text{if } 0 \leq \langle \mathcal{G}(x), Y \rangle, \forall Y \in \mathcal{P} \\ +\infty & \text{else.} \end{cases}$$

316 Hence, problem (SIP) can be expressed as the saddle-point problem $\min_{x \in \mathcal{X}} \sup_{Y \in \mathcal{K}} \mathcal{L}(x, Y)$. At this point,

317 we do the following further assumption.

318 ASSUMPTION 6. *The upper-level objective function $F(x)$ is μ -strongly-convex.*

319 Assumptions 6 is quite strong, but we remark that, if the original objective function is just convex,
 320 it is always possible to enforce this assumption by “regularizing” the bilevel problem adding a
 321 ℓ_2 penalty to the primal objective function, i.e. minimizing $F(x) + \frac{\mu}{2}\|x\|^2$ instead of $F(x)$. The
 322 Lagrangian function $\mathcal{L}(x, Y)$ is linear (thus continuous and concave) w.r.t. Y for all $x \in \mathcal{X}$ and is
 323 continuous and convex w.r.t. x for all $Y \in \mathcal{K}$. The convexity w.r.t. x follows from Assumptions 2
 324 and 3 and from the fact that $Y_{n+1, n+1} \geq 0$ for any $Y \in \mathcal{K}$. Since the set \mathcal{X} is convex (Assumption
 325 1) and the set \mathcal{K} is convex too, the Sion’s minimax theorem is applicable and the following holds:

$$326 \quad \min_{x \in \mathcal{X}} \sup_{Y \in \mathcal{K}} \mathcal{L}(x, Y) = \sup_{Y \in \mathcal{K}} \min_{x \in \mathcal{X}} \mathcal{L}(x, Y).$$

327 Defining the dual function $\theta(Y) = \min_{x \in \mathcal{X}} \mathcal{L}(x, Y)$, we know that

$$328 \quad (4.6) \quad \text{val}(\text{SIP}) = \sup_{Y \in \mathcal{K}} \theta(Y).$$

329 Notice that the dual function $\theta(Y)$ is concave, as a minimum of linear functions in Y . As a
 330 direct application of [14, Corollary VI.4.4.5], the dual function $\theta(Y)$ is differentiable because of
 331 the uniqueness of $\arg \min_{x \in \mathcal{X}} \mathcal{L}(x, Y)$, which is, in turn, a consequence of the strong convexity of
 332 $x \mapsto \mathcal{L}(x, Y)$ that follows from Assumption 6. Moreover, the gradient of the dual function is
 333 $\nabla \theta(Y) = -\mathcal{G}(x)$, where $x = \arg \min_{x \in \mathcal{X}} \mathcal{L}(x, Y)$. The differentiability of θ implies, in particular, that
 334 θ is continuous. We prove now that we can replace the sup operator with the max operator in the
 335 formulation (4.6), under the following assumption.

336 ASSUMPTION 7. *It exists $\hat{x} \in \mathcal{X}$, s.t., for all $y \in \mathcal{F}$, $g(\hat{x}, y) = \frac{1}{2}y^\top Q(\hat{x})y + q(\hat{x})^\top y - h(\hat{x}) > 0$.*

337 LEMMA 4.2. *Under Assumption 7, the dual problem of (SIP) has an optimal solution Y^* .*

338 *Proof.* We denote by $\hat{x} \in \mathcal{X}$ the primal feasible solution s.t. $g(\hat{x}, y) = \frac{1}{2}y^\top Q(\hat{x})y + q(\hat{x})^\top y -$
 339 $h(\hat{x}) > 0$ for all $y \in \mathcal{F}$. Since the set \mathcal{F} is compact and the function $y \mapsto g(\hat{x}, y)$ is continuous and
 340 positive, it exists $c > 0$ s.t. $g(\hat{x}, y) \geq c$ for all $y \in \mathcal{F}$. For any $Y \in \mathcal{K}$, we have that $Y = \sum_{k=1}^p \lambda_k M(y^k)$,
 341 for an integer $p \in \mathbb{N}$, vectors $y^1, \dots, y^p \in \mathcal{F}$ and nonnegative scalars $\lambda_1, \dots, \lambda_p \in \mathbb{R}_+$. Since
 342 $\langle \mathcal{G}(\hat{x}), M(y) \rangle = \frac{1}{2}y^\top Q(\hat{x})y + q(\hat{x})^\top y - h(\hat{x})$ for any $y \in \mathcal{F}$, the following holds by linearity:

$$343 \quad \langle \mathcal{G}(\hat{x}), Y \rangle = \left\langle \mathcal{G}(\hat{x}), \sum_{k=1}^p \lambda_k M(y^k) \right\rangle = \sum_{k=1}^p \lambda_k \langle \mathcal{G}(\hat{x}), M(y^k) \rangle \geq \sum_{k=1}^p \lambda_k c = Y_{n+1, n+1} c.$$

344 Moreover, by definition of θ :

$$346 \quad \theta(Y) = \min_{x \in \mathcal{X}} F(x) - \langle \mathcal{G}(x), Y \rangle \leq F(\hat{x}) - \langle \mathcal{G}(\hat{x}), Y \rangle \leq F(\hat{x}) - Y_{n+1, n+1} c,$$

348 this for any $Y \in \mathcal{K}$. We take then a maximizing sequence $(Y^k)_{k \in \mathbb{N}}$ of problem (4.6). Defining
 349 $V = \text{val}(\text{SIP})$, we know that $\theta(Y^k) \rightarrow V$ and hence, it exists $j \in \mathbb{N}$ s.t. for all $k \geq j$, $\theta(Y^k) \geq V - 1$.
 350 This implies that, for all $k \geq j$,

$$351 \quad 0 \leq Y_{n+1, n+1}^k \leq \frac{F(\hat{x}) - V + 1}{c}.$$

352 Defining $B = \frac{F(\hat{x}) - V + 1}{c}$, we deduce that $\forall k \geq j$, Y^k belongs to $B \text{conv}(\mathcal{F})$, which is compact. Thus,
 353 the sequence $(Y^k)_{k \in \mathbb{N}}$ admits an accumulation point Y^* , s.t. $\theta(Y^*) = V$ by continuity of θ . \square

354 According to this lemma, the dual version of problem (SIP) thus reads

355 (DSIP)
$$\max_{Y \in \mathcal{K}} \theta(Y).$$

356 This concave maximization problem on the convex cone \mathcal{K} is the Lagrangian dual of the problem
 357 (SIP) i.e. of the bilevel program (BP). Indeed, in this section, we are dualizing the whole bilevel
 358 problem (BP), contrary to Section 3, where we dualize the lower-level problem only. We are now
 359 going to see that the CP algorithm 4.1 can be interpreted, from a dual perspective, as a cone
 360 constrained Fully Corrective Frank-Wolfe (FCFW) algorithm [20] solving the dual problem (DSIP).
 361 We prove that during the execution of the CP algorithm 4.1, the dual variables obtained when
 362 solving the relaxation R_k instantiate the iterates of a FCFW algorithm. In the following, the sets
 363 $B_k \subset \mathbb{R}^{n+1 \times n+1}$ are finite sets, composed of rank-one matrices of the form $M(y)$.

364 First, the initialization of the CP can be seen, in the dual perspective, as the initialization of
 a Frank-Wolfe type algorithm, with $B_0 \leftarrow \emptyset$. Then, the generic iteration k is described in Table 1.

	Primal perspective: CP	Link	Dual perspective: FCFW
Step 1	Solve R_k and store the solution x^k	Duality	Solve the dual problem on $\text{cone}(B_k)$, i.e. $\max_{Y \in \text{cone}(B_k)} \theta(Y),$ store the solution Y^k , the associated x^k and the gradient $\nabla \theta(Y^k) = -\mathcal{G}(x^k)$
Step 2	Solve the lower-level problem P_{x^k} $\min_{y \in \mathcal{F}} \frac{1}{2} y^\top Q(x^k) y + q(x^k)^\top y$ and store the solution y^k	$Z^k = M(y^k)$	Solve the problem $\max_{Z \in \mathcal{P}} \langle \nabla \theta(Y^k), Z \rangle$ and store the solution Z^k
Step 3a	If $h(x^k) \leq \frac{1}{2} (y^k)^\top Q(x^k) y^k + q(x^k)^\top y^k$, (x^k, y^k) is the optimal solution of (BP)	Reformulation	If $\langle \nabla \theta(Y^k), Z^k \rangle \leq 0$, Y^k is the optimal solution of (DSIP), x^k is the optimal solution of (SIP)
Step 3b	If $h(x^k) > \frac{1}{2} (y^k)^\top Q(x^k) y^k + q(x^k)^\top y^k$, build R_{k+1} as R_k with the adjoined ineq. $h(x) \leq \frac{1}{2} (y^k)^\top Q(x) y^k + q(x)^\top y^k$	Reformulation	If $\langle \nabla \theta(Y^k), Z^k \rangle > 0$, set $B_{k+1} \leftarrow B_k \cup \{Z^k\}$.

Table 1: The k -th iteration of the CP (Algorithm 4.1), and of the FCFW algorithm

365 The different steps summarized in Table 1 can be explicated as follows:

- 366 • *Step 1:* At iteration k , set B_k represents, from a dual perspective, the set of CPs in the
 367 primal relaxation R_k . The dual problem of R_k is in fact a restriction of (DSIP) on $\text{cone}(B_k)$,
 368

369 which is a polyhedral subcone of \mathcal{K} , since the following holds:

$$\begin{aligned}
\max_{Y \in \text{cone}(B_k)} \theta(Y) &= \max_{Y \in \text{cone}(B_k)} \min_{x \in \mathcal{X}} (F(x) - \langle \mathcal{G}(x), Y \rangle) \\
&= \min_{x \in \mathcal{X}} \max_{Y \in \text{cone}(B_k)} (F(x) - \langle \mathcal{G}(x), Y \rangle) \\
&= \min_{x \in \mathcal{X}} \{F(x) \text{ s.t. } 0 \leq \langle \mathcal{G}(x), Z \rangle, \forall Z \in B_k\},
\end{aligned}$$

371 which we recognize being the master problem R_k . The absence of duality gap is, also in this
372 case, a direct application of Sion's Theorem. The new dual solution Y^k is obtained solving
373 this restriction of (DSIP) on $\text{cone}(B_k)$, and the primal solution $x^k = \arg \min_{x \in \mathcal{X}} \mathcal{L}(x, Y^k)$ gives

374 the gradient of the dual function in Y^k , i.e., $\nabla \theta(Y^k) = -\mathcal{G}(x^k)$.

375 • *Step 2:* Finding the *bilevel* constraint that is the most violated by x^k is equivalent to finding
376 the furthest point of \mathcal{P} in the direction $\nabla \theta(Y^k)$. Indeed, the following equality holds:

$$377 \quad (4.7) \quad \max_{Z \in \mathcal{P}} \langle \nabla \theta(Y^k), Z \rangle = -\min_{Z \in \mathcal{P}} \langle \mathcal{G}(x^k), Z \rangle$$

$$378 \quad (4.8) \quad = -\min_{y \in \mathcal{F}} \left\{ \frac{1}{2} y^\top Q(x^k) y + q(x^k)^\top y - h(x^k) \right\},$$

380 and any optimal solution Z^k in problem (4.7) has the form $Z^k = M(y^k)$, with y^k optimal
381 in problem (4.8).

382 • *Step 3a:* The CP feasibility test $\frac{1}{2}(y^k)^\top Q(x^k)y^k + q(x^k)^\top y^k \geq h(x^k)$, is equivalent to the
383 dual optimality condition $\langle \nabla \theta(Y^k), Z^k \rangle \leq 0$, according to the equality $\nabla \theta(Y^k) = -\mathcal{G}(x^k)$.

384 • *Step 3b:* Increasing the set of atoms $B_{k+1} \leftarrow B_k \cup \{Z^k\}$ is the dual point of view of adding
385 the corresponding CP (with y^k s.t. $Z^k = M(y^k)$) to R_k , which creates the relaxation R_{k+1} .

386 The following lemma states a property of the iterates Y^k .

387 LEMMA 4.3. *For any $k \in \mathbb{N}$, $\langle \nabla \theta(Y^k), Y^k \rangle = 0$.*

388 *Proof.* This property follows directly from the first order optimality condition at 1 of the dif-
389 ferentiable function $g : \begin{cases} \mathbb{R}_+ \rightarrow \mathbb{R} \\ t \mapsto \theta(tY^k) \end{cases}$. Indeed, $g'(1) = \langle \nabla \theta(Y^k), Y^k \rangle = 0$, because (i) 1 is optimal
390 for g since $Y^k \in \arg \max_{Y \in \text{cone}(B_k)} \theta(Y)$, (ii) 1 lies in the interior of the definition domain of g . \square

391 Based on the dual interpretation of the CP algorithm, we are now going to state a convergence rate
392 for this algorithm. We begin with two technical lemmas.

LEMMA 4.4. *It exists $L > 0$ s.t. function θ is L -smooth, i.e., for all $Y, Y' \in \mathcal{K}$,*

$$\|\nabla \theta(Y) - \nabla \theta(Y')\|_2 \leq L \|Y - Y'\|_2.$$

393 *Proof.* For the purpose of this proof, we introduce the linear operator \mathcal{Q}^* , defined as the adjoint
394 operator of the linear (by Assumption 3) operator $x \mapsto \mathcal{Q}(x)$. With this notation, we have that
395 $\langle \mathcal{Q}(x), Y \rangle = x^\top (\mathcal{Q}^* Y)$. We also denote by $\|\mathcal{Q}^*\|_{\text{op}}$ the operator norm of \mathcal{Q}^* . We notice that the
396 image of the bounded set \mathcal{X} by the subdifferential mapping $\partial h(\mathcal{X}) = \bigcup_{x \in \mathcal{X}} \partial h(x)$ is bounded according
397 to Theorem 6.2.2 in [14, Chapter VI]. Hence it exists $D \geq 0$ such that

$$398 \quad (4.9) \quad \forall x \in \mathcal{X}, \forall s \in \partial h(x), \quad \|s\|_2 \leq D.$$

399 Given $Y, Y' \in \mathcal{K}$, we are now going to prove that $\|\nabla\theta(Y) - \nabla\theta(Y')\|_2 \leq L\|Y - Y'\|_2$ for a constant
 400 L that is independent from Y and Y' . Being $i_{\mathcal{X}}(x)$ the indicator function of the set \mathcal{X} , we introduce
 401 the applications $w : x \mapsto \mathcal{L}(x, Y) + i_{\mathcal{X}}(x)$ and $w' : x \mapsto \mathcal{L}(x, Y') + i_{\mathcal{X}}(x)$. According to Assumptions
 402 6, as well as 1, 2, and 3 we remark that application w (resp. w') is μ -strongly convex because it
 403 is the sum of the μ -strongly convex function F and the convex function $x \mapsto -\langle \mathcal{G}(x), Y \rangle + i_{\mathcal{X}}(x)$
 404 (resp. $x \mapsto -\langle \mathcal{G}(x), Y' \rangle + i_{\mathcal{X}}(x)$). Being u (resp. u') the unique minimum of function w (resp. w'),
 405 the uniqueness following from the strong convexity, the optimality conditions of function w , and w'
 406 respectively read

$$407 \quad (4.10) \quad 0 \in \partial w(u),$$

$$408 \quad (4.11) \quad 0 \in \partial w'(u').$$

We remark that $w'(x) = F(x) + i_{\mathcal{X}}(x) + Y'_{n+1, n+1}h(x) - x^\top(\mathcal{Q}^*Y')$. The function $x \mapsto F(x) + i_{\mathcal{X}}(x)$
 is convex as a sum of convex functions; the function $x \mapsto Y'_{n+1, n+1}h(x)$ is convex since h is convex
 and $Y'_{n+1, n+1} \geq 0$ by definition of cone \mathcal{K} ; $x \mapsto -x^\top(\mathcal{Q}^*Y')$ is linear and thus convex. The
 intersection of the relative interiors of the domains of these convex functions is $\text{ri}(\mathcal{X})$. Since \mathcal{X} is a
 finite-dimensional convex set, $\text{ri}(\mathcal{X}) \neq \emptyset$ [29, Proposition 1.9]. Hence the subdifferential of the sum
 is the sum of the subdifferentials [24, Theorem 2.1]. In this respect, the subdifferential of function
 w' at u' reads

$$\partial w'(u') = \partial(F + i_{\mathcal{X}})(u') - \mathcal{Q}^*Y' + Y'_{n+1, n+1}\partial h(u').$$

410 Based on this decomposition, it follows from (4.11) that $\exists g_0 \in \partial(F + i_{\mathcal{X}})(u')$, $g_1 \in \partial h(u')$ such that

$$411 \quad (4.12) \quad g_0 - \mathcal{Q}^*Y' + Y'_{n+1, n+1}g_1 = 0.$$

412 Additionally, we have that

$$413 \quad (4.13) \quad g_0 - \mathcal{Q}^*Y + Y_{n+1, n+1}g_1 \in \partial w(u'),$$

414 since $w(x) = F(x) + i_{\mathcal{X}}(x) - x^\top(\mathcal{Q}^*Y) + Y_{n+1, n+1}h(x)$, and $g_0 \in \partial(F + i_{\mathcal{X}})(u')$, $g_1 \in \partial h(u')$.

415 Combining Eq. (4.12) with Eq. (4.13), we deduce:

$$416 \quad (4.14) \quad \mathcal{Q}^*(Y' - Y) + (Y_{n+1, n+1} - Y'_{n+1, n+1})g_1 \in \partial w(u').$$

417 Applying Theorem 6.1.2 in [14, Chapter VI], the μ -strong convexity of w gives that, for any
 418 $s_1 \in \partial w(u)$ and $s_2 \in \partial w(u')$, $\langle s_2 - s_1, u' - u \rangle \geq \mu\|u - u'\|_2^2$. Moreover, due to the Cauchy-Schwartz
 419 inequality, $\|s_1 - s_2\|_2\|u - u'\|_2 \geq \langle s_2 - s_1, u' - u \rangle$. Therefore, $\|s_2 - s_1\|_2 \geq \mu\|u - u'\|_2$ holds for any
 420 $s_1 \in \partial w(u)$ and $s_2 \in \partial w(u')$. Since $0 \in \partial w(u)$ according to (4.10), and $\mathcal{Q}^*(Y' - Y) + (Y_{n+1, n+1} -$
 421 $Y'_{n+1, n+1})g_1 \in \partial w(u')$ according to (4.14), we deduce that

$$422 \quad \|\mathcal{Q}^*(Y' - Y) + (Y_{n+1, n+1} - Y'_{n+1, n+1})g_1 - 0\|_2 \geq \mu\|u - u'\|_2.$$

423 According to the triangle inequality

$$424 \quad \|\mathcal{Q}^*(Y' - Y)\|_2 + |Y_{n+1, n+1} - Y'_{n+1, n+1}|\|g_1\|_2 \geq \mu\|u - u'\|_2,$$

425 and thus, since $\|Y - Y'\|_2 \geq |Y_{n+1, n+1} - Y'_{n+1, n+1}|$,

$$426 \quad \|\mathcal{Q}^*\|_{\text{op}}\|Y - Y'\|_2 + \|Y - Y'\|_2\|g_1\|_2 \geq \mu\|u - u'\|_2.$$

427 Defining $B = \|\mathcal{Q}^*\|_{\text{op}} + D$ and using the inequality $\|g_1\|_2 \leq D$, which holds according to (4.9), we
 428 know that

$$429 \quad B\|Y - Y'\|_2 \geq \mu\|u - u'\|_2.$$

430 According to Assumption 3, h is Lipschitz continuous and so are q and Q by the linearity Assumption
 431 2. Hence, it exists a constant $K > 0$ such that $x \mapsto \mathcal{G}(x)$ is K -Lipschitz continuous. We deduce
 432 that $K\|u - u'\|_2 \geq \|\mathcal{G}(u) - \mathcal{G}(u')\|_2$, and, consequently, $\|Y - Y'\|_2 \geq \frac{\mu}{BK}\|\mathcal{G}(u) - \mathcal{G}(u')\|_2$. We define
 433 the constant $L = \frac{BK}{\mu}$, which is clearly independent from Y, Y', u and u' . Since $\nabla\theta(Y) = -\mathcal{G}(u)$
 434 and $\nabla\theta(Y') = -\mathcal{G}(u')$, we deduce that

$$435 \quad L\|Y - Y'\|_2 \geq \|\nabla\theta(Y) - \nabla\theta(Y')\|_2,$$

436 which concludes the proof. \square

437 The following lemma is a consequence of the L -smoothness θ .

LEMMA 4.5. *Let L denote the smoothness constant associated with θ . For any $Y, Z \in \mathcal{K}$ and for any $\gamma \geq 0$,*

$$\theta(Y + \gamma Z) \geq \theta(Y) + \gamma\langle \nabla\theta(Y), Z \rangle - \frac{L\|Z\|^2}{2}\gamma^2.$$

438

439 *Proof.* For any $Y, Z \in \mathcal{K}$ and $\gamma > 0$, it holds by integration that

$$440 \quad (4.15) \quad \theta(Y + \gamma Z) - \theta(Y) = \int_{t=0}^{\gamma} \langle \nabla\theta(Y + tZ), Z \rangle dt = \gamma\langle \nabla\theta(Y), Z \rangle + \int_{t=0}^{\gamma} \langle \nabla\theta(Y + tZ) - \nabla\theta(Y), Z \rangle dt.$$

441 Since $\langle \nabla\theta(Y + tZ) - \nabla\theta(Y), Z \rangle \geq -|\langle \nabla\theta(Y + tZ) - \nabla\theta(Y), Z \rangle|$, using Cauchy-Schwartz inequality
 442 and L -smoothness of θ , we know that

$$443 \quad (4.16) \quad \langle \nabla\theta(Y + tZ) - \nabla\theta(Y), Z \rangle \geq -\|\nabla\theta(Y + tZ) - \nabla\theta(Y)\|_2 \|Z\|_2 \geq -tL\|Z\|_2^2.$$

444 Combining Eq. (4.15) with Eq. (4.16), we deduce that

$$445 \quad \theta(Y + \gamma Z) - \theta(Y) \geq \gamma\langle \nabla\theta(Y), Z \rangle - \int_{t=0}^{\gamma} tL\|Z\|_2^2 dt,$$

446 which yields finally that $\theta(Y + \gamma Z) - \theta(Y) \geq \gamma\langle \nabla\theta(Y), Z \rangle - \frac{L\|Z\|_2^2}{2}\gamma^2$. \square

447 We define the constant $T = \max_{Y \in \mathcal{P}} \|Z\|_2^2$, which is finite by compactness of \mathcal{F} , and thus of \mathcal{P} . According
 448 to Lemma 4.2, (DSIP) admits an optimal solution Y^* . We remark that the dual optimality gap at
 449 k -th iteration is $\delta_k = \theta(Y^*) - \theta(Y^k) \geq 0$, where δ_k is the objective error defined in Theorem 4.1.
 450 We define τ as the last element of the optimal dual solution Y^* , i.e. $\tau = Y_{n+1, n+1}^*$. This scalar
 451 plays a central role in the convergence rate analysis, conducted in the following theorem.

452 THEOREM 4.6. *Under Assumptions 1-7: if Algorithm 4.1 executes the iteration of index $k \in \mathbb{N}$,*
 453 *then*

$$454 \quad (4.17) \quad \delta_k \leq \frac{2LT\tau^2}{k+2}.$$

455 *Otherwise, it exists an index $j \leq k$ s.t. Y^j is optimal for (DSIP), and $x^j = \arg \min_{x \in \mathcal{X}} \mathcal{L}(x, Y^j)$ is*
 456 *optimal for (SIP).*

457 *Proof.* If the algorithm terminates at iteration $j \in \mathbb{N}$, this means that

$$458 \quad (4.18) \quad \max_{Z \in \mathcal{P}} \langle \nabla \theta(Y^j), Z \rangle \leq 0.$$

459 Defining $x^j = \arg \min_{x \in \mathcal{X}} \mathcal{L}(x, Y^j)$, we have that $\nabla \theta(Y^j) = -\mathcal{G}(x^j)$. Eq. (4.18) is thus equivalent to
 460 $\min_{Z \in \mathcal{P}} \langle \mathcal{G}(x^j), Z \rangle \geq 0$. This proves that x^j is feasible in (SIP). Moreover $\langle \mathcal{G}(x^j), Y^j \rangle = \langle \nabla \theta(Y^j), Y^j \rangle =$
 461 0, according to Lemma 4.3, and, therefore, $F(x^j) = \mathcal{L}(x^j, Y^j) = \theta(Y^j)$. Hence x^j and Y^j are feasible
 462 solutions in the primal (SIP) and the dual (DSIP) respectively, and have the same value. Therefore,
 463 x^j is optimal for (SIP), and Y^j is optimal for (DSIP).

464 We focus now on the case where Algorithm 4.1 does not terminates, and prove (4.17) by
 465 induction.

Base case: $k = 0$. Since θ is concave, we have that

$$\delta_0 = \theta(Y^*) - \theta(Y^0) = \theta(Y^*) - \theta(Y^0) \leq \langle \nabla \theta(Y^0), Y^* - Y^0 \rangle = \langle \nabla \theta(Y^0), Y^* \rangle,$$

the last equality coming from $Y^0 = 0$. We remark that $\langle \nabla \theta(Y^0), Y^* \rangle = \langle \nabla \theta(Y^0) - \nabla \theta(Y^*), Y^* \rangle$
 since $\langle \nabla \theta(Y^*), Y^* \rangle = 0$ by optimality of Y^* . Hence,

$$\delta_0 \leq \langle \nabla \theta(Y^0) - \nabla \theta(Y^*), Y^* \rangle \leq \|\nabla \theta(Y^0) - \nabla \theta(Y^*)\| \|Y^*\|,$$

where the last inequality is the Cauchy-Schwarz inequality. Using the L -Lipschitzness of $\nabla \theta$, we
 know that $\|\nabla \theta(Y^0) - \nabla \theta(Y^*)\| \leq L\|Y^0 - Y^*\| = L\|Y^*\|$. Finally, we deduce that, since $Y^* \in \tau\mathcal{P}$,

$$\delta_0 \leq L\|Y^*\|^2 \leq LT\tau^2.$$

466 *Induction.* We suppose that the algorithm runs $k + 1$ iterations, and that the property (4.17)
 467 is true for k . Using Lemma 4.5, we can compute a lower bound on the progress made during the
 468 iteration of index $k + 1$:

$$469 \quad \theta(Y^{k+1}) \geq \theta(Y^k + \gamma Z^k) \geq \theta(Y^k) + \gamma \langle \nabla \theta(Y^k), Z^k \rangle - \frac{L\|Z^k\|^2}{2} \gamma^2,$$

470 for any $\gamma \geq 0$. Multiplying by -1 , and adding $\theta(Y^*)$ to both left and right hand sides of the above
 471 inequality, and using $\|Z^k\|^2 \leq T$, we have that

$$472 \quad (4.19) \quad \delta_{k+1} \leq \delta_k - \gamma \langle \nabla \theta(Y^k), Z^k \rangle + \frac{LT}{2} \gamma^2,$$

473 for any $\gamma \geq 0$. We remark that the value T is independent from k . By concavity of θ , it also holds
 474 that $\delta_k = \theta(Y^*) - \theta(Y^k) = \theta(Y^*) - \theta(Y^k) \leq \langle \nabla \theta(Y^k), Y^* - Y^k \rangle$. We notice that $\langle \nabla \theta(Y^k), Y^k \rangle = 0$,
 475 according to Lemma 4.3. Thus, $\delta_k \leq \langle \nabla \theta(Y^k), Y^* \rangle$. As $Y_{n+1, n+1}^* = \tau$, we know that $Y^* \in \tau \text{conv}(\mathcal{P})$,
 476 and, therefore,

$$477 \quad (4.20) \quad \delta_k \leq \max_{Z \in \tau \text{conv}(\mathcal{P})} \langle \nabla \theta(Y^k), Z \rangle = \max_{Z \in \tau\mathcal{P}} \langle \nabla \theta(Y^k), Z \rangle = \tau \langle \nabla \theta(Y^k), Z^k \rangle,$$

478 the last equality following from the definition of Z^k . Combining Eq. (4.19) and (4.20), it holds that

$$479 \quad \delta_{k+1} \leq \delta_k - \gamma \tau^{-1} \delta_k + \frac{LT}{2} \gamma^2,$$

480 for every $\gamma \geq 0$. Factorizing and doing a change of variable $\eta = \gamma\tau^{-1}$, for any $\eta \geq 0$:

$$481 \quad (4.21) \quad \delta_{k+1} \leq (1 - \eta)\delta_k + \frac{LT\tau^2}{2}\eta^2.$$

482 We have derived a lower bound on optimality gap at iteration k . We apply then (4.21) with $\eta = \frac{2}{k+2}$:

$$483 \quad \delta_{k+1} \leq \left(1 - \frac{2}{k+2}\right)\delta_k + \frac{LT\tau^2}{2} \frac{4}{(k+2)^2} \leq \frac{k}{k+2} \frac{2LT\tau^2}{k+2} + \frac{LT\tau^2}{2} \frac{4}{(k+2)^2},$$

485 the second inequality coming from the application of (4.17) for k , which is true by induction
486 hypothesis. Finally, we deduce that

$$487 \quad \delta_{k+1} \leq \frac{2LT\tau^2}{k+2} \left(\frac{k}{k+2} + \frac{1}{k+2}\right) \leq \frac{2LT\tau^2}{k+2} \frac{k+1}{k+2} \leq \frac{2LT\tau^2}{k+2} \frac{k+2}{k+3} = \frac{2LT\tau^2}{k+3},$$

489 the third inequality coming from the observation that $\frac{k+1}{k+2} \leq \frac{k+2}{k+3}$. Hence, the property (4.17) is
490 true for $k+1$ as well. This concludes the proof by induction. \square

491 We remark that the convergence rate defined in (4.17) is directly related to the iteration index k ,
492 which is something different w.r.t. what is usually proved for existing CP algorithms solving SIP
493 problems [8, 17, 23], where the rate of convergence is not directly controlled by k .

494 **5. Applications.** In this section, we present two problems that can be modeled as (BP). For
495 each of these, we present both the bilevel formulation, and the corresponding single-level formulation
496 (BPR).

497 **5.1. Constrained quadratic regression.** We consider a quadratic statistical model with
498 Gaussian noise linking a vector $w \in \mathbb{R}^n$ of explanatory variables, i.e., the features vector, and an
499 output $z \in \mathbb{R}$ as follows:

$$500 \quad z = \frac{1}{2}w^\top \bar{Q}w + \bar{q}^\top w + \bar{c} + \epsilon,$$

501 where $\bar{Q} \in \mathbb{R}^{n \times n}$ s.t. $\bar{Q} = \bar{Q}^\top$, $\bar{q} \in \mathbb{R}^n$, $\bar{c} \in \mathbb{R}$ and $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Let us suppose that the
502 parameters of this model are unknown, but we are given a dataset $(w_i, z_i)_{1 \leq i \leq P} \in (\mathbb{R}^n \times \mathbb{R})^P$.
503 The problem of finding the maximum likelihood estimator for $\bar{Q} \in \mathbb{R}^{n \times n}$, $\bar{q} \in \mathbb{R}^n$, $\bar{c} \in \mathbb{R}$ just
504 consists in computing the triplet $(Q, q, c) \in \mathbb{R}^{n \times n} \times \mathbb{R}^n \times \mathbb{R}$ that minimizes the least-squares error
505 $\sum_{i=1}^P (z_i - \frac{1}{2}w_i^\top Qw_i - q^\top w_i - c)^2$. We consider that (i) the features vector belongs to a given polytope
506 $\mathcal{F} \subset \mathbb{R}^n$, (ii) the noiseless value $\frac{1}{2}y^\top \bar{Q}y + \bar{q}^\top y + \bar{c}$ is nonnegative for any $y \in \mathcal{F}$. Hence, this inverse
507 problem is a ‘‘constrained quadratic regression problem’’ that may be written as:

$$508 \quad (5.1) \quad \begin{cases} \min_{Q, q, c} & \sum_{i=1}^P (z_i - \frac{1}{2}w_i^\top Qw_i - q^\top w_i - c)^2 \\ \text{s.t.} & Q = Q^\top \\ & \frac{1}{2}y^\top Qy + q^\top y + c \geq 0 \quad \forall y \in \mathcal{F} \\ & Q \in \mathbb{R}^{n \times n}, q \in \mathbb{R}^n, c \in \mathbb{R}. \end{cases}$$

509 Formulation (5.1) is a SIP problem, having uncountably many constraints, which are parametrized
510 by $y \in \mathcal{F}$. We can reformulate this SIP problem as a bilevel problem just replacing the SIP constraint

511 $\frac{1}{2}y^\top Qy + q^\top y + c \geq 0 \forall y \in \mathcal{F}$ with the bilevel constraint $\min_{y \in \mathcal{F}} \{\frac{1}{2}y^\top Qy + q^\top y\} \geq -c$. This model
 512 fits in the general setting of formulation (BP), where the matrix Q is itself the upper-level variable
 513 of dimensions $n \times n$. As in Section 3, we assume that $\mathcal{F} = \{y \in \mathbb{R}^n : a_j^\top y \leq b_j, \forall j = 1, \dots, r\}$ is
 514 included in the centered ℓ_2 -ball with radius $\rho > 0$, and we use the notation $\mathcal{A}_j = \begin{pmatrix} 0_n & \frac{a_j}{2} \\ \frac{a_j^\top}{2} & 0 \end{pmatrix}$ for
 515 all $j \in \{1, \dots, r\}$. Then, the (BPR) formulation corresponding to (5.1) reads:

$$516 \quad (5.2) \quad \left\{ \begin{array}{l} \min_{Q, q, c, \lambda, \alpha, \beta} \sum_{i=1}^P (z_i - \frac{1}{2}w_i^\top Qw_i - q^\top w_i - c)^2 \\ \text{s.t. } Q = Q^\top \\ -\lambda^\top b - \alpha(1 + \rho^2) - \beta \geq -c \\ \frac{1}{2} \begin{pmatrix} Q + 2\alpha I_n & q \\ q^\top & 2(\beta + \alpha) \end{pmatrix} + \sum_{j=1}^r \lambda_j \mathcal{A}_j \succeq 0 \\ Q \in \mathbb{R}^{n \times n}, q \in \mathbb{R}^n, c \in \mathbb{R} \\ \lambda \in \mathbb{R}_+^r, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}. \end{array} \right.$$

517 Formulation (5.2) is feasible, because the all-zero solution satisfies every constraint. In general,
 518 (5.2) is a restriction of (5.1) since Q may not necessarily be PSD. In order to benchmark our
 519 approaches, we can solve the following relaxation of (5.1) — it is be a reformulation if Q is PSD —
 520 obtained by replacing the lower-level problem by its KKT conditions:

$$521 \quad (5.3) \quad \left\{ \begin{array}{l} \min_{Q, q, c, y, \gamma} \sum_{i=1}^P (z_i - \frac{1}{2}w_i^\top Qw_i - q^\top w_i - c)^2 \\ \text{s.t. } Q = Q^\top \\ \frac{1}{2}y^\top Qy + q^\top y \geq -c \\ Ay \leq b \\ Qy + q + A^\top \gamma = 0 \\ \gamma^\top (Ay - b) = 0 \\ Q \in \mathbb{R}^{n \times n}, q \in \mathbb{R}^n, c \in \mathbb{R}, y \in \mathbb{R}^n, \gamma \in \mathbb{R}_+^r, \end{array} \right.$$

522 where γ is the KKT multiplier vector associated to the lower-level constraints $Ay \leq b$. This relax-
 523 ation/reformulation of problem (5.1) is a non-convex polynomial optimization problem involving
 524 multivariate polynomials of degree up to three.

525 **5.2. Zero-sum game with cubic payoff.** In this section, we are interested in solving a two-
 526 player zero-sum game that is related to an undirected graph $\mathcal{G} = (V, E)$. We assume that player
 527 1 benefits from a strategical advantage on player 2, which will be explained more precisely later.
 528 We let n denote the cardinality of V . Each player positions a resource on each node $i \in V$. After
 529 normalization, we can consider that the action set of both players is $\Delta_n = \{x \in \mathbb{R}_+^n : \sum_{i=1}^n x_i = 1\}$.

530 A two-player zero-sum game is a two-player game s.t., for every strategy $x \in \Delta_n$ of player 1, and for
 531 every strategy $y \in \Delta_n$ of player 2, the payoffs of the two players sum to zero. If we define $P_i(x, y)$
 532 the payoff of player i related to the strategy pair (x, y) , we thus have that $P_1(x, y) = -P_2(x, y)$.
 533 Since the payoffs sum to zero, we can write the zero-sum game by specifying only one *game payoff*.
 534 Player 1 wishes to minimize it, and player 2 wishes to maximize it. The game payoff $P(x, y)$ related
 535 to the pair of strategies $(x, y) \in \Delta_n \times \Delta_n$ is the sum of:

- 536 • the opposite of a term describing the “proximity” between x and y in the graph, $x^\top My$,
- 537 where $M \in \mathbb{R}^{n \times n}$ is the matrix defined as $M_{ij} = 1$ if $i = j$ or $\{i, j\} \in E$, and $M_{ij} = 0$
- 538 otherwise,

539 • the quadratic costs that player 1 has to pay to deploy his resources on the graph: $c_1(x) =$
540 $\frac{1}{2}x^\top Q_1 x + q_1^\top x,$
541 • the opposite of the quadratic costs that player 2 has to pay to deploy her resources on the
542 graph, and that is influenced by player 1 strategy: $c_2(x, y) = \frac{1}{2}y^\top Q_2(x)y + q_2^\top y.$ In this
543 sense, player 1 has a strategic advantage over player 2.
544 Hence, this zero-sum game can then be written as $\min_{x \in \Delta_n} \max_{y \in \Delta_n} -x^\top M y + c_1(x) - c_2(x, y).$ Loosely
545 speaking, player 1 trades off his costs for placing his resource where player 2's one is (i.e., maximizing
546 the proximity) and for augmenting player 2's costs. In the meantime, player 2 tries to *avoid* player
547 1, while minimizing her own costs. From player 1's perspective, this problem can be cast as the
548 following bilevel formulation:

$$549 \quad (5.4) \quad \left\{ \begin{array}{l} \min_{x,v} \quad \frac{1}{2}x^\top Q_1 x + q_1^\top x + v \\ \text{s.t.} \quad -v \leq \min_{y \in \Delta_n} \frac{1}{2}y^\top Q_2(x)y + (q_2 + M^\top x)^\top y \\ \\ x \in \Delta_n, v \in \mathbb{R}. \end{array} \right.$$

550

551 This latter formulation clearly fits in the general setting of formulation (BP). Hence, we apply the
552 methodology of Section 3 with $r = n + 2$, and

- 553 • $a_1 = \mathbf{1}$ and $b_1 = 1$,
554 • $a_2 = -\mathbf{1}$ and $b_2 = 0$,
555 • $\forall j \in \{1, \dots, n\} \quad a_{j+2} = -e_j$ and $b_j = 0$,
556 • $\rho = 1$,

557 where e_j is the j -th vector of the standard basis in \mathbb{R}^n and $\mathbf{1}$ the all-ones n -dimensional vector. The
558 dual variable is $\lambda \in \mathbb{R}_+^{n+2}$. In this application, the single-level formulation (BPR) reads

$$559 \quad (5.5) \quad \left\{ \begin{array}{l} \min_{x,v,\lambda,\alpha,\beta} \quad v + \frac{1}{2}x^\top Q_1 x + q_1^\top x \\ \text{s.t.} \quad -v \leq -\lambda_1 - 2\alpha - \beta \\ \\ \frac{1}{2} \begin{pmatrix} Q_2(x) + 2\alpha I_n & W(x, \lambda) \\ W(x, \lambda)^\top & 2\beta + 2\alpha \end{pmatrix} \succeq 0 \\ \\ x \in \Delta_n, v \in \mathbb{R} \\ \lambda \in \mathbb{R}_+^{n+2}, \alpha \in \mathbb{R}_+, \beta \in \mathbb{R}, \end{array} \right.$$

560

561 where $W(x, \lambda) = q_2 + M^\top x - \sum_{j=1}^n \lambda_{j+2} e_j + (\lambda_1 - \lambda_2) \mathbf{1}$. If $Q_2(x) \succeq 0$ is PSD for any $x \in \Delta_n$,
562 formulation (5.5) is a reformulation of (5.4). Otherwise, it is just a restriction of (5.4). In any case,
563 such formulation is feasible, because for given vectors $x \in \Delta_n$, $\lambda \in \mathbb{R}_+^{n+2}$ and scalar $\beta \in \mathbb{R}$, taking
564 arbitrary large scalars α and v , the two constraints are satisfied.

565 As for the first application, we benchmark our two approaches with the KKT-based relax-
566 ation/reformulation (depending on the convexity of the lower-level problem). Given the KKT
567 multipliers γ_1 and γ_2 associated respectively to the lower-level constraint $\sum_{i=1}^n y_i = 1$, and the non-
568 negativity constraint $y \geq 0$, the single-level formulation obtained by replacing the lower level of

569 (5.4) by its KKT conditions, is

$$\begin{aligned}
 & \min_{x,v,y,\gamma_1,\gamma_2} \quad v + \frac{1}{2}x^\top Q_1 x + q_1^\top x \\
 & \text{s.t.} \quad -v \leq \frac{1}{2}y^\top Q_2(x)y + (q_2 + M^\top x)^\top y \\
 & \quad \quad Q_2(x)y + q_2 + M^\top x + \gamma_1 \mathbf{1} - I_n \gamma_2 = 0 \\
 & \quad \quad -\gamma_2^\top (I_n y) = 0 \\
 & \quad \quad x \in \Delta_n, y \in \Delta_n, v \in \mathbb{R}, \gamma_1 \in \mathbb{R}, \gamma_2 \in \mathbb{R}_+^n.
 \end{aligned}
 \tag{5.6}$$

572 The KKT multiplier γ_1 is associated to an equality constraint, hence it can be either nonnegative
 573 or negative, and we have no complementarity constraint involving it in formulation (5.6). This
 574 relaxation/reformulation of problem (5.4), as well as (5.6), is a non-convex polynomial optimization
 575 problem involving multivariate polynomials of degree up to three.

576 **6. Numerical results.** In this section we present the numerical results obtained by testing
 577 several instances of the two applications presented in Section 5, available online at the public
 578 repository <https://github.com/aoustry/Bilevel-programs-with-QP-as-LL>.

579 For the constrained quadratic regression (Section 5.1), we solved twenty randomly generated
 580 instances. Each of these instances was generated by choosing the statistical parameters $\bar{Q}, \bar{q}, \bar{c}$ at
 581 random, drawing $P = 4000$ random features vectors $w_i \in \mathbb{R}^n$, and then computing the associated
 582 outputs $z_i \in \mathbb{R}$ with a centered Gaussian noise. Ten instances — named *PSD_inst#* in Table 2 —
 583 were produced with \bar{Q} PSD and ten instances — named *notPSD_inst#* in Table 2 — with an
 584 indefinite \bar{Q} .

585 For the zero-sum game with cubic payoff application (Section 5.2), we tested twenty-two in-
 586 stances where the matrix M is taken from the DIMACS graph coloring challenge¹. We randomly
 587 generated Q_1 in a way such that it is PSD, as well as the coefficients of the linear mapping $x \mapsto Q_2(x)$
 588 such that $Q_2(x)$ is PSD for all feasible x in the instances named *#_PSD* in Table 3. Regarding the
 589 instances named *#_notPSD* in Table 3, no particular precaution was taken to enforce that $Q_2(x)$
 590 is PSD. Hence, the sign of the eigenvalues of $Q_2(x)$ depends on x . The code that generated all the
 591 instances is available online.

592 We implemented the single-level formulations based on the *dual approach* using the Python
 593 programming language [30] and solve them with the conic optimization solver Mosek [2]. The
 594 bilevel formulations were solved using the CP algorithm (Algorithm 4.1 presented in Section 4) and
 595 implemented using the AMPL modeling language [11]. Both the master problem R_k and the lower
 596 level problem P_{x^k} were solved using the global optimization solver Gurobi [12]. The tolerance for
 597 the feasibility error $\epsilon_k = (h(x^k) - \text{val}(P_{x^k}))^+$ is set to 10^{-6} . With AMPL, we also implemented
 598 the traditional relaxation/reformulation approach based on the KKT conditions of the lower-level
 599 problem. We solved the KKT-based formulations using the global optimization solver Couenne
 600 [5], chosen after some preliminary computational experiments. These formulations are particularly
 601 hard to solve for Couenne, mainly because of the complementarity constraints. Indeed, for all the
 602 tested instances, Couenne does not terminate within the time limit, and we just display, in italic
 603 font, the LB given by the optimal value of the best relaxation of the KKT formulation found by
 604 Couenne within the time limit. All the solvers were run with their default settings. The tests were
 605 performed on a computer with 24 2.53GHz Intel(R) Xeon(R) CPUs and with 49.4 GB of RAM. For
 606 all the approaches we set a time limit (t.l.) of 18000 seconds (5 hours).

¹ <https://mat.tepper.cmu.edu/COLOR/instances.html>

607 The results for Application 1 and Application 2 are reported in Table 2 and Table 3 respectively.
 608 The headings are the following: “ n ” is the dimension of the lower-level variable y (or, equivalently,
 609 for Application 1 of the matrix Q , for Application 2 of the upper-level variable x); for the single-level
 610 formulation approach “obj” is the optimal value found by Mosek (i.e., either the bilevel optimal
 611 value, or an upper bound of it); for the KKT approach, “ LB ”, reported in italics, is the best LB
 612 of the KKT formulation value found by the solver Couenne within the time limit, which is a lower
 613 bound for the bilevel optimal value too; for the CP approach “obj/ LB - UB ” is, respectively, either
 614 the optimal value of the bilevel formulation, or a pair of values corresponding to: the best lower
 615 bound (LB) and the best feasible solution, i.e., upper bound (UB), found by the algorithm within
 616 the time limit; “time(s)” is the computing time in seconds; “it” is the number of CP iterations,
 617 i.e., the number of times R_k and (P_{x^k}) are solved; “% time (P_{x^k})” is the percentage of the total
 618 computing time, i.e. time(s), used to solve (P_{x^k}) . In Table 2, the “Avg LSE”, which is the average
 619 least-squares error of the regression, is reported as well. In Table 2 and Table 3, the best objective
 620 values and minimum required times are reported in bold for each instance.

Instances		Single-level formulation			KKT approach	CP approach				
Name	n	obj	Avg LSE	time(s)	LB	obj/ LB - UB	Avg LSE	time(s)	it	% time (P_{x^k})
PSD_inst1	5	358.64	0.08966	0.19	<i>355.78</i>	358.64	0.08966	1.21	6	3.9
PSD_inst2	5	365.60	0.09140	0.26	<i>363.85</i>	365.60	0.09140	0.63	3	4.1
PSD_inst3	5	363.43	0.09086	0.07	<i>359.16</i>	363.43	0.09086	2.62	8	18.0
PSD_inst4	5	353.90	0.08847	0.07	<i>353.19</i>	353.90	0.08847	1.93	5	32.2
PSD_inst5	10	391.21	0.09780	0.37	<i>359.48</i>	391.21	0.09780	23.5	17	0.7
PSD_inst6	10	397.59	0.09940	0.41	<i>353.55</i>	397.59	0.09940	24.2	17	0.7
PSD_inst7	13	440.84	0.11021	0.36	<i>358.19</i>	440.84	0.11021	64.3	19	0.3
PSD_inst8	13	382.22	0.09555	0.34	<i>345.52</i>	<i>381.81 - 383.34</i>	0.09545	t.l.	5	99.9
PSD_inst9	15	572.77	0.14319	0.92	<i>351.95</i>	<i>557.71 - 1362.6</i>	0.13943	t.l.	4	100.0
PSD_inst10	15	528.93	0.13223	1.37	<i>346.43</i>	<i>526.22 - 544.90</i>	0.13156	t.l.	8	100.0
notPSD_inst1	5	493.19	0.12330	0.14	<i>345.12</i>	358.47	0.08962	0.38	2	5.8
notPSD_inst2	5	425.14	0.10628	0.15	<i>370.89</i>	378.28	0.09457	0.39	2	5.7
notPSD_inst3	5	345.81	0.08645	0.06	<i>345.81</i>	345.81	0.08645	0.33	1	4.0
notPSD_inst4	5	353.25	0.08831	0.07	<i>353.25</i>	353.25	0.08831	0.19	1	3.6
notPSD_inst5	10	743.81	0.18595	0.55	<i>360.42</i>	503.88	0.12597	28.3	19	12.9
notPSD_inst6	10	637.62	0.15940	0.28	<i>357.48</i>	482.96	0.12074	412	41	86.6
notPSD_inst7	13	903.44	0.22586	0.35	<i>351.31</i>	647.08	0.16177	657	57	69.7
notPSD_inst8	13	932.21	0.23305	0.30	<i>358.28</i>	588.19	0.14705	3825	77	92.9
notPSD_inst9	15	1592.60	0.39815	0.99	<i>345.44</i>	1126.44	0.28161	15002	99	95.5
notPSD_inst10	15	897.89	0.22447	0.83	<i>350.60</i>	580.60	0.14515	2537	56	87.0

Table 2: Numerical results of the first application

621 As expected, the *dual approach* leads to a single-level formulation which is a restriction for
 622 most of the BP problems with a non-convex lower level, but for the instances *notPSD_inst3* and
 623 *notPSD_inst4* of Table 2, where the bilevel global optimal solution is attained using both the two
 624 approaches, despite the matrix Q is indefinite. It is clear that, in terms of computational time, the
 625 *dual approach* is more efficient than the CP approach, not only when Mosek deals with a restriction
 626 of the original BP but also when a reformulation is solved. This is the main reason why the *dual*
 627 *approach* is promising, even if a restriction of the original BP program is solved. In fact, it let
 628 us compute either the bilevel optimal solution or an upper bound of such solution within a small
 629 CPU time. As concerns the computation of lower bounds, we see that the CP algorithm provides
 630 much tighter lower-bounds than the best lower bound of the KKT relaxation computed by Couenne
 631 within the time limit. Indeed, this formulation is particularly hard to solve mainly because of the
 632 complementarity constraints. To understand the causes of the long computational time required

Instances		Single-level formulation		KKT approach	CP approach			
Name	n	obj	time(s)	LB	obj/LB-UB	time(s)	it	% time (P_{x^k})
jean_PSD	80	-0.0760	18.4	-4.5808	-0.0760	4.68	186	38.5
myciel4_PSD	23	-0.3643	0.06	-1.9429	-0.3643	14.3	422	26.8
myciel5_PSD	47	-0.3164	1.45	-4.0081	-0.3164	85.4	752	9.2
myciel6_PSD	95	-0.2841	41.4	-9.1222	-0.2841	2781	2323	1.0
myciel7_PSD	191	-0.2608	4359	-14.9495	-0.2608 - -0.2608	t.l.	3565	0.4
queen5.5_PSD	25	-0.5536	0.10	-5.6076	-0.5536	4.16	161	44.3
queen6.6_PSD	36	-0.4619	0.38	-5.6353	-0.4619	34.4	512	18.3
queen7.7_PSD	49	-0.4054	1.47	-7.8210	-0.4054	155	969	7.8
queen8.8_PSD	64	-0.3614	4.22	-12.7220	-0.3614	742	1651	3.1
queen8.12_PSD	96	-0.3000	34.8	-16.0606	-0.3000 - -0.3000	t.l.	4082	0.4
queen9.9_PSD	81	-0.3247	14.4	-14.5807	-0.3247	3544	2578	0.8
jean_notPSD	80	3.2708	17.4	-8.5541	2.3979	37.6	6	99.7
myciel4_notPSD	23	0.8668	0.07	-2.5166	0.5198	466	44	99.9
myciel5_notPSD	47	1.9571	1.27	-7.4343	1.2779	315	32	99.8
myciel6_notPSD	95	3.9171	39.2	-13.9108	2.9378	2735	38	100
myciel7_notPSD	191	7.8030	3419	$-\infty$	6.2486 - 6.2486	t.l.	19	100
queen5.5_notPSD	25	0.8112	0.08	-4.7699	0.3800	326	53	99.8
queen6.6_notPSD	36	1.3876	0.37	-9.7370	0.8511	15872	71	100.0
queen7.7_notPSD	49	1.9740	1.56	-12.4690	1.3510	852	42	99.9
queen8.8_notPSD	64	2.6032	5.79	-15.0751	1.8123	10410	42	100
queen8.12_notPSD	96	3.8131	41.0	-31.4660	2.8102	7035	30	100
queen9.9_notPSD	81	3.2449	17.3	-17.4348	2.2975 - 2.2996	t.l.	23	100

Table 3: Numerical results of the second application

633 by the CP algorithm, we can look at the last column of Table 2 and 3. For the first application,
634 the time required to perform step 4 of the CP algorithm (i.e. to solve P_{x^k}) is longer than the time
635 required to perform step 3 (i.e. to solve R_k) only for the bigger instances ($n \geq 13$ for instances with
636 a convex lower level and $n \geq 10$ for instances with a non-convex lower level). In fact, when n grows,
637 more time is needed to solve a possibly non-convex QP problem having Q and q as coefficients,
638 rather than a convex QP having Q and q as variables. When n is small, it is different: even if
639 the inner problem is quadratic non-convex, it has a small size so it is not harder to solve than the
640 master problem. For the second application, the time required to solve the lower-level problem is
641 longer than the time required to solve the outer relaxation only for the instances having a non-
642 convex lower level, i.e., the second half of the Table 3 rows. In fact, problem R_k has a convex
643 quadratic objective function, since the matrix Q_1 is always PSD, while the inner problem has a
644 convex quadratic objective function only when the matrix $Q_2(x^k)$ is PSD. When $Q_2(x^k)$ is not
645 PSD, problem P_{x^k} is possibly non-convex and it becomes harder to solve than the master problem.

646 Figures 1 and 2 are aggregated plots showing, for all the tested instances, the trend of the
647 feasibility error ϵ_k over the iterations of the CP algorithm indexed by k . As already said, we set
648 a tolerance of 10^{-6} : for most of the instances, the algorithm stops when ϵ_k reaches or is less than
649 such value. For the instances where the algorithm reaches the time limit, the curve ends at a value
650 of ϵ_k greater than 10^{-6} . For all the instances, anyhow, we can see that the sequence of ϵ_k converges
651 towards 0, as proved in Theorem 4.1.

652 **7. Conclusion.** We focus on a class of bilevel programs having a possibly non-convex qua-
653 dratic programming problem at the lower level. These bilevel programs are, in fact, linear semi-
654 infinite programming problems with an infinite number of quadratically parameterized constraints.
655 From the point of view of Robust Optimization, it is about handling constraints with quadratic
656 perturbations and a polytopic uncertainty set. We propose two independent approaches to deal

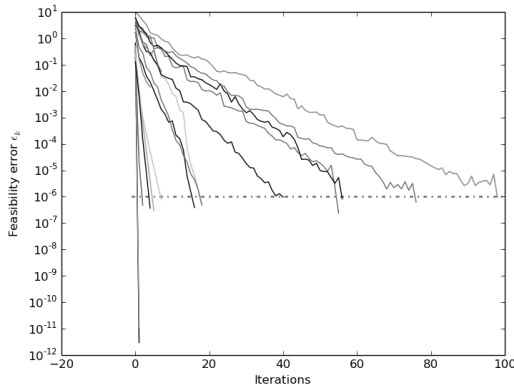


Fig. 1: Constrained quadratic regression

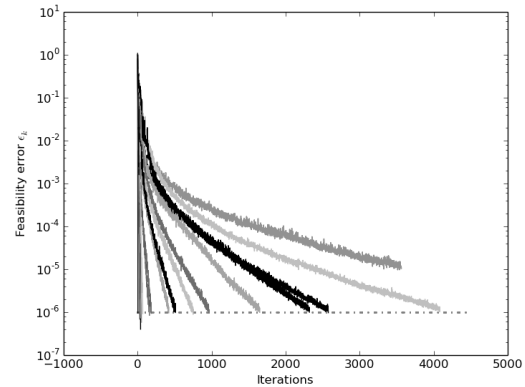


Fig. 2: Zero-sum game with cubic payoff

657 with such bilevel problems. First, a convex single-level formulation obtained via the *dual approach*
 658 provides a feasible solution, which is optimal in the case where the quadratic lower-level problem
 659 is convex. Second, a cutting plane algorithm enables one to solve directly the bilevel formulation
 660 with a guaranteed convergence rate, at the price of solving possibly non-convex quadratic inner
 661 problems. At each iteration, such algorithm provides a lower bound on the value of the bilevel
 662 program, which allows one to bound the optimality gap of the feasible solution obtained with the
 663 dual approach. Our computational experiments on small and medium-scale instances show the su-
 664 periority, in terms of solution time, of the dual approach for the instances with a convex lower-level
 665 problem. As concerns the cases with a non-convex lower-level problem, the two approaches are
 666 complementary: the dual approach is faster but provides “only” a feasible solution, the cutting
 667 plane approach is slower, but solves the bilevel problem to optimality with good accuracy. A pos-
 668 sible extension of our work could be implementing a cutting plane algorithm with the lower-level
 669 problem solved with an “on-demand” accuracy at each iteration. Regarding the dual approach, the
 670 sparse structure of the lower-level problem would be worth exploiting with the celebrated cliques
 671 decomposition technique. These possibilities will be addressed in future works.

672

REFERENCES

- 673 [1] S. ABBOTT, *Understanding Analysis*, Undergraduate Texts in Mathematics, Springer New York, 2016, <https://doi.org/10.1007/978-1-4939-2712-8>.
 674
 675 [2] M. APS, *The MOSEK python optimizer API manual. Version 9.2.36*, 2021, <https://docs.mosek.com/9.2/pythonapi/index.html>.
 676
 677 [3] J. BARD, *An algorithm for solving the general bilevel programming problem*, Mathematics of Operations Re-
 678 search, 8 (1983), pp. 260–272, <https://doi.org/10.1287/moor.8.2.260>.
 679 [4] J. BARD AND J. MOORE, *A branch and bound algorithm for the bilevel programming problem*, Siam Journal on
 680 Scientific and Statistical Computing, 11 (1990), <https://doi.org/10.1137/0911017>.
 681 [5] P. BELOTTI, J. LEE, L. LIBERTI, F. MARGOT, AND A. WÄCHTER, *Branching and bounds tightening techniques*
 682 *for non-convex MINLP*, Optimization Methods and Software, 24 (2009), pp. 597–634, <https://doi.org/10.1080/10556780903087124>.
 683
 684 [6] A. BEN-TAL, L. EL GHAOU, AND A. NEMIROVSKI, *Robust optimization*, Princeton university press, 2009.
 685 [7] A. BEN-TAL AND A. NEMIROVSKI, *Lectures on Modern Convex Optimization*, Society for Industrial and Applied
 686 Mathematics, 2001, <https://doi.org/10.1137/1.9780898718829>.

- 687 [8] B. BETRÓ, *An accelerated central cutting plane algorithm for linear semi-infinite programming*, Mathematical
688 Programming, 101 (2004), pp. 479–495, <https://doi.org/10.1007/s10107-003-0492-5>.
- 689 [9] I. D. COOPE AND G. A. WATSON, *A projected lagrangian algorithm for semi-infinite programming*, Mathematical
690 Programming, 32 (1985), pp. 337–356, <https://doi.org/10.1007/BF01582053>.
- 691 [10] S. FANG, C. LIN, AND S. WU, *Solving quadratic semi-infinite programming problems by using relaxed cutting-*
692 *plane scheme*, Journal of Computational and Applied Mathematics, 129 (2001), pp. 89–104, [https://doi.org/10.1016/S0377-0427\(00\)00544-6](https://doi.org/10.1016/S0377-0427(00)00544-6).
- 693 [11] R. FOURER, D. M. GAY, AND B. W. KERNIGHAN, *AMPL: A Modeling Language for Mathematical Programming*,
694 Cengage Learning, Boston, MA, 2002.
- 695 [12] L. GUROBI OPTIMIZATION, *Gurobi optimizer reference manual*, 2021, <http://www.gurobi.com>.
- 696 [13] R. HETTICH, *An implementation of a discretization method for semi-infinite programming*, Mathematical Pro-
697 gramming, 34 (1986), pp. 354–361, <https://doi.org/10.1007/BF01582235>.
- 698 [14] J. HIRIART-URRUTY AND C. LEMARÉCHAL, *Convex analysis and minimization algorithms I: Fundamentals*,
699 vol. 305, Springer-Verlag Berlin Heidelberg, 2013, <https://doi.org/10.1007/978-3-662-02796-7>.
- 700 [15] J. KELLEY, JR., *The cutting-plane method for solving convex programs*, Journal of the society for Industrial
701 and Applied Mathematics, 8 (1960), pp. 703–712, <https://www.jstor.org/stable/2099058>.
- 702 [16] M. KOJIMA AND L. TUNÇEL, *Cones of matrices and successive convex relaxations of nonconvex sets*, SIAM
703 Journal on Optimization, 10 (2000), pp. 750–778, <https://doi.org/10.1137/S1052623498336450>.
- 704 [17] K. O. KORTANEK AND H. NO, *A central cutting plane algorithm for convex semi-infinite programming problems*,
705 SIAM Journal on optimization, 3 (1993), pp. 901–918, <https://doi.org/10.1137/0803047>.
- 706 [18] L. LIBERTI, S. CAFIERI, AND F. TARISSAN, *Reformulations in mathematical programming: A computational*
707 *approach*, in Foundations of Computational Intelligence Volume 3: Global Optimization, A. Abraham
708 et al., eds., Springer, Berlin, Heidelberg, 2009, pp. 153–234, https://doi.org/10.1007/978-3-642-01085-9_7.
- 709 [19] G. LIN, M. XU, AND J. YE, *On solving simple bilevel programs with a nonconvex lower level program*, Mathe-
710 matical Programming, 144 (2014), pp. 277–305, <https://doi.org/10.1007/s10107-013-0633-4>.
- 711 [20] F. LOCATELLO, M. TSCHANNEN, G. RÄTSCH, AND M. JAGGI, *Greedy algorithms for cone constrained optimiza-*
712 *tion with convergence guarantees*, arXiv preprint, (2017), <http://arxiv.org/abs/1705.11041>.
- 713 [21] M. MERKERT, G. ORLINSKAYA, AND D. WENINGER, *An exact projection-based algorithm for bilevel mixed-*
714 *integer problems with nonlinearities*, Optimization Online, (2020), http://www.optimization-online.org/DB_FILE/2020/12/8153.pdf.
- 715 [22] A. MITSOS, P. LEMONIDIS, AND P. BARTON, *Global solution of bilevel programs with a nonconvex inner program*,
716 J. Global Optimization, 42 (2008), pp. 475–513, <https://doi.org/10.1007/s10898-007-9260-z>.
- 717 [23] R. REEMTSEN AND S. GÖRNER, *Numerical methods for semi-infinite programming: A survey*, in Semi-Infinite
718 Programming, R. Reemtsen and J. Rückmann, eds., Springer, Boston, 1998, pp. 195–275, https://doi.org/10.1007/978-1-4757-2868-2_7.
- 719 [24] G. ROMANO, *New results in subdifferential calculus with applications to convex optimization*, Applied Mathe-
720 matics and Optimization, 32 (1995), pp. 213–234, <https://doi.org/10.1007/BF01187900>.
- 721 [25] G. SAGNOL, *On the semidefinite representation of real functions in symmetric matrices*, Linear Algebra
722 and its Applications, 439 (2013), pp. 2829–2843, <https://doi.org/10.1016/j.laa.2013.08.021>.
- 723 [26] G. STILL, *Discretization in semi-infinite programming: the rate of convergence*, Mathematical programming,
724 91 (2001), pp. 53–69, <https://doi.org/10.1007/s101070100239>.
- 725 [27] A. TAKEDA AND M. KOJIMA, *Successive convex relaxation approach to bilevel quadratic optimization problems*,
726 in Complementarity: Applications, Algorithms and Extensions., M. Ferris et al., eds., Springer-Boston,
727 2000, pp. 317–340, https://doi.org/10.1007/978-1-4757-3279-5_15.
- 728 [28] Y. TANAKA, M. FUKUSHIMA, AND T. IBARAKI, *A globally convergent SQP method for semi-infinite nonlinear*
729 *optimization*, Journal of Computational and Applied Mathematics, 23 (1988), pp. 141–153, [https://doi.org/10.1016/0377-0427\(88\)90276-2](https://doi.org/10.1016/0377-0427(88)90276-2).
- 730 [29] H. TUY, *Convex Analysis and Global Optimization*, vol. 22, Springer, Boston, MA, 2 ed., 1998, <https://doi.org/10.1007/978-3-319-31484-6>.
- 731 [30] G. VAN ROSSUM AND F. L. DRAKE, JR., *Python tutorial*, Centrum voor Wiskunde en Informatica Amsterdam,
732 The Netherlands, 1995.
- 733 [31] L. VANDENBERGHE AND S. BOYD, *Semidefinite programming*, SIAM review, 38 (1996), pp. 49–95, <https://doi.org/10.1137/1038003>.
- 734 [32] G. WOEINGER, *The trouble with the second quantifier*, 4OR, 19 (2021), pp. 157–181, <https://doi.org/10.1007/s10288-021-00477-y>.
- 735 [33] S. YAAKOB AND J. WATADA, *Solving bilevel quadratic programming problems and its application*, in Proceedings
736 of the 15th International Conference on Knowledge-Based and Intelligent Information and Engineering
737 Systems - Volume Part III, Springer-Verlag, 2011, pp. 187–196, <https://dl.acm.org/doi/10.5555/2041420>.

- 745 2041443.
746 [34] D. YUE, J. GAO, AND F. YOU, *A projection-based reformulation and decomposition algorithm for global opti-*
747 *mization of mixed integer bilevel linear programs*, *Journal of Global Optimization*, 73 (2019), pp. 27–57,
748 <https://doi.org/10.1007/s10898-018-0679-1>.