Submitted to *INFORMS Journal on Computing*
manuscript (Please, provide the mansucript number!)

Complexity Results and Effective Algorithms for Worst-case Linear Optimization under Uncertainties

Hezhi Luo
Department of Mathematics, College of Science, Zhejiang Sci-Tech University, Hangzhou, Zhejiang 310032, China. hzluo@zjut.edu.cn, hzluo@zstu.edu.cn

Xiaodong Ding
Department of Applied Mathematics, College of Science, Zhejiang University of Technology, Hangzhou, Zhejiang 310032, China. dxdopt@zjut.edu.cn

Jiming Peng
Department of Industrial Engineering, University of Houston, Houston, TX, 77204. jopeng@uh.edu

Rujun Jiang
School of Data Science, Fudan University, Shanghai, China. rjjiang@fudan.edu.cn

Duan Li
Corresponding author. School of Data Science, City University of Hong Kong, Hong Kong. dli226@cityu.edu.hk

We consider the so-called worst-case linear optimization (WCLO) with uncertainties in the right-hand-side of the constraints. Such a problem often arises in applications such as in systemic risk estimation in finance and stochastic optimization. In this paper, we first show that the WCLO problem with the uncertainty set corresponding to the ℓ_p-norm ((WCLO$_p$)) is NP-hard for $p \in (1, \infty)$. Second, we combine several simple optimization techniques such as the successive convex optimization method, quadratic convex relaxation, initialization and branch-and-bound (B&B), to develop an algorithm for (WCLO$_2$) that can find a globally optimal solution to (WCLO$_2$) within a pre-specified ϵ-tolerance. We establish the global convergence of the algorithm and estimate its complexity. We also develop a finite B&B algorithm for (WCLO$_\infty$) to identify a global optimal solution to the underlying problem, and establish the finite convergence of the algorithm. Numerical experiments are reported to illustrate the effectiveness of our proposed algorithms in finding globally optimal solutions to medium and large-scale WCLO instances.

Key words: Worst-case linear optimization; successive convex optimization; convex relaxation; branch-and-bound; computational complexity.
History:

1. Introduction

Linear optimization (LO) with uncertainty arises from a broad range of applications (cf. Birge and Louveaux (1997), Bertsimas et al. (2011)). Two popular approaches for uncertain LO are stochastic programming (cf. Birge and Louveaux (1997)) and robust optimization (cf. Ben-Tal et al. (2009), Bertsimas et al. (2011)), where decisions have to be made before the realization of the uncertain data. In practice, there exist cases where some decisions are of a "wait-and-see" type that can be made after partial realization of the uncertain data. One such example is the so-called adjustable robust optimization (cf. Ben-Tal et al. (2004)). There are also cases where it is critical to identify the worst-case scenario of the underlying uncertain LO problem. One such example is the assessment of systemic risk in a financial network where only limited and incomplete information regarding the network is available (cf. Eisenberg and Noe (2001), Peng and Tao (2015)).

In this paper, we consider the following worst-case linear optimization (WCLO) problem with uncertainties in the right-hand-side of the constraints

$$(\mathrm{WCLO}_p) \qquad \max_{b\in\mathcal{U}_p} \min_{x\in\mathcal{X}(b)} c^T x,$$

where $\mathcal{X}(b) := \{x\in\mathbb{R}^n : Ax\le b,\ x\ge 0\}$, and $\mathcal{U}_p\subseteq\mathbb{R}^m$ denotes the uncertainty set corresponding to the ℓ_p-norm defined by

$$\mathcal{U}_p := \{b = Qu + b_0 : \|u\|_p \le 1,\ u\in\mathbb{R}^r\}, \quad 1\le p\le\infty.$$

Here, $A\in\mathbb{R}^{m\times n}$, $c\in\mathbb{R}^n$, $Q\in\mathbb{R}^{m\times r}$ for some $r\le m$, $b_0\in\mathbb{R}^m$, and $\|\cdot\|_p$ denotes the ℓ_p-norm on $\mathbb{R}^r$, $1\le p<\infty$, defined by $\|u\|_p = (\sum_{i=1}^r |u_i|^p)^{1/p}$, and $\|u\|_\infty = \max_{i=1,\ldots,r}|u_i|$. As pointed out in Peng and Tao (2015), such a problem arises naturally in estimating the systemic risk in financial systems (cf. Eisenberg and Noe (2001)) where the uncertainty appears in the assets of financial institutions due to market shocks.

The WCLO model with uncertainties also appears as a subproblem in the two-stage stochastic optimization and robust optimization for a wide spectra of applications (cf. Ben-Tal and Nemirovski (1999), Bertsimas and Goyal (2010, 2012), Birge and Louveaux (1997), Shu and Song (2014)). The single-ellipsoid uncertainty set $\mathcal{U}_2$ has been a very popular choice in modeling uncertainty and approximating general convex uncertainty set (cf. Ben-Tal (1998), Ben-Tal and Nemirovski (1999)). Shu and Song (2014) considered the three uncertainty sets $\mathcal{U}_p$ $(p=1,2,\infty)$ in the two-stage robust optimization models and estimated the complexities of the corresponding models. Peng and Tao (2015) showed that (WCLO_2) is NP-hard.

One major goal of this work is to extend the results in Peng and Tao (2015), Shu and Song (2014) to the generic case with $p \in (1, \infty)$. Specifically, we show that (WCLO_p) is strongly NP-hard for fixed $p > 1$ and NP-hard when $p = \infty$. We also present a new LO reformulation for (WCLO_1) and demonstrate that our new reformulation is computationally more effective than the model in Shu and Song (2014).

Another goal of this work is to develop effective algorithms to find a strong bound or the global optimal solution for (WCLO_p) with $p = 2$ and $p = \infty$ respectively. It should be mentioned that due to the hardness of the (WCLO_2) problem, several researchers have proposed tractable approaches to obtain an approximate solution to it. For example, Ben-Tal et al. (2004) developed a two-stage robust optimization approach that can provide an upper bound to (WCLO_2) by reformulating it as a functional optimization problem and restricting feasible functional solutions to be affine. Bertsimas and Goyal (2010, 2012) further estimated the approximation rate of the solution obtained from the robust optimization model. Different from the above-mentioned robust optimization approach, Peng and Tao (2015) transformed (WCLO_2) into a non-convex quadratically constrained linear optimization (QCLO) problem, which can be relaxed to semidefinite optimization. The semidefinite relaxation (SDR) has been widely used to obtain bounds and approximate solutions for many hard optimization problems (Goemans and Williamson (1995), Nesterov (1998), Ye (1999), Anstreicher (2009), Saxena et al. (2010, 2011), Peng et al. (2015), Luo et al. (2019)). Specifically, Peng and Tao (2015) proposed an enhanced nonlinear SDR for (WCLO_2) derived by adding a non-convex quadratic cut to the standard SDR and developed a bisection search procedure to find the optimal solution of the nonlinear SDR. In this work, we will discuss how to further enhance the nonlinear SDR for (WCLO_2).

Effective global algorithms for subclasses of quadratic programming (QP) problems have been reported in the literatures. Floudas and Visweswaran (1994), Pardalos (1991) summarized various algorithms and theoretical results on QP up to that time. As pointed out in Floudas and Visweswaran (1994), most existing global algorithms for non-convex QP use a branch-and-bound (B&B) procedure, which is finitely convergent but with a complexity exponential in terms of the numbers of variables. Vandenbussche and Nemhauser (2005) used the first-order Karush–Kuhn–Tucker (KKT) conditions to develop a finite B&B algorithm for box constrained QP. Burer and Vandenbussche (2008, 2009) proposed a B&B method for linearly constrained nonconvex quadratic programming (LCQP) in which SDRs of the first-order KKT conditions of LCQP are used with finite KKT-branching. Chen and Burer (2012) further adopted the so-called completely co-positive program to improve the B&B approach for LCQP. Both the cutting plane methods and B&B approaches have been developed for a special subclass of QP (bilinear programming problems) (cf. Al-Khayyal and Falk (1983), Sherali and Alameddine (1992), Audet et al. (1999), Alarie et al. (2001), Ding and Burer (2007)). Recently, Luo et al. (2019) combined several simple optimization techniques such as the alternative direction method, convex relaxation and initialization, to develop a new global algorithm for QP with a few negative eigenvalues subject to linear and convex quadratic constraints. Numerical experiments demonstrated that the proposed approach can effectively find the globally optimal solution to large-scale nonconvex QPs when the involved Hessian matrix has only a few negative eigenvalues.

Motivated by the success of the approach in Luo et al. (2019), in this paper we develop effective global solvers for medium and large-scale problem (WCLO$_p$) with $p = 2, \infty$. For such a purpose, by following a similar procedure in Peng and Tao (2015), we first cast (WCLO$_2$) as an equivalent ℓ_2-norm maximization problem, which can further be reformulated as a QCLO problem. Then we introduce artificial variables to lift the QCLO into a higher dimensional space. We also propose to use a linear function to approximate the negative quadratic term in the constraint function of the lifted problem, resulting in a convex quadratic approximation to the original nonconvex QCLO. Based on the solution to this convex approximation problem, we develop a successive convex optimization (SCO) approach for solving QCLO that updates variable and parameter alternatively. We show that the sequence generated by the SCO approach converges to a KKT point of the reformulated QCLO problem. Then, we combine the SCO algorithm with the B&B framework, convex relaxation and initialization technique, to develop an efficient global algorithm (called SCOBB) that can find a global optimal solution to (WCLO$_2$) within a pre-specified ϵ-tolerance. We establish the convergence of the algorithm, and show that, the SCOBB algorithm has a complexity bound $\mathcal{O}\left(N \prod_{i=1}^{r} \lceil \frac{\sqrt{r}(z_u^i - z_l^i)}{2\epsilon} \rceil\right)$, where N is the complexity to solve a relaxed subproblem (a convex QP), $z_l, z_u \in \mathbb{R}^r$ denote the lower and upper bounds of z, respectively, and $-\|z\|_2^2$ is the only negative quadratic term in the constraint of the lifted problem of (WCLO$_2$). Numerical experiments illustrate that the SCOBB algorithm can effectively find a globally optimal solution for randomly generated medium and large-scale instances of (WCLO$_2$) in which the column number r of the matrix Q is less than or equal to 10. For generic (WCLO$_2$) with large r, we develop a hybrid algorithm that integrates the SCO approach with a strengthened nonlinear SDR, where the strengthened nonlinear SDR is derived by adding a series of disjunctive cuts based on the information from the solution of the current nonlinear SDR from Peng and Tao (2015). Our numerical experiments show that the solutions derived by the hybrid algorithm are optimal for most test problems, and the resulting gap is much smaller than that from nonlinear SDR in Peng and Tao (2015).

We also propose a global algorithm for (WCLO$_\infty$). By using the duality theory for LO, we first cast (WCLO$_\infty$) as an equivalent ℓ_1-norm maximization problem, which is further reformulated as an equivalent LO problem with complementarity constraints. Then we propose a finite B&B algorithm that integrates LO relaxation with finite complementarity-branching for finding a globally optimal solution of (WCLO$_\infty$). The finite convergence of the algorithm is proved as well.

The remaining of the paper is organized as follows. In Section 2, we present the equivalent convex maximization formulation for (WCLO$_p$) with $p \in (1, \infty]$. Specifically, we show that (WCLO$_p$) with $p \in (1, \infty)$ is strongly NP-hard, (WCLO$_\infty$) is NP-hard, and (WCLO$_1$) is equivalent to a tractable LO problem. In Section 3, we propose the SCO method for (WCLO$_2$) and investigate its convergence properties. In Section 4, based on the SCO method and quadratic convex relaxation, we introduce a B&B procedure to find a globally optimal solution to (WCLO$_2$). We also establish global convergence of the proposed algorithm and its complexity. In Section 5, we propose a finite B&B algorithm for (WCLO$_\infty$). In Section 6.1, we consider an application of WCLO, i.e., the worst-case estimation of the systemic risk in Eisenberg and Noe (2001). In Section 6.2, we integrate the proposed global algorithms for WCLO to develop global algorithms for the two-stage adaptive robust optimization under the ℓ_p-norm-based uncertainty set. We test the performance of the proposed algorithms and report numerical results in Section 7. Finally we conclude the paper in Section 8 by discussing some future research directions. Proofs of all the technical results are provided in the online supplement to this paper.

2. The Hardness of the WCLO Model

In this section, we first reformulate (WCLO$_p$) with $p \in (1,\infty]$ as an equivalent ℓ_q-norm maximization problem with $1 \le q < \infty$. Then we show that (WCLO$_p$) is strongly NP-hard when $p \in (1,\infty)$, NP-hard when $p = \infty$ and is polynomial-time solvable when $p = 1$. We also show that the WCLO model under a polyhedral uncertainty set is strongly NP-hard.

2.1. The Model (WCLO$_p$) with $p \in (1,\infty)$

We first reformulate (WCLO$_p$) with $p \in (1,\infty)$ as an equivalent ℓ_q-norm maximization problem and discuss how to recover a global solution to (WCLO$_p$) from the solution of the reformulated problem. For this, let $Q^T y \neq 0$, $q > 1$ and let us define $\mu(y;q) \in \mathbb{R}^r$ by

$$\mu_i(y;q) = \mathrm{sign}\left(q_i^T y\right) \left|q_i^T y\right|^{q-1} \left\|Q^T y\right\|_q^{1-q}, \quad i = 1,\dots,r, \tag{1}$$

where sign$(\cdot)$ denotes the sign function and $q_i \in \mathbb{R}^m$ denotes the i-th column of matrix Q. We have

Proposition 1. (WCLO$_p$) *with* $p \in (1,\infty)$ *has the same optimal value with the following ℓ_q-norm maximization problem*

$$\begin{aligned} &\max\ f_q(y) := \|Q^T y\|_q + b_0^T y \\ &\text{s. t. } y \in \mathcal{C} := \left\{y \in \mathbb{R}^m | A^T y \le c,\ y \le 0\right\}, \end{aligned} \tag{2}$$

where $q > 1$ and $\frac{1}{p} + \frac{1}{q} = 1$. Furthermore, let y^* be a globally optimal solution of problem (2) satisfying $Q^T y^* \neq 0$. Then we can recover a globally optimal solution of (WCLO$_p$) *with $p \in (1,\infty)$ by $b^* = Q\mu(y^*;q) + b_0$, where $\mu(y^*;q) \in \mathbb{R}^r$ is defined by (1).*

The following result follows from Theorem 4.1 in Mangasarian and Shiau (1986).

Proposition 2. *The norm maximization problem (2) is NP-hard for $q \in [1,\infty)$.*

We next present a stronger result regarding the hardness of problem (2) for fixed $q \in (1,\infty)$. We have

Theorem 1. *The norm maximization problem (2) is strongly NP-hard for $q \in (1,\infty)$.*

The proof of the above theorem follows a similar vein as the proof of the main result in Ge et al. (2011) where the authors proved the strong NP-hardness of the ℓ_q-norm minimization problem $\min_{y \in \mathcal{C}} \|y\|_q$ with $0 < q < 1$. For self-completeness, we give the proof in the online supplement to this paper.

From Theorem 1 and Proposition 1, we immediately obtain the following result.

Proposition 3. (WCLO$_p$) *is strongly NP-hard for $p \in (1,\infty)$.*

2.2. The Model (WCLO$_p$) with $p=1,\infty$

First, we reformulate (WCLO$_\infty$) as an ℓ_1-norm maximization problem.

PROPOSITION 4. (WCLO$_\infty$) *has the same optimal value with the following ℓ_1-norm maximization problem*

$$\max_{y\in\mathcal{C}} f_1(y) := \|Q^T y\|_1 + b_0^T y. \tag{3}$$

Moreover, denote by y^* the globally optimal solution of problem (3). Then we can recover a globally optimal solution of (WCLO$_\infty$) *by $b^* = Qu^* + b_0$, where $u^* \in \mathbb{R}^r$ is defined by*

$$u_i^* = \begin{cases} 1, & q_i^T y^* \geq 0, \\ -1, & q_i^T y^* < 0, \end{cases} \quad i=1,\ldots,r, \tag{4}$$

and $q_i \in \mathbb{R}^m$ denotes the i-th column vector of matrix Q.

From Propositions 4 and 2, we immediately have the following result.

PROPOSITION 5. (WCLO$_\infty$) *is NP-hard.*

Next, we show that (WCLO$_1$) can be solved via solving a series of LO problems. Let I_r be the identity matrix of order r and e_i denote the i-th column of the matrix $(I_r, -I_r)$, $i=1,\ldots,2r$.

PROPOSITION 6. *Let z_1^* be the optimal value of* (WCLO$_1$). *Then $z_1^* = \max_{i=1,\ldots,2r}\{c^T \bar{x}^i\}$, where $\bar{x}^i$ is the optimal solution of the following LO problem*

$$\min\left\{c^T x : Ax \leq Qe_i + b_0,\ x\geq 0\right\} \tag{5}$$

for $i=1,\ldots,2r$. Furthermore, let $i_0 = \arg\max_{i=1,\ldots,2r}\{c^T\bar{x}^i\}$ and $\bar{x}^{i_0}$ be the optimal solution to the corresponding LO problem in (5). Then we can recover the optimal solution of (WCLO$_1$) *by $b^* = Qe^{i_0} + b_0$.*

We remark that Shu and Song (2014) reformulate (WCLO$_1$) as the following LO problem

$$\begin{aligned} \min_{z,x^1,\ldots,x^{2r}} \quad & z \\ \text{s. t.} \quad & z \geq c^T x^i, \quad i=1,\ldots,2r, \\ & Ax^i \leq Qe_i + b_0, \quad i=1,\ldots,2r, \\ & x^i \geq 0, \quad i=1,\ldots,2r. \end{aligned} \tag{6}$$

Proposition 6 presents a decomposed approach to solve the above problem. As we shall see later, our new approach computationally outperforms the one in Shu and Song (2014).

2.3. The WCLO Model under the Polyhedron-based Uncertainty Set

Observe that the uncertainty set $\mathcal{U}_p$ is a special polyhedron when $p=1,\infty$. We then focus on the WCLO model under the general polyhedral uncertainty set,

$$\max_{b\in\mathcal{U}} \min_{x\in\mathcal{X}(b)} c^T x, \tag{7}$$

where $\mathcal{U}:=\{b=Qu+b_0 : Pu\le\xi,\ u\in[-1,1]^r\}$ with $P\in\mathbb{R}^{l\times r}$ and $\xi\in\mathbb{R}^l$ denotes the polyhedral uncertainty set. Note that problem (7) can be converted into a linear max-min (LMM) problem of the form $\max\limits_{u\in\mathcal{V}} \min\limits_{x\in\mathcal{L}(u)} c^T x$, where $\mathcal{V}:=\{u\in[-1,1]^r : Pu\le\xi\}$, and $\mathcal{L}(u):=\left\{x\in\mathbb{R}^n_+ : Ax-Qu\le b_0\right\}$. Hansen et al. (1992) showed that the LMM problem is strongly NP-hard (see Theorem 3.1). Therefore, we can obtain the following result.

Proposition 7. *The WCLO problem (7) is strongly NP-hard.*

It is worth mentioning that both problems (2) and (3) are non-convex and their global optimal solutions are hard to obtain. In the subsequent sections, we will develop effective algorithms to find global optimal solutions to these two problems.

3. The SCO Method and Its Convergence Properties

In this section, we present a successive convex optimization (SCO) approach for problem (2) with $q=2$, and explore its convergence property to a KKT point of the reformulated QCLO problem. To start, we assume that the following assumption always holds.

Assumption 1. *The set* $\mathcal{C}:=\left\{y\in\mathbb{R}^m | A^T y\le c,\ y\le 0\right\}$ *is bounded.*

The above assumption ensures the existence of optimal solutions of problem (2). One can show that if the matrix A is of full row rank, then Assumption 1 always holds. We also observe that if $Q^T y=0$ for any $y\in\mathcal{C}$, then problem (2) reduces to an LO problem $\max\limits_{y\in\mathcal{C}} b_0^T y$. To avoid this trivial case, we make the following assumption.

Assumption 2. *There exists some* $y^0\in\mathcal{C}$ *such that* $Q^T y^0\ne 0$.

As mentioned in Peng and Tao (2015), problem (2) with $q=2$ can be reformulated as the following linear optimization problem with a non-convex quadratical constraint (QCLO):

$$\begin{aligned} &\max\ t \\ &\text{s. t. }\ h(t,y):=(t-b_0^T y)^2-\|Q^T y\|_2^2\le 0, \\ &\qquad y\in\mathcal{C}. \end{aligned} \tag{8}$$

Next we lift the above problem to a higher dimensional space as follows:

$$\begin{aligned} &\max\ t \\ &\text{s. t. } (t-b_0^T y)^2 - \|z\|_2^2 \le 0, \\ &\qquad Q^T y = z, \quad y \in \mathcal{C}. \end{aligned} \tag{9}$$

Let z_l and $z_u \in \mathbb{R}^r$ be lower and upper bounds of $z = Q^T y$ over $\mathcal{C}$, obtained via solving the following LO problems respectively

$$z_l^i = \min_{y\in\mathcal{C}} q_i^T y, \quad z_u^i = \max_{y\in\mathcal{C}} q_i^T y, \quad i = 1, \cdots, r, \tag{10}$$

where q_i is the i-th column vector of matrix Q. For every $\mu \in [z_l, z_u]$, it holds

$$-\|z\|_2^2 \le -\|\mu\|_2^2 - 2\mu^T(z-\mu) = -2\mu^T z + \|\mu\|_2^2, \quad \forall z \in [z_l, z_u]. \tag{11}$$

Take $\mu = Q^T y^0 \neq 0$ for some $y^0 \in \mathcal{C}$ by Assumption 2. Using the linear function $-2\mu^T z + \|\mu\|_2^2$ to approximate the negative quadratic term $-\|z\|_2^2$ in the constraint of lifted problem (9), we derive the following quadratic convex approximation

$$\begin{aligned} &\max_{t,y}\ t \\ &s.\ t.\ g_\mu(t,y) := (t-b_0^T y)^2 - 2\mu^T Q^T y + \|\mu\|_2^2 \le 0, \\ &\qquad y \in \mathcal{C}, \end{aligned} \tag{12}$$

where $\mu = Q^T y^0 \neq 0$ for some $y^0 \in \mathcal{C}$. Note that inequality (11) with $z = Q^T y$ implies that $g_\mu(t,y) \ge h(t,y)$, thus the optimal solution of problem (12) is also feasible to problem (8). Hence, the objective function value at the optimal solution of problem (12) provides a lower bound to problem (8).

Denote by $\mathcal{F} = \{(t,y) \in \mathbb{R}\times\mathcal{C} \mid h(t,y) \le 0\}$ the feasible set of problem (8). Let us define

$$\mathcal{F}_\mu = \{(t,y) \in \mathbb{R}\times\mathcal{C} \mid g_\mu(t,y) \le 0\}, \quad \text{int}\mathcal{F}_\mu = \{(t,y) \in \mathbb{R}\times\mathcal{C} \mid g_\mu(t,y) < 0\}.$$

We now state a simple property of the sets $\mathcal{F}_\mu$ and $\text{int}\mathcal{F}_\mu$.

Lemma 1. *Let $\mu = Q^T y^0 \neq 0$ for some $y^0 \in \mathcal{C}$. Then $\mathcal{F}_\mu$ is a nonempty closed convex set in $\mathcal{F}$ and* $\text{int}\mathcal{F}_\mu \neq \emptyset$.

From Lemma 1, we see that problem (12) is feasible and well-defined, and the Slater condition holds for problem (12). It follow immediately

PROPOSITION 8. *Suppose that $\bar{\mu} = Q^T\bar{y} \neq 0$ for some $\bar{y} \in \mathcal{C}$. If $(\bar{t}, \bar{y})$ is the optimal solution of problem (12) with $\mu = \bar{\mu}$, then $(\bar{t}, \bar{y})$ is a KKT point of problem (8).*

We now describe the SCO method for problem (8).

ALGORITHM 1 (**Successive Convex Optimization SCO**(y^0, ϵ)).

Input: Initial point $y^0 \in \mathcal{C}$ with $Q^T y^0 \neq 0$ and stopping criterion $\epsilon > 0$;

Step 0 Set $t^0 = b_0^T y^0$ and $\mu^0 = Q^T y^0$. Set $k = 0$.

Step 1 Solve problem (12) with $\mu = \mu^k$ to obtain the optimal solution (t^{k+1}, y^{k+1}). Set $\mu^{k+1} = Q^T y^{k+1}$.

Step 2 If $\|\mu^{k+1} - \mu^k\| > \epsilon$, then set $k = k+1$ and go back to Step 1; Otherwise, stop and output (t^k, y^k) as the final solution.

Note that since $\{y^k\} \subseteq \mathcal{C}$ and $\mathcal{C}$ is bounded by Assumption 1, there exists at least one accumulation point for the sequence $\{y^k\}$ generated by Algorithm 1.

We next present several technical results regarding the sequences $\{(t^k, y^k)\}$ and $\{\mu^k\}$ generated by Algorithm 1. We have

LEMMA 2. *Let the sequence $\{(t^k, y^k)\}$ be generated by Algorithm 1. Then $\{(t^k, y^k)\} \subseteq \mathcal{F}$.*

LEMMA 3. *Let the sequence $\{\mu^k\}$ be generated by Algorithm 1. Then $\mu^k \neq 0$ and $\mathrm{int}\mathcal{F}_{\mu^k} \neq \emptyset$ for all k.*

LEMMA 4. *Let the sequence $\{t^k\}$ be generated by Algorithm 1. Then $\{t^k\}$ is a nondecreasing and convergent sequence.*

Based on Lemmas 1-4, we obtain the following lemma.

LEMMA 5. *Let the sequence $\{(t^k, \mu^k, y^k)\}$ be generated by Algorithm 1 with an accumulation point $(\hat{t}, \hat{\mu}, \hat{y})$ satisfying $Q^T\hat{y} \neq 0$. Then, $(\hat{t}, \hat{y}) \in \mathcal{F}_{\hat{\mu}}$ and $t \leq \hat{t}$ for any $(t, y) \in \mathcal{F}_{\hat{\mu}}$.*

Lemma 5 indicates that any accumulation point $(\hat{t}, \hat{y})$ of the sequence $\{(t^k, y^k)\}$ generated by Algorithm 1 is the optimal solution to problem (12) with $\mu = \hat{\mu} = Q^T\hat{y} \neq 0$. Combining the above lemma with Proposition 8, we immediately obtain the following theorem.

THEOREM 2. *Let $\epsilon = 0$ and $\{(t^k, y^k)\}$ be a sequence generated by Algorithm 1. Then, any accumulation point $(\bar{t}, \bar{y})$ of the sequence $\{(t^k, y^k)\}$ with $Q^T\bar{y} \neq 0$ is a KKT point of problem (8).*

Theorem 2 shows that the SCO algorithm converges to a KKT point of problem (8). As we shall see later, this property of the SCO facilitates the design of new global solver for problem (8). For example, the solution derived by the SCO can be used as a lower bound in the new B&B approach to be introduced in the next section.

3.1. Initialization of the SCO algorithm

In this subsection, we describe how to find a starting point y^0 or multiple starting points for the SCO algorithm. For this, we propose to solve the following LO problems

$$\max_{y,z}\left\{\xi_j^T z : y \in \mathcal{C},\ Q^T y = z\right\}, \quad j = 1, \dots, \rho, \tag{13}$$

where ρ and ξ_j are chosen either by $\rho = 2^r$ and $\xi_j \in \{-1, 1\}^r$, $j \in \{1, \dots, \rho\}$, or by $\rho = 2r$, $\xi_j = e_j$, $\xi_{j+r} = -e_j$, $j = 1, \dots, r$. Here, e_j denotes the j-th unit vector of $\mathbb{R}^r$ and r is the column number of matrix Q.

The following proposition states that solving the series of LO problems in (13) can generate at least one feasible solution $y^0 \in \mathcal{C}$ with $Q^T y^0 \neq 0$.

Proposition 9. *Let (y_j^0, z_j^0) be the optimal solution of problem (13) for $j = 1, \dots, \rho$. Then $J = \left\{j : z_j^0 \neq 0, j = 1, \dots, \rho\right\} \neq \emptyset$.*

4. A Global Optimization Algorithm for (WCLO$_2$)

In this section, we develop an algorithm to find a global optimal solution of (WCLO$_2$) within a prescribed ϵ tolerance via combining the SCO approach, a branch-and-bound (B&B) framework, convex relaxation and initialization techniques. We also establish the convergence of the algorithm and estimate its complexity.

4.1. The Quadratic Convex Relaxation

We consider a restricted version of the lifted problem (9) where the variable z is in a rectangle $[l, u]$:

$$\begin{aligned} &\max\ t \\ &\text{s.t.}\ \ (t - b_0^T y)^2 - \|z\|_2^2 \leq 0, \\ &\qquad Q^T y = z, \quad z \in [l, u], \quad y \in \mathcal{C}, \end{aligned} \tag{14}$$

where $l,u\in\mathbb{R}^r$ with $[l,u]\subseteq[z_l,z_u]$, and $z_l,z_u\in\mathbb{R}^r$ are given in (10). Let $s_i=z_i^2$ for $i=1,\dots,r$. The convex envelope of $s_i=z_i^2$ on $[l_i,u_i]$ is $\{(s_i,z_i): z_i^2\le s_i,\ s_i\le(l_i+u_i)z_i-l_iu_i\}$. We can then derive the following convex relaxation for problem (14):

$$\begin{aligned}\max_{t,y,z,s}\ & t \\ \text{s.t.}\ & (t-b_0^Ty)^2-\sum_{i=1}^r s_i\le 0,\\ & Q^Ty=z,\quad y\in\mathcal{C},\quad z\in[l,u],\\ & z_i^2\le s_i,\quad s_i\le(l_i+u_i)z_i-l_iu_i,\quad i=1,\dots,r.\end{aligned} \tag{15}$$

As pointed out in the introduction, there exist other strong relaxation models such as the linear and nonlinear SDR for (WCLO_2) that usually involve intensive computation. In this work, we integrate the relaxation model (15) with other simple optimization techniques to develop a global algorithm for (WCLO_2). Our choice is based on the relative simplicity of the relaxation model (15) and its good approximation behavior as shown in our next theorem, which compares the objective values at the optimal solutions to problem (14) and its relaxation (15).

THEOREM 3. *Let $f^*_{[l,u]}$ and $v^*_{[l,u]}$ be the optimal values of problem (14) and its relaxation (15), respectively. Let $(\bar t,\bar y,\bar z,\bar s)$ be the optimal solution to problem (15). Then,*

$$0\le v^*_{[l,u]}-f^*_{[l,u]}\le v^*_{[l,u]}-f_2(\bar y)\le\sqrt{\sum_{i=1}^r(\bar s_i-\bar z_i^2)}\le\frac{1}{2}\|u-l\|_2. \tag{16}$$

We now rewrite problem (14) as the following

$$\begin{aligned}\max\ & f_2(y):=\|Q^Ty\|_2+b_0^Ty\\ \text{s.t.}\ & l\le Q^Ty\le u,\quad y\in\mathcal{C}.\end{aligned} \tag{17}$$

Note that the problems (14) and (17) are equivalent in the sense that they have the same optimal solutions and optimal value.

From Theorem 3, we immediately have the following corollary.

COROLLARY 1. *Let $(\bar t,\bar y,\bar z,\bar s)$ be an optimal solution to problem (15) and $\epsilon>0$. If $\sum_{i=1}^r[\bar s_i-\bar z_i^2]\le\epsilon^2$, then $\bar y$ is an ϵ-optimal solution to problem (17).*

Theorem 3 indicates that when $\|u-l\|_\infty$ is very small, the relaxed model (15) can provide a good approximation to problem (17). Moreover, from Theorem 3 and Corollary 1, if $\|u-l\|_\infty \le 2\epsilon/\sqrt{r}$, then $\bar{y}$ can be viewed as an ϵ-approximation solution to problem (17). Motivated by this observation, we can divide the interval $[z_l^i, z_u^i]$ into $\lceil \frac{\sqrt{r}(z_u^i - z_l^i)}{2\epsilon} \rceil$ subintervals such that each subinterval has a width of $2\epsilon/\sqrt{r}$, where z_l^i, z_u^i are defined in (10). Then we can solve the relaxed problem (15) for every sub-rectangle to obtain an ϵ-approximate solution for the restricted problem (17) over this sub-rectangle. After that, we can choose the best solution from the obtained approximate solutions as the final solution, which is clearly a global ϵ-approximate solution to problem (8). From the above discussion, we can see that such a partitioning procedure can provide an ϵ-approximate solution to problem (8). For convenience, we call such a procedure the brutal force algorithm. The following result is an immediate consequence of our above discussion.

THEOREM 4. *The brutal force partitioning algorithm can find a global ϵ-approximate solution to problem (8) in $\mathcal{O}\left(N\prod_{i=1}^r \lceil \frac{\sqrt{r}(z_u^i - z_l^i)}{2\epsilon} \rceil\right)$ time, where N is the complexity to solve problem (15).*

We remark that the brutal force algorithm is too conservative because it searches within very small sub-rectangles and thus is not effective. In the next subsection, we will discuss how to integrate the SCO algorithm with other techniques to develop an effective global algorithm for problem (8).

4.2. The SCOBB Algorithm

In this subsection, we present a global algorithm (termed SCOBB) for problem (8) that integrates the SCO algorithm with branch-and-bound techniques based on quadratic convex relaxation. Different from the other existing global algorithms for LCQP, the SCOBB algorithm has the following two features:

(i) It can either check the global optimality of the solutions computed by SCO, or improve the lower bound by restarting SCO under certain circumstance;

(ii) It can utilize the lower bound computed by the SCO to accelerate the convergence.

We are now ready to present the new global algorithm for problem (8).

ALGORITHM 2 (**The SCOBB Algorithm**).

Input: Q, c, A, b_0, and stopping criteria $\epsilon > 0$.

Output: an ϵ-optimal solution y^*.

Step 0 (**Initialization**)

(i) Set $l^0 = z_l$ and $u^0 = z_u$, where z_l and z_u are computed by (10).

(ii) Let r be the column number of matrix Q. If $r \le 5$, set $\rho = 2^r$, $\xi_j \in \{-1,1\}^r$, $j \in \{1,\dots,\rho\}$. Else, set $\rho = 2r+2$, $\xi_j = e_j$, $\xi_{j+r} = -e_j$, $j = 1,\dots,r$, $\xi_{2r+1} = e$, $\xi_{2r+2} = -e$, where e_j denotes the j-th unit vector of $\mathbb{R}^r$, and $e \in \mathbb{R}^r$ is the vector of all ones. Solve the LO problems in (13) to get optimal solutions $(\bar{y}_j, \bar{z}_j)$, $j = 1,\dots,\rho$. Set $J = \{j : \bar{z}_j \neq 0, j = 1,\dots,\rho\}$.

Step 1 Find KKT points y_j^0 of problem (8) by running SCO(y^0,ϵ) with $y^0 = \bar{y}_j$ for $j \in J$. Set $y^* = \arg\max\left\{f_2(y_j^0), j \in J\right\}$, $v^* = f_2(y^*)$.

Step 2 Solve problem (15) over $[l,u] = [l^0,u^0]$ to obtain an optimal solution (t^0, y^0, z^0, s^0). If $f(y^0) > v^*$, then update lower bound $v^* = f(y^0)$ and solution $y^* = y^0$. Set $k = 0$, $\mathcal{B}^k := [l^k, u^k]$, $\Omega := \left\{[\mathcal{B}^k, (t^k, y^k, z^k, s^k)]\right\}$.

Step 3 **While** $\Omega \neq \emptyset$ **Do** (the main loop)

(S3.1) (**Node Selection**) Choose a node $[\mathcal{B}^k, (t^k, y^k, z^k, s^k)]$ from Ω with the largest upper bound t^k and delete it from Ω.

(S3.2) (**Termination**) If $t^k \le v^* + \epsilon$, then y^* is an ϵ-optimal solution to problem (8), stop.

(S3.3) (**Partition**) Choose $i^* = \arg\max\limits_{i=1,\dots,r}\left\{s_i^k - (z_i^k)^2\right\}$. Set $w_{i^*} = \frac{l_{i^*}^k + u_{i^*}^k}{2}$,

$$\Gamma_k(w_{i^*}) = \left\{ (s_{i^*}, z_{i^*}) \;\middle|\; \begin{array}{l} s_{i^*} > (l_{i^*}^k + w_{i^*}) z_{i^*} - l_{i^*}^k w_{i^*} \\ s_{i^*} > (w_{i^*} + u_{i^*}^k) z_{i^*} - w_{i^*} u_{i^*}^k \end{array} \right\}.$$

If $(s_{i^*}^k, z_{i^*}^k) \in \Gamma_k(w_{i^*})$, then set the branching value $\beta_{i^*} = w_{i^*}$; else set $\beta_{i^*} = z_{i^*}^k$. Partition $\mathcal{B}^k$ into two sub-rectangles $\mathcal{B}^{k_1}$ and $\mathcal{B}^{k_2}$ along the edge $[l_{i^*}^k, u_{i^*}^k]$ at point β_{i^*}.

(S3.4) For $j = 1,2$, solve problem (15) over $\mathcal{B}^{k_j}$ to obtain an optimal solution $(t^{k_j}, y^{k_j}, z^{k_j}, s^{k_j})$, set $\Omega = \Omega \cup \left\{[\mathcal{B}^{k_j}, (t^{k_j}, y^{k_j}, z^{k_j}, s^{k_j})]\right\}$.

(S3.5) (**Restart SCO**) Set $\hat{y} = \arg\max\{f_2(y^{k_1}), f_2(y^{k_2})\}$. If $f_2(\hat{y}) > v^*$, then find a KKT point $\bar{y}^k$ of problem (8) by running SCO$(\hat{y},\epsilon)$, and update solution $y^* = \arg\max\{f_2(\hat{y}), f_2(\bar{y}^k)\}$ and lower bound $v^* = f_2(y^*)$.

(S3.6) (**Node deletion**) Delete from Ω all the nodes $[\mathcal{B}^j, (t^j, y^j, z^j, s^j)]$ with $t^j \le v^* + \epsilon$. Set $k = k+1$.

End while

We list three main differences between the SCOBB and other existing global algorithms for LCQP in the literatures (cf. Burer and Vandenbussche (2008, 2009)).

(i) In Step 1, we apply the SCO algorithm from different feasible points of the lifted problem (14) as starting points to find a good feasible solution of problem (8).

(ii) In Step (S3.5), we restart SCO to find a better feasible solution if the objective function value at the feasible point derived from the solution of the relaxation problem is greater than the current lower bound.

(iii) In Step (S3.3), we cut off the optimal solution of the relaxation problem after each iteration to improve the upper bound (see an illustration in Figure 1 in Luo et al. (2019)).

We next present several technical results for the sequences $\{s^k\}$ and $\{z^k\}$ generated by Algorithm 2. Recall that (y^k, t^k, s^k, z^k) is the optimal solution of problem (15) over $\mathcal{B}^k$. From the proof of Theorem 3, we immediately have the following results.

LEMMA 6. *For each k, $s_i^k - (z_i^k)^2 \le \frac{1}{4}(u_i^k - l_i^k)^2$, $i = 1, \dots, r$.*

LEMMA 7. *At the k-th iteration, if $\max\limits_{i=1,\dots,r}\{s_i^k - (z_i^k)^2\} \le \frac{\epsilon^2}{r}$, then Algorithm 2 stops and both y^* and y^k are global ϵ-approximate solutions to problem (8).*

We now establish the convergence of Algorithm 2 based on Lemmas 6 and 7.

THEOREM 5. *Algorithm 2 can find a global ϵ-approximate solution to problem (8) by solving at most $\prod_{i=1}^r \lceil \frac{\sqrt{r}(z_u^i - z_l^i)}{2\epsilon} \rceil$ relaxed subproblem (15).*

4.3. The SCO-NLSDR Algorithm

The complexity of SCOBB grows exponentially in terms of r (the column number of the matrix Q), which indicates that SCOBB may not be efficient for instances of (WCLO_2) with large r. To remedy this, in this subsection, we propose a mixed algorithm (called SCO-NLSDR) for (WCLO_2) with large r by combining the SCO algorithm with the nonlinear SDR (NLSDR) from Peng and Tao (2015) and the disjunctive cut technique from Saxena et al. (2010). Specifically, we strengthen the NLSDR from Peng and Tao (2015) to obtain a tighter upper bound by adding the series of disjunctive cuts based on the information from the solution of the current NLSDR.

We denote by $\mathrm{Tr}(\cdot)$ the trace of a matrix, and by $\mathcal{S}^m$ the space of $m \times m$ real symmetric matrices. For $B \in \mathcal{S}^m$, the notation $B \succeq 0$ means that the matrix B is positive semidefinite. Let us first describe the NLSDR for (WCLO_2) proposed by Peng and Tao (2015) as follows:

$$\max_{(y,Y)\in\mathcal{Y}} b_0^T y + \sqrt{(b_0^T y)^2 + \mathrm{Tr}\left((QQ^T - b_0 b_0^T)Y\right)}, \tag{18}$$

where

$$\mathcal{Y}=\left\{(y,Y)\in\mathbb{R}^m\times\mathcal{S}^m \,\middle|\, \begin{array}{l} A^Ty\le c,\ y\le 0,\ A^TY\ge cy^T, \\ Y-yy^T\succeq 0,\ Y\ge 0, \end{array}\right\}.$$

As pointed out by Peng and Tao (2015), the NLSDR (18) can provide a tight upper bound for the optimal value of (WCLO_2). A bisection search algorithm (BSA) was proposed in Peng and Tao (2015) for finding the optimal solution of problem (18) in polynomial time.

Next, we present a strengthened NLSDR for (WCLO_2) by adding disjunctive cuts to the NLSDR (18). For this, we describe the approach for generating disjunctive cuts in Saxena et al. (2010). Let $(\hat{y},\hat{Y})$ be the solution to the NLSDR (18) which we want to cut off. Let $\lambda_1\ge\ldots\ge\lambda_q>\lambda_{q+1}\ldots=\lambda_n=0$ be eigenvalues of the matrix $\hat{Y}-\hat{y}\hat{y}^T$, and let $p_1,\ldots,p_n$ be the corresponding set of orthonormal eigenvectors. Let $p=p_k$, $k\in\{1,\ldots,q\}$. We define

$$\eta_l(p)=\min_{(y,Y)\in\mathcal{Y}} p^Ty,\quad \eta_u(p)=\max_{(y,Y)\in\mathcal{Y}} p^Ty. \tag{19}$$

Let $\theta=p^T\hat{y}$. As pointed out in Saxena et al. (2010), the following disjunction can be derived by splitting the range $[\eta_l(p),\eta_u(p)]$ of the function p^Ty over $\mathcal{Y}$ into two intervals $[\eta_l(p),\theta]$ and $[\theta,\eta_u(p)]$ and constructing a secant approximation of the function $-(p^Ty)^2$ in each of the intervals, respectively.

$$\left[\begin{array}{c}\eta_l(p)\le p^Ty\le\theta \\ -(p^Ty)(\eta_l(p)+\theta)+\theta\eta_l(p)\le -\mathrm{Tr}(pp^TY)\end{array}\right]\bigvee\left[\begin{array}{c}\theta\le p^Ty\le\eta_u(p) \\ -(p^Ty)(\eta_u(p)+\theta)+\theta\eta_u(p)\le -\mathrm{Tr}(pp^TY)\end{array}\right].$$

The above disjunction can be used to derive the following disjunctive cuts by using the apparatus of CGLP (see Theorem A.1 in Appendix of Luo et al. (2019) for CGLP):

$$\alpha_k^Ty+\mathrm{Tr}(U_kY)\ge\beta_k,\quad k=1,\ldots,q. \tag{20}$$

Adding disjunctive cuts (20) to the NLSDR (18) yeilds an enhanced NLSDR for (WCLO_2):

$$\begin{aligned} &\max\ b_0^Ty+\sqrt{(b_0^Ty)^2+\mathrm{Tr}\left((QQ^T-b_0b_0^T)Y\right)}, \\ &s.\ t.\ \alpha_k^Ty+\mathrm{Tr}(U_kY)\ge\beta_k,\quad k=1,\ldots,q, \\ &\qquad (y,Y)\in\mathcal{Y}. \end{aligned} \tag{21}$$

We now present the following hybrid algorithm for (WCLO_2) with large r based on the NLSDR, disjunctive cut technique and the SCO approach.

ALGORITHM 3 (**The SCO-NLSDR Algorithm**).

Input: Q, c, A, b_0, and stopping criteria $\epsilon > 0$.

Output: lower bound v^* and upper bound $\hat{v}$.

Step 1 Use the BSA algorithm in Peng and Tao (2015) to solve problem (18) to obtain an optimal solution $(\hat{y}, \hat{Y})$ and optimal value $\hat{v}$.

Step 2 Use the CGLP apparatus to generate disjunctive cuts from the eigenvectors of the matrix $\hat{Z} = \hat{Y} - \hat{y}\hat{y}^T$ with positive eigenvalues to cut off the solution $(\hat{y}, \hat{Y})$.

Step 3 If a violated cut was generated, then add generated disjunctive cuts to problem (18) and go to Step 1. Else, go to Step 4.

Step 4 Run SCO$(\hat{y}, \epsilon)$ to obtain a KKT point $\bar{y}$ of problem (8), set $v^* = f_2(\bar{y})$, stop and output v^* and $\hat{v}$ as the final lower bound and upper bound, respectively.

Numerically, the loop of the algorithm is repeated until a time-limit of 60 minutes is reached or the code is unable to find any violated cut. In the main loop, we generate a disjunctive cut from each positive eigenvalue of $\hat{Z}$ by using CGLP to strengthen the NLSDR to obtain a tighter upper bound of problem (8). At Step 4, we apply the SCO algorithm from the solution of strengthened NLSDR as a starting point to find a better feasible solution of problem (8). We will report our numerical results for Algorithm 3 in Tables 1 and 2 of Section 3 in the online supplement to this paper. As we can see later from the tables, the solutions derived by SCO are optimal for most of test problems. We also observe that the gap between the lower and upper bounds derived from Algorithm 3 is much smaller than that from the NLSDR (18).

5. A Global Algorithm for (WCLO$_\infty$)

In this section, we propose a finite B&B algorithm for (WCLO$_\infty$) to globally solve the underlying problem via combining LO relaxation and complementarity branching technique.

We start by reformulating problem (3) into the following lifted problem,

$$\max \left\{ \|z\|_1 + b_0^T y : \; y \in \mathcal{C}, \; Q^T y = z \right\}. \tag{22}$$

For any $z \in \mathbb{R}^r$, let us define a vector $\mu \in \mathbb{R}^{2r}$ whose component is

$$\mu_i = \max\{z_i, 0\}, \quad \mu_{i+r} = -\min\{z_i, 0\}, \quad i = 1, \ldots, r. \tag{23}$$

Clearly, we have

$$z_i = \mu_i - \mu_{i+r}, \quad |z_i| = \mu_i + \mu_{i+r}, \quad \mu_i \mu_{i+r} = 0, \quad i = 1, \ldots, r, \quad \mu \geq 0.$$

Then problem (22) can be further reformulated as the following LO problem with complementarity constraints:

$$\begin{aligned} \max_{(y,\mu)\in\mathbb{R}^{m+2r}} \quad & e^T\mu + b_0^T y \\ \text{s.t.} \quad & Q^T y = P\mu, \quad y \in \mathcal{C}, \quad 0 \leq \mu \leq \hat{\mu}, \\ & \mu_i \mu_{i+r} = 0, \ i = 1, \ldots, r, \end{aligned} \tag{24}$$

where $e \in \mathbb{R}^{2r}$ is the vector of all ones, $P = (I_r, -I_r) \in \mathbb{R}^{r\times 2r}$, I_r is the identity matrix of order r, and $\hat{\mu} \in \mathbb{R}^{2r}$ is the upper bound of μ defined by

$$\hat{\mu}_i = \max\{z_u^i, 0\}, \quad \hat{\mu}_{i+r} = -\min\{z_l^i, 0\}, \quad i = 1, \ldots, r,$$

where z_l and z_u are given in (10). By dropping the complementarity constraint in (24), we can obtain the following LO relaxation:

$$\max_{(y,\mu)\in\mathcal{H}} e^T\mu + b_0^T y, \tag{25}$$

where $\mathcal{H} = \left\{(y,\mu) \in \mathbb{R}^{m+2r} : Q^T y = P\mu, \ y \in \mathcal{C}, \ \ 0 \leq \mu \leq \hat{\mu}\right\}$.

Based on relaxation (25) and complementarity branching technique, we propose a finite B&B algorithm (called FBB) for problem (3) as follows.

Algorithm 4 (**The FBB Algorithm**).

Input: Q, c, A, b_0, and stopping criteria $\epsilon > 0$.

Step 0 (**Initialization**) Solve the following problem to get the optimal solution $(\bar{y}, \bar{z})$:

$$\max\left\{e^T z + b_0^T y : y \in \mathcal{C}, \ Q^T y = z\right\}.$$

Compute $\bar{\mu}$ by (23). Set $\gamma = e^T\bar{\mu} + b_0^T\bar{y}$.

Step 1 Solve problem (25) to obtain an optimal solution (y^0, μ^0) and an initial upper bound v^0. Set $k = 0$, $R_k = [0, \hat{\mu}]$, $\mathcal{R} = \{[R_k, (y^k, \mu^k, v^k)]\}$.

Step 2 Choose $i^* = \arg\max_{i=1,\ldots,r}\{\mu_i^k \mu_{i+r}^k\}$. Partition R_k into R_{k_1} and R_{k_2} via the index i^* by the branching procedure:

$$R_{k_1} = \{\mu \in R_k : \mu_{i^*} = 0\}, \quad R_{k_2} = \{\mu \in R_k : \mu_{i^*+r} = 0\}.$$

Step 3 Solve problem (P_{k_j}) to obtain an optimal solution (y^{k_j}, μ^{k_j}) and optimal value v^{k_j}:

$$(P_{k_j}) \quad \max\left\{e^T\mu + b_0^T y : (y,\mu) \in \mathcal{H}, \quad \mu \in R_{k_j}\right\}, \quad j = 1, 2.$$

Set $\mathcal{R} = \mathcal{R} \cup \left\{[R_{k_1}, (y^{k_1}, \mu^{k_1}, v^{k_1})], [R_{k_2}, (y^{k_2}, \mu^{k_2}, v^{k_2})]\right\}$. For $j = 1, 2$, if (y^{k_j}, μ^{k_j}) is feasible to problem (24) and $v^{k_j} > \gamma$, then update lower bound $\gamma = v^{k_j}$ and $(y^*, \mu^*) = (y^{k_j}, \mu^{k_j})$.

Step 4 Delete from $\mathcal{R}$ all the nodes $[R_j, (y^j, \mu^j, v^j)]$ with $v^j \le \gamma + \epsilon$. Choose a node $[R_k, (y^k, \mu^k, v^k)]$ from $\mathcal{R}$ with the largest upper bound v^k and delete it from $\mathcal{R}$.

Step 5 If $\mathcal{R} = \emptyset$, then stop, (y^*, μ^*) is an ϵ-optimal solution. Else, set $k = k + 1$, go to Step 2.

We then have the finite convergence of the algorithm, due to the complementarity branching rule used in the algorithm.

Theorem 6. *Algorithm 4 terminates after finitely many iterations and obtains an ϵ-optimal solution of problem (3).*

6. Some Application Examples

In this section, we describe a couple of application examples of the WCLO model.

6.1. Estimating the Worst-Case Systemic Risk

We first describe an example of the WCLO model for estimating the worst-case systemic risk in financial systems in Peng and Tao (2015). Consider an interbank network consisting of n banks. Let $L \in \Re^{n\times n}$ be the liability matrix in the network where the element L_{ij} represents the liability of bank i to bank j. Naturally, we can assume that $L_{ij} \ge 0$ for $i \ne j$ and $L_{ii} = 0$. Let $\hat{b} \ge 0$ be the asset vector where $\hat{b}_i$ represents the asset of bank i. The systemic loss of a financial system can be found via solving the following linear optimization problem (cf. Eisenberg and Noe (2001)):

$$\begin{aligned} \min_{x\in\mathbb{R}^n} \ & \sum_{i=1}^n (1 - x_i) \\ \text{s.t.} \ & \left(\sum_{j=1}^n L_{ij}\right) x_i - \sum_{j=1}^n L_{ji} x_j \le \hat{b}_i, \quad i = 1, \ldots, n, \\ & 0 \le x_i \le 1, \quad i = 1, \ldots, n. \end{aligned} \tag{26}$$

Here $x_i \in [0, 1]$ $(i = 1, \ldots, n)$, represents the percentage of the payment from bank i proportional to its total liability, and $1 - x_i$ denotes the percentage of the total liability bank i default.

Note that the asset of a financial institution is usually subject to market shocks, i.e., there exists uncertainties in the asset of a bank. To account for the uncertain asset, Peng and Tao (2015) introduced the worst-case systemic risk (WCSR) problem defined as follows

$$\max_{u\in\mathcal{B}_p}\ \min_{x\in\mathcal{F}_1(u)}\ e^T(e-x), \tag{27}$$

where $e\in\mathbb{R}^n$ is the vector of all ones, $\mathcal{B}_p := \{u\in\mathbb{R}^r : \|u\|_p\le 1\}$ with $p=1,2,\infty$ denotes the uncertainty set, and

$$\mathcal{F}_1(u) := \left\{x\in\mathbb{R}^n : \left(\mathrm{diag}(Le)-L^T\right)x \le Qu+\hat{b},\ 0\le x\le e\right\},$$

with $Q\in\mathbb{R}^{n\times r}$ for some $r\le n$. Here, $\mathrm{diag}(d)$ denotes the diagonal matrix whose diagonal is d for a vector $d\in\mathbb{R}^n$.

In order to formulate (27) into the form of (WCLO$_p$), we consider the following problem by removing the non-negativity constraint in the inner problem of (27) (cf. Khabazian and Peng (2019)),

$$\max_{u\in\mathcal{B}_p}\ \min_{x\in\mathcal{F}_2(u)}\ e^T(e-x), \tag{28}$$

where

$$\mathcal{F}_2(u) = \left\{x\in\mathbb{R}^n : \left(\mathrm{diag}(Le)-L^T\right)x \le Qu+\hat{b},\ x\le e\right\}.$$

Under the assumption that $\mathcal{F}_1(u)\neq\emptyset$ for any $u\in\mathcal{B}_p$, one can show the following result.

Proposition 10. *The problems (27) and (28) have the same optimal value and optimal solutions.*

Let $z=e-x$, problem (28) can be written as

$$\max_{u\in\mathcal{B}_p}\ \min_{z\in\mathcal{F}(u)}\ e^T z, \tag{29}$$

where

$$\mathcal{F}(u) = \left\{z\in\mathbb{R}^n : Az\le Qu+b_0,\quad z\ge 0\right\},$$

with $A=L^T-\mathrm{diag}(Le)$ and $b_0=Ae+\hat{b}$. It is easy to see that problem (29) reduces to a special case of problem (2) where the associated set $\mathcal{C}$ is defined by $\mathcal{C}=\left\{y\in\mathbb{R}^m | A^Ty\le e,\ y\le 0\right\}$. Note that $A^Te=0$ and hence the rank of A is at most $n-1$. Since $A^T(-e)=0<e$, the interior of $\mathcal{C}$ is nonempty and unbounded. Therefore, the global

methods developed in the previous sections can not be applied directly to solve problem (29) with $p=2,\infty$. To overcome this difficulty, we now construct the following auxiliary problem,

$$\max_{u\in\mathcal{B}_p}\ \min_{(z,v)\in\mathcal{F}_0(u)}\ e^Tz+\bar{M}v \tag{30}$$

where $\bar{M}>0$ is sufficiently large, and

$$\mathcal{F}_0(u)=\left\{(z,v)\in\mathbb{R}^{n+1}: Az-ve\le Qu+b_0,\ z\ge 0,\ v\ge 0\right\}.$$

We remark that since $\mathcal{F}_1(u)$ is nonempty, it follows that $\mathcal{F}_2(u)$ and hence $\mathcal{F}(u)$ are also nonempty, which further implies that $\mathcal{F}_0(u)$ is also nonempty. Thus problem (30) is feasible and well-defined. The following result shows that problems (30) and (29) are equivalent.

Proposition 11. *For a given $u\in\mathcal{B}_p$, if $(z^*(u),v^*(u))$ is the optimal solution to the inner problem of (30), then $v^*(u)=0$ and $z^*(u)$ is the optimal solution to the inner problem of (29). Moreover, problems (30) and (29) have the same optimal value.*

Proposition 11 indicates that the global optimal solution of problem (29) can be obtained via solving the auxiliary problem (30) when $p=2,\infty$. Moreover, from Propositions 1 and 4, problem (30) with $p=2,\infty$ is equivalent to the following problem,

$$\begin{aligned}&\max\ \|Q^Ty\|_q+b_0^Ty\\&\text{s. t. }\ y\in\bar{\mathcal{C}}=\left\{y\in\mathbb{R}^n\mid A^Ty\le e,\ e^Ty\ge-\bar{M},\ y\le 0\right\},\end{aligned}\tag{31}$$

where $A=L^T-\operatorname{diag}(Le)$, $b_0=Ae+\hat{b}$, $q=2$ if $p=2$ and $q=1$ if $p=\infty$, and $\bar{M}>0$ is sufficiently large. It should be pointed out that the set $\bar{\mathcal{C}}$ is bounded and has nonempty interior. Thus problem (31) can be solved by global algorithms developed in the previous sections. We will report numerical results for the WCSR problem (27) in Section 7.1.

6.2. Solving the Two-Stage Adjustable Robust Optimization

As mentioned in the introduction, the WCLO appears as a subproblem in the two-stage adjustable robust optimization (TSARO) that deals with the situation when a decision maker needs to adjust his decisions after the uncertainty is revealed. The TSARO has recently become a popular topic in the optimization community due to its difficulty and wide applications (cf. Atamturk and Zhang (2007), Ben-Tal et al. (2004), Bertsimas and

Goyal (2010, 2012), Bertsimas et al. (2013), Gabrel et al. (2014), Shu and Song (2014), Zeng and Zhao (2013)). In this subsection, we study how to apply the proposed global algorithms for WCLO to solve the TSARO problem of the following form

$$(\mathrm{TSARO}_p) \qquad \min_{z\in\mathcal{Z}}\left\{d^Tz+\max_{b\in\mathcal{U}_p}\min_{x\in\mathcal{X}(z,b)}c^Tx\right\},$$

where $\mathcal{Z}:=\left\{z\in\mathbb{R}^l: Bz\le\rho,\ z\ge 0\right\}$, $\mathcal{U}_p:=\{b=Qu+b_0:\|u\|_p\le 1,\ u\in\mathbb{R}^r\}$ with $1\le p\le\infty$ denotes the uncertainty set based on the ℓ_p-norm, $\mathcal{X}(z,b):=\{x\in\mathbb{R}^n: Ax+Dz\le b,\ x\ge 0\}$. Here, $B\in\mathbb{R}^{m_1\times l}$, $\rho\in\mathbb{R}^{m_1}$, $Q\in\mathbb{R}^{m\times r}$ for some $r\le m$, $b_0\in\mathbb{R}^m$, $A\in\mathbb{R}^{m\times n}$, $D\in\mathbb{R}^{m\times l}$. We assume that $\mathcal{Z}$ is bounded and $\mathcal{X}(z,b)$ is nonempty for any (z,b). It is worth mentioning that the recourse matrix A of the recourse decision x is independent of the uncertainty u. This is referred to as fixed recourse in the stochastic programming and makes the problem relatively easier than those with non-fixed recourse. Several cutting plane based methods have been developed for solving the TSARO problem with a general polyhedral uncertainty set (cf. Bertsimas et al. (2013), Gabrel et al. (2014), Jiang et al. (2012)). Zeng and Zhao (2013) proposed a column-and-constraint generation (C&CG) method to solve it.

We next discuss how to combine our algorithms for WCLO with the cutting plane algorithm to develop new global solvers for (TSARO_p) with $p=2,\infty$. For such a purpose, we first reformulate (TSARO_p) with $p\in(1,\infty]$ as a bi-level optimization problem whose subproblem involves the l_q-norm maximization. From Propositions 1 and 4, we can obtain the following result.

Proposition 12. (TSARO_p) *with* $p\in(1,\infty]$ *has the same optimal value with the following two-level optimization problem*

$$\min_{z\in\mathcal{Z}}\left\{d^Tz+\max_{y\in\mathcal{C}}\left\{\|Q^Ty\|_q+(b_0-Dz)^Ty\right\}\right\}, \tag{32}$$

where $q>1$, $\frac{1}{p}+\frac{1}{q}=1$ *if* $p\in(1,\infty)$ *and* $q=1$ *if* $p=\infty$, *and the set* $\mathcal{C}$ *is given in (2).*

It is worth mentioning that in problem (32), there is no any joint constraint on the here-and-now decision z and the optimal wait-and-see decision y^* that solves the inner maximization problem. Such a feature leads problem (32) to only a special case of the general bi-level optimization problem that is harder to solved.

In the sequence we propose two cutting plane algorithms for (TSARO_p) with $p=2,\infty$. To start, we assume that both Assumptions 1 and 2 hold. Assumption 1 ensures the existence of optimal solutions of the inner problem of (32):

$$\psi_q(z) := \max_{y\in\mathcal{C}} \left\{ \|Q^T y\|_q + (b_0 - Dz)^T y \right\}. \tag{33}$$

It is clear that $\psi_q(z) < \infty$ due to Assumption 1. We have

PROPOSITION 13. *$\psi_q(z)$ is a piecewise linear convex function on $\mathbb{R}^l$. Let y_z be a globally optimal solution of problem (33), then $-D^T y_z$ is a subgradient of $\psi_q(z)$ at z.*

By Propositions 12 and 13, we see that (TSARO_p) with $p\in(1,\infty]$ can be reformulated equivalently as the following convex problem

$$\min_{z\in\mathcal{Z}} \left\{ d^T z + \psi_q(z) \right\}, \tag{34}$$

where $q>1$, $\frac{1}{p}+\frac{1}{q}=1$ if $p\in(1,\infty)$ and $q=1$ if $p=\infty$.

Based on the reformulated problem (34), by following a similar idea in Kelley (1960), we present the following cutting plane (CP) algorithm in a bi-level optimization framework for solving problem (32).

ALGORITHM 5 (**The CP Algorithm**).

Input: $d, c, A, D, Q, b_0, B, \rho$, and stopping criteria $\epsilon > 0$.

Output: an ϵ-optimal solution z^k.

Step 0 (**Initialization**)

(i) Solve the LO problem $\max_{z\in\mathcal{Z}} d^T z$ to get the optimal solution z^0.

(ii) Solve problem (33) with $z=z^0$ to obtain the optimal solution y^1 and the optimal value $\psi_q(z^0)$. Set the upper bound $v_0 = d^T z^0 + \psi_q(z^0)$, and $k=1$.

Step 1 Solve the following master problem:

$$\begin{aligned} \min_{z,\eta} \quad & d^T z + \eta \\ \text{s. t.} \quad & \eta \ge \|Q^T y^i\|_q + (b_0 - Dz)^T y^i, \quad i=1,\dots,k, \\ & z\in\mathcal{Z}, \end{aligned} \tag{35}$$

Derive an optimal solution (z^k, η_k) and the optimal value $f_k = d^T z^k + \eta_k$.

Step 2 Solve subproblem (33) with $z=z^k$ to obtain the optimal solution y^{k+1} and the optimal value $\psi_q(z^k)$. Update the upper bound $v_k = \min\{v_{k-1}, d^T z^k + \psi_q(z^k)\}$.

Step 3 If $v_k - f_k \leq \epsilon$, then z^k is an ϵ-optimal solution to problem (32), stop. Otherwise, set $k = k+1$, go to Step 1.

It should be pointed out that at Steps 0-(ii) and 2, subproblem (33) with $z = z^k$ can be solved by Algorithm 2 when $q = 2$ and solved by Algorithm 4 when $q = 1$. Note that $d^T z^k + \psi_q(z^k)$ provides a upper bound and $d^T z^k + \eta_k$ provides a lower bound to the optimal value of problem (34). Note that since $\{z^k\} \subseteq \mathcal{Z}$ and $\mathcal{Z}$ is bounded, there exists at least one accumulation point for the sequence $\{z^k\}$ generated by Algorithm 5.

We next present a technical result for the sequence $\{(z^k, \eta^k)\}$ generated by Algorithm 5. Let $\mathcal{R} := \{(z, \eta) : \eta \geq \psi_q(z),\ z \in \mathcal{Z}\}$, and let $\mathcal{P}_k$ denote the set of feasible solutions of problem (35). We have

LEMMA 8. *$\mathcal{R} \subseteq \mathcal{P}_k$ for all k.*

Based on Lemma 8, we can establish the global convergence of Algorithm 5 with $\epsilon = 0$.

THEOREM 7. *Let $\{z^k\}$ be an infinite sequence generated by Algorithm 5 with $\epsilon = 0$. Then any accumulation point of $\{z^k\}$ is an optimal solution of problem (32).*

Next, by following a similar procedure in Zeng and Zhao (2013), we present an improved cutting plane (ICP) algorithm for solving problem (32) as follows.

ALGORITHM 6 (**The ICP Algorithm**).

The algorithm is identical to Algorithm 5 except that Step 1 is replaced by the following step:

Step $1'$ If $q \in (1, \infty)$, compute $u_q^k = \mu(y^k; q)$ by (1). If $q = 1$, compute u_q^k by (4) with $y^* = y^k$. Solve the following master problem:

$$\begin{aligned} \min_{z,\eta,x^1,\dots,x^k} \quad & d^T z + \eta \\ \text{s. t.} \quad & \eta \geq c^T x^i, \quad i = 1, \dots, k, \\ & Ax^i + Dz \leq Qu_q^i + b_0, \quad i = 1, \dots, k, \\ & x^i \geq 0, \quad i = 1, \dots, k, \\ & z \in \mathcal{Z}. \end{aligned} \tag{36}$$

Derive an optimal solution $(z^k, \eta_k, \bar{x}^1, \dots, \bar{x}^k)$ and the optimal value $f_k = d^T z^k + \eta_k$.

Compared with Algorithm 5, Algorithm 6 solves a master problem with a polynomial number of variables and constraints in each iteration. However, this master problem provides a tighter lower bound to problem (32) than one in Algorithm 5 (see Proposition 14 below). Furthermore, compared with the C&CG algorithm in Zeng and Zhao (2013), Algorithm 6 in its Step 2 solves an ℓ_q-norm maximization subproblem, while the C&CG algorithm solves a subproblem that is a 0-1 mixed integer program with a large number of variables and constraints, which generally requires more computational time. As we see later, Algorithm 6 is computationally more effective than both Algorithm 5 and the C&CG algorithm.

We next present a technical result for the sequence $\{(z^k, \eta^k)\}$ generated by Algorithm 6.

LEMMA 9. *Let the sequence* $\{(z^k, \eta^k)\}$ *be generated by Algorithm 6, and let* f_p^* *be the optimal value of* (TSARO_p). *Then* $(z^k, \eta^k) \in \mathcal{P}_k$ *and* $f_p^* \geq f_k = d^T z^k + \eta_k$ *for all* k.

From Lemma 9, we see that the optimal solution of problem (36) is also feasible to problem (35). We immediately have the following result.

PROPOSITION 14. *Let* f_k *and* f_k' *denote the optimal value of problems (36) and (35), respectively. Let* f_p^* *be the optimal value of problem (32). Then* $f_p^* \geq f_k \geq f_k'$.

Proposition 14 indicates that problem (36) provides a tighter lower bound to problem (32) than problem (35). This result indicates that Algorithm 6 requires less the number of iterations than Algorithm 5.

Using Lemma 9, we now establish the global convergence of Algorithm 6 with $\epsilon = 0$.

THEOREM 8. *Let* $\{z^k\}$ *be an infinite sequence generated by Algorithm 6 with* $\epsilon = 0$. *Then any accumulation point of* $\{z^k\}$ *is an optimal solution of problem (32).*

6.3. Application to Two-stage Robust Location-Transportation Problem

In this subsection, we present an application of TSARO, i.e., the two-stage robust location-transportation problem in Gabrel et al. (2014), Zeng and Zhao (2013).

Let us consider the following location-transportation problem with m potential facilities and n customers. Let ξ_i be the fixed cost of the building facilities at site i and ζ_i be the unit capacity cost for $i = 1, \ldots, m$. Let d_j be the demand from customer j for $j = 1, \ldots, n$, and the unit transportation cost from facility i to customer j be c_{ij}. Let the maximal allowable capacity of the facility at site i be η_i and $\sum_i \eta_i \geq \sum_j d_j$ ensures feasibility of the problem. Let $y \in \{0,1\}^m$ be the facility location variable, $z \in \mathbb{R}^m_+$ be the capacity variable, and $X = (x_{ij}) \in \mathbb{R}^{m\times n}_+$ be the transportation variable. Then the location-transportation problem can be defined as the following binary linear optimization problem:

$$\begin{aligned} &\min_{y,z,X} \ \xi^T y + \zeta^T z + \mathrm{Tr}(CX^T) \\ &\text{s. t. } z \le \mathrm{diag}(\eta) y, \quad X\bar{e} \le z, \quad X^T \hat{e} \ge d, \\ &\qquad y \in \{0,1\}^m, \quad z \in \mathbb{R}^m_+, \quad X \in \mathbb{R}^{m\times n}_+. \end{aligned} \tag{37}$$

where $\xi = (\xi_1, \dots, \xi_m)^T$, $\zeta = (\zeta_1, \dots, \zeta_n)^T$, $C = (c_{ij})_{m\times n}$, $d = (d_1, \dots, d_n)^T$, $\eta = (\eta_1, \dots, \eta_m)^T$, $\bar{e} \in \mathbb{R}^n$ and $\hat{e} \in \mathbb{R}^m$ are the column vector of all ones.

In practice, however, the demand is unknown before any facility is built and the capacity is installed. To account for the uncertain demand, Gabrel et al. (2014), Zeng and Zhao (2013) proposed the following worst-case robust location-transportation (WCRLT) problem:

$$\begin{aligned} &\min_{y,z} \left\{ \xi^T y + \zeta^T z + \max_{d\in\mathcal{D}_p} \min_{X\in\mathcal{S}(d,z)} \mathrm{Tr}(CX^T) \right\} \\ &\text{s. t. } z \le \mathrm{diag}(\eta) y, \quad y \in \{0,1\}^m, \quad z \in \mathbb{R}^m_+, \end{aligned} \tag{38}$$

where $\mathcal{D}_p := \{d = Qu + d_0 : \|u\|_p \le 1, \ u \in \mathbb{R}^r\}$ with $p = 1, 2, \infty$ denotes the uncertainty set based on the ℓ_p-norm, $\mathcal{S}(d,z) := \left\{X \in \mathbb{R}^{m\times n}_+ : X\bar{e} \le z, \ X^T\hat{e} \ge d\right\}$. Here, $Q \in \mathbb{R}^{n\times r}$ for some $r \le n$, $d_0 \in \mathbb{R}^n$. Gabrel et al. (2014), Zeng and Zhao (2013) dealt with the polyhedral uncertainty set $\mathcal{D} := \left\{d = Qu + d_0 : e^T u \le \Gamma, \ u \in [0,1]^n\right\}$, where $\Gamma > 0$ is an integer and Q is an $n \times n$ diagonal matrix. It is easy to see that problem (38) is a special case of TSARO.

We denote by A the coefficient matrix of constraints $X\bar{e} \le z$ and $X^T\hat{e} \ge d$. Note that the rank of A is at most $m + n - 1$ and thus the associated set $\mathcal{C}$ is unbounded for problem (38). Thus, the global methods developed in Subsection 6.2 can not be used directly to solve problem (38) with $p = 2, \infty$. To overcome this difficulty, using the same technique as in constructing auxiliary problem (30), we can construct an auxiliary problem that is equivalent to problem (38). Then the constructed auxiliary problem can be solved by global algorithms developed in Subsection 6.2. We will report numerical results for the WCRLT problem (38) in Section 7.2.

7. Numerical Experiments

In this section, we present computational results of the SCOBB algorithm (Algorithm 2) for (WCLO$_2$) and the results of the FBB algorithm (Algorithm 4) for (WCLO$_\infty$). We also give numerical results of the CP and ICP algorithms (Algorithms 5 and 6) for (TSARO$_p$) with $p=2,\infty$. [1] The algorithms are coded in Matlab R2013b and run on a PC (3.33GHz, 8GB RAM). All the linear and convex quadratic subproblems in the algorithms are solved by the QP solver in CPLEX 12.6 with Matlab interface (cf. IBM ILOG CPLEX (2013)).

In our numerical experiments, the stopping parameter ϵ is set as $\epsilon=10^{-5}$. We use the notations described in the following table in our discussion of the computational results.

Table 1 Notations

Opt.val:	the average optimal value obtained by the algorithm for 5 test instances
Time:	the average CPU time of the algorithm (unit: seconds) for 5 test instances
Iter:	the average number of iterations in the main loop of the algorithm for 5 test problems
Val$_{\rm SCO}$:	the average objective value at the solution computed by SCO in SCOBB for 5 test problems

7.1. Numerical results of the SCOBB and FBB algorithms

In this subsection, we test the SCOBB algorithm and the FBB algorithm on the WCSR problem (27) with both synthetic data and data from the financial networks reported in the literature. Since the complexity of the SCOBB algorithm grows exponentially in terms of the column number of the matrix Q, we restrict instances with a small number of columns of Q (say, $r\le 10$).

Note that the WCSR problem (27) is equivalent to problem (31) in the sense that they have the same optimal value, due to Propositions 10 and 11. We apply the SCOBB and FBB to solve problem (31) with $q=2$ and $q=1$. The nonlinear SDR (18) associated with problem (27) with $p=2$ becomes the following problem,

$$\begin{aligned}
\max\ & b_0^T y+\sqrt{(b_0^T y)^2+\mathrm{Tr}\left((QQ^T-b_0b_0^T)Y\right)} \\
\text{s.t.}\ & A^T y\le e,\quad e^T y\ge -\bar{M},\quad y\le 0, \\
& A^T Y\ge e y^T,\quad e^T Y\le -\bar{M}y,\quad Y\ge 0, \\
& Y-yy^T\succeq 0,
\end{aligned}\tag{39}$$

where $A=L^T-\mathrm{diag}(Le)$, $b_0=Ae+\hat{b}$, and $\bar{M}>0$ is sufficiently large. In our test, the penalty parameter $\bar{M}$ in (31) and (39) is set as 10. We also present an example (see Example

[1] All the data and the codes used in Section 7 can be downloaded from `http://www.sdspeople.fudan.edu.cn/jiangrujun/papers/Linear_time_GTRS.pdf`

1.1 in the online supplement) showing that the NLSDP problem (39) may not necessarily find the globally optimal solution of the WCSR problem (27) with $p=2$.

We compare SCOBB with the NLSDP problem (39) and the global optimization package BARON (cf. Sahinidis (1996)) for small-scale random instances of the WCSR problem (27) with $p=2$. In our experiments, the NLSDP problem (39) is solved by the bisection search algorithm (BSA) in polynomial time in Peng and Tao (2015), and the SDP subproblems in BSA are solved by the SDP solver SDPT3 (cf. Toh et al. (1999)).

We also compare both SCOBB and FBB with the MIPR (mixed integer program reformulation) approach proposed in Zeng and Zhao (2013) for the WCSR problem (27) with $p=2,\infty$, where the MIPR for WCSR is given in Section 2 in the online supplement. In our experiments, the 0-1 mixed integer problem in MIPR is solved by the MIQP solver in CPLEX 12.6. The parameter 'TolXInteger' in solver which controls the precision of integer variables is set as default value 10^{-7}. The penalty parameter M in 0-1 mixed integer problem is set as 10^5.

7.1.1. Numerical results for randomly generated test problems. In this test, the data $(L,Q,\hat{b})$ in the WCSR model (27) are randomly generated in the same way as in Peng and Tao (2015). That is, the off-diagonal entries of L are drawn from $LN(0.4,1)$ (i.e., lognormal distribution with mean 0.4 and standard deviation 1), entries of Q are drawn from $U(-3,3)$ (i.e., uniformly distributed within interval $[-3,3]$) and entries of $\hat{b}$ are drawn from $U(0.5,5)$.

The average numerical results for randomly generated 5 test problems of the same size are summarized in Tables 2-7, where the following notations are also used:

- "NLSDP" denotes the nonlinear SDP relaxation (39);
- "LB" and "UB" denotes the average lower and upper bounds obtained by NLSDP for 5 test problems, respectively;
- "T" denotes the number of the instances for which NLSDP can find the global solution.

In Table 2, we compare the average performance of SCOBB, BARON, MIPR and NLSDP for WCSR with $p=2$. The sign $*$ for SCOBB represents that global optimal solutions are found by SCO for all the 5 instances. From Table 2, one can see that SCOBB is able to find the globally optimal solution for all the test problems, while BARON can only find the globally optimal solution of 16 instances out of the 50 test problems within 600 seconds. Both MIPR and NLSDP can find global optimal solutions for numerous test instances, but

Table 2 Comparison of the average performance of SCOBB, BARON, MIPR and NLSDP for WCSR with $p=2$

Size		SCOBB				BARON		MIPR		NLSDP			
n	r	Time	Opt.val	Iter	Val$_{SCO}$	Time	Opt.val	Time	Opt.val	Time	LB	UB	T
20	5	1.8	4.5363	125.0	∗4.5363	140.2	4.6117(4)	0.9	4.5363	1.9	4.5363	4.5363	5
30	5	2.8	6.0828	169.8	∗6.0828	356.2	6.2683(3)	0.8	6.0828	3.4	6.0828	6.0828	5
50	5	5.7	9.3514	209.6	∗9.3514	177.5	13.1258(1)	3.7	9.3514	29.0	9.3505	9.3514	4
80	5	9.7	12.4629	160.6	∗12.4629	394.0	12.4629(5)	14.5	12.4629	62.8	12.4629	12.4629	5
100	5	11.6	17.6994	122.0	∗17.6994	307.3	18.9620(1)	21.0	17.6994	211.5	17.6994	17.6994	5
20	10	27.1	3.4492	1922.6	∗3.4492	–	–	3.6	3.7892(4)	2.9	3.4461	3.4493	4
30	10	21.4	5.6409	1403.4	∗5.6409	–	–	18.4	5.6409	2.6	5.6409	5.6409	5
50	10	53.1	8.5300	2105.8	∗8.5300	–	–	197.9	8.5300	21.0	8.5300	8.5300	5
80	10	70.9	12.8140	1482.0	∗12.8140	–	–	317.4	13.0233(3)	195.9	12.8137	12.8140	4
100	10	73.3	16.4826	996.2	∗16.4826	–	–	370.2	17.7078(2)	206.2	16.4826	16.4826	5

Remark: The number in parentheses stands for the number of the instances for which BARON or MIPR can verify the global optimality of the solution within 600 seconds. Time and Opt.val for BARON or MIPR denote the average CPU time and optimal value for the instances that are globally solved by BARON or MIPR in 5 instances, respectively. The sign "–" stands for the situations where the method failed to find the global solution within 600 seconds in all cases.

Table 3 Numerical results of SCOBB and NLSDP for three specifical instances of WCSR with $p=2$

Instance			SCOBB			NLSDP		
ID	n	r	Time	Opt.val	Iter	Time	LB	UB
1	50	5	4.2	9.815921	162	66.7	9.811750	9.816183
2	20	10	75.9	2.089064	4913	7.8	2.073435	2.089810
3	80	10	113.0	12.446227	2425	466.2	12.444812	12.446284

NLSDP fails to globally solve three specific instances listed in Table 3. Moreover, BARON usually requires more CPU time than SCOBB for the solved instances. Also, for numerous test instances, BARON only reported the best solution obtained within 600 seconds and failed to verify the global optimality of the obtained solution. We observe that SCOBB is more effective than MIPR in terms of CPU time for all the instances with $n > 50, r = 5$ and $n \geq 50, r = 10$. We also observe that the CPU time of SCOBB grows rapidly in terms of the column number r of matrix Q, while the CPU time for BARON, MIPR and NLSDP increases very fast as the size n of the test problem grows. The numerical results also show that SCO often obtains a solution with good quality from the fact that the computed solutions are global optimal solutions for all the instances with $r = 5, 10$.

Table 4 summarizes the average numerical results of SCOBB for medium and large scale instances of WCSR with $p=2$, where x^* is the global optimal solution found by the algorithm. As one can see from Table 4 that for all the test problems of WCSR with $p=2$, SCOBB can effectively find the globally optimal solution within 2700 seconds. It should be also pointed out that the SCO algorithm can often find globally optimal solutions for all the instances with $r = 5$ and $r = 10$.

Table 4 Average numerical results of SCOBB for WCSR with $p=2$

Size		SCOBB				
n	r	Time	Opt.val	Iter	$\mathrm{Val}_{\mathrm{SCO}}$	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$
200	5	121.0	27.526448	112.4	$*$27.526448	0.8624
300	5	168.6	38.731214	144.6	$*$38.731214	0.8709
400	5	269.8	44.957784	132.0	$*$44.957784	0.8876
500	5	482.0	53.499306	148.6	$*$53.499306	0.8930
200	10	654.6	28.473318	989.4	$*$28.473318	0.8576
300	10	1094.2	35.837753	1407.0	$*$35.837753	0.8805
400	10	1778.5	47.939705	1419.8	$*$47.939705	0.8802
500	10	2460.5	54.067573	1155.6	$*$54.067573	0.8917

Table 5 Average numerical results of Algorithm 4 and MIPR for WCSR with $p=\infty$

Size		Algorithm 4				MIPR	
n	r	Time	Opt.val	Iter	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$	Time	Opt.val
30	5	0.0574	7.564067	19.6	0.7479	0.6310	7.564067
50	5	0.0947	10.553717	21.6	0.7889	5.2957	10.553717
80	5	0.2449	13.701750	22.2	0.8287	24.7190	13.701750
100	5	0.3149	19.184807	18.0	0.8082	21.6093	19.184807
150	5	1.0158	25.409655	18.2	0.8306	188.0417	25.409655
30	10	0.8409	9.944164	253.0	0.6685	7.0121	9.944164
50	10	1.3870	12.602065	280.8	0.7480	101.3447	12.602065
80	10	4.9948	15.649310	423.8	0.8044	165.0869	16.307455(2)
100	10	4.8578	19.725000	287.2	0.8027	245.3200	20.200269(1)
150	10	16.1125	23.323694	408.2	0.8445	−	−

Remark: The number in parentheses stands for the number of the instances for which MIPR cannot verify the global optimality of the solution within 600 seconds. Time and Opt.val for MIPR denote the average CPU time and optimal value for the instances that are globally solved by MIPR in 5 instances. The sign "−" denotes the situations where the method failed to find the global solution within 600 seconds in all cases.

In Table 5, we compare the average performance of Algorithm 4 and MIPR for 5 random instances of WCSR with $p=\infty$. We can see that Algorithm 4 can find the global optimal solution for all test problems. Moreover, Algorithm 4 outperforms MIPR in terms of the CPU time for the solved instances with $r=5,10$.

Table 6 summarizes the average numerical results of Algorithm 4 for medium and large scale instances of WCSR with $p=\infty$, where x^* is the global optimal solution found by the algorithm. From Table 6, we see that Algorithm 4 can effectively find the globally optimal solution for all the test instances of WCSR with $p=\infty$ within 400 seconds.

For convenience, we denote by Algorithm A (Algorithm B, respectively) the method where the optimal value of (WCLO_1) is derived by solving the $2r$ linear programs in (5) according to Proposition 6 (solving a linear program (6), respectively). Table 7 summarizes the average numerical results of Algorithms A and B for medium and large scale instances

Table 6 Average numerical results of Algorithm 4 for WCSR with $p=\infty$

Size		Algorithm 4				Size		Algorithm 4			
n	r	Time	Opt.val	Iter	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$	n	r	Time	Opt.val	Iter	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$
200	5	1.8	28.699626	21.8	0.8565	200	10	21.4	31.934539	277.4	0.8403
300	5	5.3	39.641063	21.0	0.8679	300	10	104.6	38.075087	499.0	0.8731
400	5	10.1	45.904413	19.2	0.8852	400	10	97.5	51.233594	214.2	0.8719
500	5	19.5	54.703325	19.4	0.8906	500	10	226.0	57.002375	254.8	0.8860

Table 7 Average numerical results of Algorithms A and B for WCSR with $p=1$

Size		Algorithm A			Algorithm B		
n	r	Time	Opt.val	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$	Time	Opt.val	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$
200	5	0.5	27.278095	0.8636	0.8	27.278095	0.8636
300	5	1.0	38.455844	0.8718	3.6	38.455844	0.8718
400	5	2.4	44.707294	0.8882	8.4	44.707294	0.8882
500	5	4.4	53.228812	0.8935	18.2	53.228812	0.8935
200	10	0.6	27.874673	0.8606	2.5	27.874673	0.8606
300	10	1.9	35.402705	0.8820	8.5	35.402705	0.8820
400	10	4.8	47.286932	0.8818	22.2	47.286932	0.8818
500	10	8.5	53.377167	0.8933	44.7	53.377167	0.8933

of WCSR with $p=1$. We observe that Algorithm A is more effective than Algorithm B in terms of CPU time for medium and large scale problems. This is because, compared to linear program (5), which has $n+m$ variables and constraints, linear program (6) has $\mathcal{O}(r(m+n))$ variables and constraints and hence is much slower to solve.

7.1.2. Numerical results for test problems with partial real data. Capponi et al. (2016) considered the system consisting of the banking sectors in eight European countries for seven years, starting from 2008 and ending in 2014. These countries are well representative of interbank activities in the European market as their liabilities account for 80% of the total liabilities of the European banking sector. Banks consolidated foreign claims data from the BIS (Bank for International Settlements) Quarterly Review are summarized in Table 8 (also see Table 1 in Capponi et al. (2016)).

In this test, the data L in the WCSR model (27) are from the interbank liability matrix in Table 8, while the data $(Q, \hat{b})$ are randomly generated. That is, entries of Q are drawn from $U(-15,15)$ and entries of $\hat{b}$ are drawn from $U(50,200)$. Table 9 summarizes the average numerical results of SCOBB, Algorithm 4 and Algorithm A for 5 random instances of WCSR with partial real data for $p=1,2,\infty$, where x^* is the global optimal solution found by the algorithm. From Table 9, we see that for all the test instances, SCOBB can effectively find the global optimal solution within about one second, while both Algorithm 4 and Algorithm A can find the global optimal solution in less time.

Table 8 Banks' consolidated foreign claims (in USD billion)

December 2009	United Kingdom	Germany	France	Spain	Netherland	Ireland	Belgium	Portugal
United Kingdom	0.00	500.62	341.62	409.36	189.95	231.97	36.22	10.43
Germany	172.97	0.00	292.94	51.02	176.58	36.35	20.52	4.62
France	239.17	195.64	0.00	50.42	92.73	20.60	32.57	8.08
Spain	114.14	237.98	219.64	0.00	119.73	30.23	26.56	28.08
Netherland	96.69	155.65	150.57	22.82	0.00	15.47	28.11	11.39
Ireland	187.51	183.76	60.33	15.66	30.82	0.00	64.50	21.52
Belgium	30.72	40.68	301.37	9.42	131.55	6.11	0.00	1.17
Portugal	24.26	47.38	44.74	86.08	12.41	5.43	3.14	0.00
June 2010	United Kingdom	Germany	France	Spain	Netherland	Ireland	Belgium	Portugal
United Kingdom	0.00	462.07	327.72	386.37	135.37	208.97	43.14	7.72
Germany	172.18	0.00	255.00	39.08	149.82	32.11	20.93	3.93
France	257.11	196.84	0.00	26.26	80.84	18.11	29.70	8.21
Spain	110.85	181.65	162.44	0.00	72.67	25.34	18.75	23.09
Netherland	141.39	148.62	126.38	20.66	0.00	12.45	23.14	11.11
Ireland	148.51	138.57	50.08	13.98	21.20	0.00	53.99	19.38
Belgium	29.15	35.14	253.13	5.67	108.68	5.32	0.00	0.39
Portugal	22.39	37.24	41.90	78.29	5.13	5.15	2.57	0.00

Data source. BIS Quarterly Review Table 9B. The ijth entry of each matrix denotes the interbank liabilities from the banking sector of country i to the banking sector of country j.

Table 9 The average performance of SCOBB, Algorithm 4 and Algorithm A for WCSR with partial real data

	SCOBB ($p=2$)			Algorithm 4 ($p=\infty$)			Algorithm A ($p=1$)		
	Opt.val	Time	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$	Opt.val	Time	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$	Opt.val	Time	$\frac{1}{n}\sum_{i=1}^{n} x_i^*$
Dec. 2009	2.007399	0.881	0.749	2.141554	0.015	0.732	1.969954	0.004	0.754
Jun. 2010	1.243539	1.023	0.845	1.419406	0.015	0.823	1.189241	0.005	0.851

7.2. Numerical results of the CP and ICP algorithms

In this subsection, we present numerical results of the CP and ICP algorithm (Algorithms 5 and 6) for the TSARO problem. We test Algorithms 5 and 6 on the two-stage robust location-transportation problem (38) with $p=2,\infty$.

To test the performance of Algorithms 5 and 6, we randomly generate the parameters in the test problems in the same fashion as in Gabrel et al. (2014), Zeng and Zhao (2013). The entries of ξ are drawn from $U[100,1000]$, entries of ζ are drawn from $U[10,100]$, entries of C are drawn from $U[1,1000]$, entries of η are drawn from $U[200,700]$, entries of Q are drawn from $U[-100,100]$, and entries of d_0 are drawn from $U[10,500]$. The relative gap is computed by using the formula

$$\text{gap} := (UB - LB)/\max\{1, |LB|\},$$

where LB and UB denotes the lower bound and upper bound found by the algorithm.

We compare the Algorithms 5 and 6 with the C&CG algorithm in Zeng and Zhao (2013) for random instances of problem (38) with $p=2,\infty$. The C&CG algorithm is coded in

Matlab R2013b. The subproblem in C&CG, which is a 0-1 mixed integer program (0-1MIP), is solved by the MIQP solver in CPLEX 12.6. The penalty parameter M in 0-1MIP is set as 10^3.

In Tables 10 and 11, we summarize the average numerical results of Algorithm 5, Algorithm 6 and C&CG for 5 instances of problem (38) with $p=2$ and $p=\infty$, respectively. We see from Tables 10 and 11 that for all the solved instances, Algorithm 6 is more effective than both Algorithm 5 and C&CG in terms of gap and CPU time. We also see that Algorithm 6 can find the global optimal solution in a small number of iterations and requires much less the number of iterations than Algorithm 5. In comparison with C&CG, Algorithm 6 can find a global solution in less CPU time for numerous instances of TSRLTP with $p=2,\infty$. This indicates that Algorithm 6 is promising for real applications in transportation systems.

From Proposition 6, we observe that (TSARO_1) has the same optimal value with the following LO problem

$$\begin{aligned}
\min_{z,\xi,x^1,\ldots,x^{2r}} \quad & d^T z+\xi \\
\text{s.t.} \quad & \xi \geq c^T x^i, \quad i=1,\ldots,2r, \\
& Ax^i + Dz \leq Qe_i + b_0, \quad i=1,\ldots,2r, \\
& x^i \geq 0, \quad i=1,\ldots,2r, \\
& z \in \mathcal{Z}.
\end{aligned} \tag{40}$$

We denote by Algorithm C the method where the optimal value of (TSARO_1) is derived by solving the LO problem (40). Table 12 summarizes the average numerical results of Algorithm C for random 5 instances of TSRLTP with $p=1$.

8. Conclusions

In this paper, we have considered the WCLO problem with the uncertainty set $\mathcal{U}_p$ ($1 \leq p \leq \infty$) (denoted by (WCLO_p)) that arises from numerous important applications. We prove that (WCLO_p) is strongly NP-hard for $p \in (1,\infty)$ and (WCLO_∞) is NP-hard. By combining several simple optimization approaches such as the SCO, the B&B framework and initialization technique, we have developed the SCOBB algorithm to find a global optimal solution to (WCLO_2). We have established the global convergence of the SCOBB algorithm and estimated its complexity. Preliminary numerical experiments demonstrated

Table 10 Average numerical results of Algorithm 5, Algorithm 6 and C&CG for TSRLTP with $p = 2$

Size			Algorithm 6			Algorithm 5			C&CG		
m	n	r	Gap	Time	Iter	Gap	Time	Iter	Gap	Time	Iter
3	3	2	1.4e-007(5)	1.987	3.2	2.6e-007(5)	1.514	4.8	1.3e-008(5)	0.753	3.2
5	5	3	3.1e-007(5)	4.780	3.8	3.0e-007(5)	4.536	7.4	3.5e-008(5)	1.312	3.8
7	7	4	1.3e-007(5)	13.933	4.6	1.3e-007(5)	32.560	14.0	3.7e-008(5)	4.399	4.6
10	10	5	1.5e-007(5)	126.783	7.6	3.3e-008(5)	395.347	42.8	1.8e-008(5)	89.200	8.0
15	15	5	3.1e-004(2)	601.532	8.4	2.5e-001(0)	612.796	23.8	–	–	–
20	20	5	2.1e-003(0)	786.662	4.0	1.6e-001(0)	635.536	9.6	–	–	–

Remark: The number in parentheses stands for the number of the instances for which the absolute gap obtained by the algorithm is less than $\epsilon = 10^{-5}$ within 600 seconds. The sign "–" stands for the situations where the first iteration of the algorithm is not terminated within 600 seconds in all cases.

Table 11 Average numerical results of Algorithm 5, Algorithm 6 and C&CG for TSRLTP with $p = \infty$

Size			Algorithm 6			Algorithm 5			C&CG		
m	n	r	Gap	Time	Iter	Gap	Time	Iter	Gap	Time	Iter
3	3	2	5.1e-017(5)	0.198	1.4	3.0e-016(5)	0.124	4.4	-2.5e-011(5)	0.371	1.4
5	5	3	1.2e-016(5)	0.229	1.6	1.2e-016(5)	0.229	7.2	-9.2e-017(5)	0.678	1.6
7	7	4	8.6e-017(5)	0.283	1.8	2.4e-016(5)	0.444	10.6	8.6e-017(5)	2.153	1.8
10	10	5	-5.0e-016(5)	0.508	2.6	1.7e-016(5)	1.707	21.4	9.8e-017(5)	22.713	3.2
15	15	5	1.1e-018(5)	0.639	2.8	1.4e-008(5)	7.062	62.6	4.2e-001(0)	769.124	1.0
20	20	5	8.1e-017(5)	0.865	3.0	5.2e-016(5)	10.750	71.2	–	–	–
30	30	5	2.0e-016(5)	2.068	4.0	1.4e-007(5)	50.174	141.2	–	–	–
50	50	5	1.1e-016(5)	6.455	3.6	1.4e-002(1)	546.924	249.8	–	–	–
100	100	5	4.6e-008(5)	22.838	3.2	5.6e-002(0)	605.861	106.4	–	–	–
20	20	10	1.5e-016(5)	24.723	3.8	7.7e-014(5)	169.490	31.8	–	–	–
30	30	10	1.4e-016(5)	57.245	4.4	2.5e-002(0)	604.890	57.2	–	–	–
50	50	10	1.8e-016(5)	182.154	3.6	1.0e-001(0)	616.701	14.6	–	–	–
100	100	10	3.4e-004(3)	599.337	3.8	1.5e-001(0)	629.574	4.2	–	–	–

Remark: The number in parentheses stands for the number of the instances for which the absolute gap obtained by the algorithm is less than $\epsilon = 10^{-5}$ within 600 seconds. The sign "–" stands for the situations where the first iteration of the method is not terminated within 600 seconds in all cases.

Table 12 Average numerical results of Algorithm C for TSRLTP with $p = 1$

Size			Algorithm C		Size			Algorithm C	
m	n	r	Opt.val	Time	m	n	r	Opt.val	Time
20	20	5	2.830245	0.104	20	20	10	2.281078	0.124
30	30	5	3.967852	0.142	30	30	10	2.646612	0.199
50	50	5	4.543949	0.294	50	50	10	5.383468	0.492
100	100	5	10.460254	1.218	100	100	10	9.426302	2.484

that the SCOBB algorithm can effectively find a global optimal solution to ($WCLO_2$) when the involved matrix Q in the underlying problem has only a few columns. We have proposed a hybrid algorithm, which combines the SCO approach with the nonlinear SDR and disjunctive cut techniques, to find a strong bound for generic ($WCLO_2$). Numerical experiments illustrate that the obtained bounds are stronger than what have been reported in the literature. We have also integrated LO relaxation and complementarity branching techniques to develop a finite branch-and-bound algorithm to solve globally (WCLO$_\infty$) and evaluated its performance. Finally, we have integrated the proposed global algorithms for WCLO with cutting plane approaches to develop effective algorithms to find a globally optimal solution for (TSARO$_p$) with $p = 2, \infty$. A future research topic is to investigate whether we can develop effective global algorithms for generic (WCLO$_p$) and (TSARO$_p$).

Supplemental Material

Supplemental material to this paper is available at http://dx.doi.org/10.1287/ijoc.?.?.

Acknowledgments

The research of the first two authors is partially supported by NSFC grants 11871433 and 11371324, and Zhejiang Provincial NSFC grants LY17A010023 and LY18A010011. The research of the third author is supported by NSF grants CMMI-1131690, CMMI-1537712 and CNS-1637347. The research of the last two authors is supported by Hong Kong Research Grants Council grants 14213716 and 14202017.

References

Alarie, S., C. Audet, B. Jaumard, G. Savard. 2001. Concavity cuts for disjoint bilinear programming. *Math. Program.* **90(2)** 373–398.

Al-Khayyal, F., J. E. Falk. 1983. Jointly constrained biconvex programming. *Math. Oper. Res.* **8** 273–286.

Anstreicher, K. 2009. Semidefinite programming versus the reformulation linearization technique for non-convex quadratically constrained quadratic programming. *J. Glob. Optim.* **43** 471–484.

Atamturk, A., M. Zhang. 2007. Two-stage robust network flow and design under demand uncertainty. *Oper. Res.* **55(4)** 662–673.

Audet, C., P. Hansen, B. Jaumard, G. Savard. 1999. A symmetrical linear maxmin approach to disjoint bilinear programming. *Math. Program.* **85(3)** 573–592.

Ben-Tal, A. 1998. Robust convex optimization. *Math. Oper. Res.* **23(4)** 769–805.

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

Ben-Tal, A., A. Goryashko, E. Guslitzer. 2004. Adjustable robust solutions of uncertain linear programs. *Math. Program.* **99(2)** 351–376.

Ben-Tal, A., A. Nemirovski. 1999. Robust solutions of uncertain linear programs. *Oper. Res. Lett.* **25(1)** 1–14.

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

Bertsimas, D., V. Goyal. 2010. On the power of robust solutions in two-stage stochastic and adaptive optimization problems. *Math. Oper. Res.* **35(2)** 284–305.

Bertsimas, D., V. Goyal. 2012. On the power and limitations of affine policies in two-stage adaptive optimization. *Math. Program.* **34(2)** 491–531.

Bertsimas, D., E. Litvinov, X. A. Sun, J. Y. Zhao, T. X. Zheng. 2013. Adaptive robust optimization for the security constrained unit commitment problem. *IEEE Trans. Power Syst.* **28(1)** 52–63.

Birge, J., F. Louveaux. 1997. *Introduction to Stochastic Programming*. Springer, Berlin.

Burer, S., D. Vandenbussche. 2008. A finite branch-and-bound algorithm for nonconvex quadratic programming via semidefinite relaxations. *Math. Program.* **113(2)** 259–282.

Burer, S., D. Vandenbussche. 2009. Globally solving box-constrained nonconvex quadratic programs with semidefinite-based finite branch-and-bound. *Comput. Optim. Appl.* **43(2)** 181–195.

Capponi, A., P. Chen, D. Yao. 2016. Liability concentration and systemic losses in financial networks. *Oper. Res.* **64(5)** 1121–1134.

Chen, J., S. Burer. 2012. Globally solving nonconvex quadratic programming problems via completely positive programming. *Math. Program. Comput.* **4** 33–52.

Ding, X., F. Al-Khayyal. 2007. Accelerating convergence of cutting plane algorithms for disjoint bilinear programming *J. Glob. Optim.* **38** 421–436.

Eisenberg, L., T. Noe. 2001. Systemic risk in financial systems. *Manag. Sci.* **47(2)** 236–249.

Floudas, C., V. Visweswaran. 1994. Quadratic optimization. *Handbook of Global Optimization, R.Horst and P.M. Pardalos, Eds., Kluwer Academic Publishers* 217–270.

Gabrel, V., M. Lacroix, C. Murat, N. Remli. 2014. Robust location transportation problems under uncertain demands. *Discrete Appl. Math.*, **164** 100–111.

Ge, D., X. Jiang, Y. Ye. 2011. A note on the complexity of ℓ_p minimization. *Math. Program.* **129(2)** 285–299.

Goemans, M., D. Williamson. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM* **42(6)** 1115–1145.

Hansen, P., B. Jaumard, G. Savard. 1992. New branch and bound rules for linear bilevel programming. *SIAM J. Sci. Stat. Comput.* **13(5)** 1194–1217.

Henrion, D., J. Lasserre, J. Loefberg. 2009. Gloptipoly 3: moments, optimization and semidefinite programming. *Optim. Methods Softw.* **24(4-5)** 761–779.

IBM ILOG CPLEX. 2013. IBM ILOG CPLEX 12.6 user's manual for CPLEX .

Jiang, R., M. Zhang, G. Li, Y. Guan. Benders decomposition for the two-stage security constrained robust unit commitment problem. 62nd IIE Annual Conference and Expo 2012, pp. 3332-3341.

Kelley, J. E. 1960. The cutting-plane method for solving convex programs. *J. Soc. Indust. Appl. Math.* **8** 703–712.

Kabazian, A., J.M. Peng. 2019. Vulnerability analysis of the financial network. Forthcoming in *Management Science*, https://doi.org/10.1287/mnsc.2018.3106.

Luo, H.Z., X.D. Bai, G. Lim, J.M. Peng. 2019. New global algorithms for quadratic programming with a few negative eigenvalues based on alternative direction method and convex relaxation. *Math. Program. Comput.* **11(1)** 119–171.

Luo, H.Z., X.D. Bai, J.M. Peng. 2019. Enhancing semidefinite relaxation for quadratically constrained quadratic programming via penalty methods. *J. Optim. Appl.* **180(3)** 964–992.

Mangasarian, O. L., T. H. Shiau. 1986. A variable-complexity norm maximization problem. *SIAM J. Algebraic Discret. Methods* **7(3)** 455–461.

Nesterov, Y. 1998. Semidefinite relaxation and nonconvex quadratic optimization. *Optim. Methods Softw.* **9** 140–160.

Pardalos, P. 1991. Global optimization algorithms for linearly constrained indefinite quadratic problems. *Comput. Math. Appl.* **21** 87–97.

Peng, J.M., Z. Tao. 2015. A nonlinear semidefinite optimization relaxation for the worst-case linear optimization under uncertainties. *Math. Program.* **152(1)** 593–614.

Peng, J., T. Zhu, H. Luo, K. Toh. 2015. Semi-definite programming relaxation of quadratic assignment problems based on nonredundant matrix splitting. *Comput. Optim. Appl.* **60(1)** 171–198.

Sahinidis, N. V. 1996. BARON: A general purpose global optimization software package. *J. Glob. Optim.* **8** 201–205.

Saxena, A., P. Bonami, J. Lee. 2010. Convex relaxation of non-convex mixed integer quadratically constrained programs: extended formulations. *Math. Program. (Ser. B)* **124** 383–411.

Saxena, A., P. Bonami, J. Lee. 2011. Convex relaxation of nonconvex mixed integer quadratically constrained programs: projected formulations. *Math. Program.* **130** 359–413.

Sherali, H., A. Alameddine. 1992. A new reformulation linearization algorithm for bilinear programming problems. *J. Glob. Optim.* **2** 379–410.

Shu, J., M. Song. 2014. Dynamic dontainer deployment: Two-stage robust model, complexity, and computational results. *INFORMS J. Comput.* **26(1)** 135–149.

Toh, K., M. Todd, R. Tutuncu. 1999. SDPT3: Matlab software package for semidefinite programming. *Optim. Methods Softw.* **11** 545–581.

Vandenbussche, D., G. Nemhauser. 2005. A branch-and-cut algorithm for nonconvex quadratic programming with box constraints. *Math. Program.* **102** 559–575.

Ye, Y. 1999. Approximating quadratic programming with bound and quadratic constraints. *Math. Program.* **84(2)** 219–226.

Zeng, B., L. Zhao. 2013. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Oper. Res. Lett.* **41** 457–461.