

# Convex Maximization via Adjustable Robust Optimization

Aras Selvi

Imperial College Business School, Imperial College London, a.selvi19@imperial.ac.uk

Aharon Ben-Tal

Faculty of Industrial Engineering and Management, Technion - Israel Institute of Technology,  
Shenkar College, Ramat Gan, Israel,  
Center of Economic and Business Research, Tilburg University, abental@technion.ac.il

Ruud Brekelmans

Department of Econometrics and Operations Research, Tilburg University, r.c.m.brekelmans@uvt.nl

Dick den Hertog

Faculty of Economics and Business, University of Amsterdam, d.denhertog@uva.nl

Maximizing a convex function over convex constraints is an NP-hard problem in general. We prove that such a problem can be reformulated as an adjustable robust optimization (ARO) problem where each adjustable variable corresponds to a unique constraint of the original problem. We use ARO techniques to obtain approximate solutions to the convex maximization problem. In order to demonstrate the complete approximation scheme, we distinguish the case where we have just one nonlinear constraint and the case where we have multiple linear constraints. Concerning the first case, we give three examples where one can analytically eliminate the adjustable variable and approximately solve the resulting static robust optimization problem efficiently. More specifically, we show that the norm constrained log-sum-exp (geometric) maximization problem can be approximated by (convex) exponential cone optimization techniques. Concerning the second case of multiple linear constraints, the equivalent ARO problem can be represented as an adjustable robust linear optimization (ARLO) problem. Using linear decision rules then returns a safe approximation of the constraints. The resulting problem is a convex optimization problem, and solving this problem gives an upper bound on the global optimum value of the original problem. By using the optimal linear decision rule, we obtain a lower bound solution as well. We derive the approximation problems explicitly for quadratic maximization, geometric maximization, and sum-of-max-linear-terms maximization problems with multiple linear constraints. Numerical experiments show that, contrary to the state-of-the-art solvers, we can approximate large-scale problems swiftly with tight bounds. In several cases, we have equal upper and lower bounds, which concludes that we have global optimality guarantees in these cases.

*Key words:* nonlinear optimization; convex maximization; adjustable robust optimization

*History:* revised on September 2, 2021

---

## 1. Introduction

We propose a new approximation method for the convex maximization problem:

$$\begin{aligned} \max_{x \in \mathbb{R}^n} f(Ax + b) \\ \text{s. t. } x \in U, \end{aligned}$$

where  $U \subset \mathbb{R}^n$  is a compact set defined by convex constraints and  $f : \mathbb{R}^m \mapsto \mathbb{R}$  is a closed convex function with an arbitrary linear input  $Ax + b$  for  $A \in \mathbb{R}^{m \times n}$ , and  $b \in \mathbb{R}^m$ . The seminal paper of Tuy (1964) is regarded as the first approach to solve convex maximization problems, where  $U$  is a polyhedron.

There are many real-life problems that can be reformulated as convex maximization problems. Rebennack et al. (2009) show that the fixed charge network flow problem can be formulated as a convex maximization problem. Moreover, Zwart (1974) shows that two important problem classes are equivalent to convex maximization problems, namely cost minimization problems with the cost function being subject to economies of scale, and linear optimization problems that involve ‘yes’ or ‘no’ decisions (binary variables). Many machine learning (ML) problems can be formulated as convex maximization problems, for instance Mangasarian (1996) shows that fundamental problems in ML, misclassification minimization and feature selection, are equivalent to convex maximization problems. The same author more recently proposes the use of absolute value inequalities for classifying unlabeled data, which results in a problem of minimizing a concave function on a polyhedral set (Mangasarian 2015). Other important examples from data science are variants of principal component analysis (PCA). Zass and Shashua (2007) show that the nonnegative PCA problem and the sparse PCA problem are convex quadratic maximization problems. Additionally, a popular approach to solve difference of convex functions (DC) programs is the convex-concave method, which iteratively solves convex minimization and convex maximization problems (assuming the constraints are convex) (Lipp and Boyd 2016). DC programming is being used to solve many problems in machine learning, data science, biology, security and transportation (Le Thi and Pham Dinh 2018). Also many problems arising in graph theory can be formulated as convex maximization problems, a well known example is the MAX-CUT problem (Goemans and Williamson 1994). A lot of variations of integer linear and integer quadratic optimization problems over polyhedra can be written as convex maximization problems (Benson 1995). Convex maximization naturally appears

in robust optimization when finding the worst-case scenario of a constraint which is a convex function of the uncertain parameter. A similar problem appears when one applies the adversarial approach (Bienstock and Özbay 2008) to solve a robust convex optimization problem. In this approach, at the step of adding worst-case uncertainty realization to the discrete uncertainty set, one needs to maximize convex functions.

A local or global solution of the convex maximization problem is necessarily at an extreme point of the feasible region (Rockafellar 1970), hence there exist many methods to solve convex maximization problems by searching for extreme point solutions, but this approach is itself very hard. It is shown that the convex maximization problem is NP-hard in very simple cases (e.g., quadratic maximization over a hypercube), and even verifying local optimality is NP-hard (Pardalos and Schnitger 1988). Hence, there are many papers to approximate the convex maximization problem (Benson 1995). The survey paper (Pardalos and Rosen 1986) collects such works until the 1980s. Most of the proposed methods use linear underestimator functions, which are derived by the so-called convex envelopes. These algorithms have a disadvantage, namely, the size of the sub problems grows in every new iteration, which makes them impractical in general. Moreover, the proposed methods in the literature are designed only for some specific cases (e.g., (Zwart 1974)). All of the well-accepted methods to solve the most studied convex maximization problem, quadratic maximization, are based on cutting plane methods, iterative numerical methods such as the element methods, and techniques of branches and borders based on the decomposition of the feasible set as summarized in (Audet et al. 2005, Andrianova et al. 2016). These papers also indicate that such methods do not provide solutions in reasonable time for practical problems.

Convex maximization is frequently being studied in the scope of DC programming in recent optimization research. As summarized by Lipp and Boyd (2016), the early approaches reformulated the DC programming problems as convex maximization problems (Tuy 1986, Tuy and Horst 1988, Horst et al. 1991). One can see how the methods in convex maximization are adopted for DC programming literature in the work of Horst and Thoai (1999). Lipp and Boyd (2016) provide a thorough literature review in convex maximization, and state that the literature to solve such problems mostly relies on branch and bound or cutting plane methods which are very slow in practice. In this respect, in order to be able to cope with large DC programming problems, Lipp and Boyd (2016)

propose a heuristic algorithm to find a decent local solution of the convex maximization problem.

In this paper a new method to approximately solve the convex maximization problem is presented. The method starts by reformulating the convex maximization problem as an adjustable robust optimization (ARO) problem. The ARO problem has a number of adjustable variables equal to the number of the original constraints. The adjustable variables appear nonlinearly, hence the ARO problem is still a hard problem. Therefore, we apply approximation methods used in the ARO literature in order to approximate the ARO problem. This way we derive convex optimization problems which provide upper and lower bounds of the original convex maximization problem.

The rest of the paper is organized as follows. In Section 2, we present our main theorem for single constrained convex maximization, and show how to reformulate the convex maximization problem as an ARO problem. We exploit the relationship between equivalent formulations to show how to obtain a solution in the original problem by using the solution of the ARO problem. We give three cases of convex maximization over a single norm constraint, show how to analytically eliminate the adjustable variable, and (approximately) solve the resulting static robust optimization problem. The approximate solution of the ARO problem gives an upper bound to the convex maximization problem, and by using this solution we show how to obtain a lower bound for the original problem. In Section 3, we derive the ARO reformulation of a general convex maximization problem with arbitrary convex constraints, and show that one way to approximately solve this ARO problem is by relaxing adjustable variables to static variables. We specifically investigate the case of convex maximization over multiple linear constraints. It is shown that the ARO reformulation of this case is an adjustable robust linear optimization (ARLO) problem. Adoption of linear decision rules for the adjustable variables enables one to derive efficient upper and lower bound approximation problems. Explicit approximations are obtained for convex quadratic, geometric, and sum-of-max-linear-terms maximization problems. The numerical experiments follow in Section 4, which illustrate that our approximation problems can be solved significantly faster than the state-of-the-art optimization solvers, and provide tight optimality gaps in most of the cases. In cases where the upper bound is equal to the lower bound, a guarantee of global maximizer is obtained. We conclude the paper in Section 5 by discussing our findings and giving future research directions.

## 2. ARO Reformulation of Single-Constrained Convex Maximization

In this section we show how to reformulate a single convex constrained convex maximization problem as an equivalent ARO problem, and how to (approximately) solve it. In the next section we will generalize these results to problems with multiple constraints.

Let  $f : \mathbb{R}^m \mapsto \mathbb{R}$  and  $g : \mathbb{R}^n \mapsto \mathbb{R}$  be closed convex functions. We assume that  $\exists \bar{x} \in \mathbb{R}^n : g(\bar{x}) < \rho$  for scalar  $\rho \in \mathbb{R}$ . Moreover, let  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ . We consider the following convex maximization problem with a single convex constraint:

$$\begin{aligned} & \max_{x \in \mathbb{R}^n} f(Ax + b) \\ & \text{s. t. } g(x) \leq \rho. \end{aligned} \tag{1}$$

The feasible set is denoted by  $U$ , i.e.,  $U = \{x \in \mathbb{R}^n : g(x) \leq \rho\}$ . Let  $f^*$  denote the convex conjugate of  $f$ . The perspective of  $f$  is defined as  $z_0 f(\frac{z}{z_0})$  for  $z_0 \geq 0$ , and our understanding for  $0f(\frac{z}{0})$  is the recession function  $\lim_{z_0 \downarrow 0} z_0 f(\frac{z}{z_0})$  (Rockafellar 1970) for the rest of the paper. The following theorem shows that problem (1) is equivalent to an ARO problem where  $w \in \text{dom } f^*$  is the uncertain parameter and  $\lambda \geq 0$  is the adjustable variable.

**THEOREM 1.** *The optimal objective value of problem (1) is equal to the optimal objective value of the following adjustable robust optimization problem:*

$$\begin{aligned} & \inf_{\tau \in \mathbb{R}} \tau \\ & \text{s. t. } \forall w \in \text{dom } f^*, \exists \lambda \geq 0: \quad \lambda \rho + \lambda g^* \left( \frac{A^\top w}{\lambda} \right) + b^\top w - f^*(w) \leq \tau. \end{aligned} \tag{2}$$

*The solution  $\bar{x}$  that attains this value satisfies  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : x \in U\}$  with  $\bar{w} \in \text{dom } f^*$  being the parameter realization where the constraint of (2) is tight at optimality.*

*Proof* Problem (1) can be written as:

$$\begin{aligned} & \inf_{\tau \in \mathbb{R}} \tau \\ & \text{s. t. } f(Ax + b) \leq \tau, \forall x \in U. \end{aligned} \tag{3}$$

Since  $f$  is a closed convex function, we have:

$$f(z) = f^{**}(z) = \sup_{w \in \text{dom } f^*} \{z^\top w - f^*(w)\},$$

where  $f^{**}$  is the biconjugate of  $f$ . Hence, the constraint of problem (3) becomes:

$$\begin{aligned} \forall x \in U : f(Ax + b) \leq \tau &\iff \forall x \in U : \sup_{w \in \text{dom } f^*} \{(Ax + b)^T w - f^*(w)\} \leq \tau \\ &\iff \sup_{x \in U} \left\{ \sup_{w \in \text{dom } f^*} \{(A^T w)^T x + b^T w - f^*(w)\} \right\} \leq \tau \end{aligned} \quad (4a)$$

$$\iff \sup_{w \in \text{dom } f^*} \left\{ \sup_{x \in U} \{(A^T w)^T x\} + b^T w - f^*(w) \right\} \leq \tau, \quad (4b)$$

where in step (4a) we exploit the fact that if a constraint holds for the worst-case then it holds for any case, and in step (4b) we change the order of the supremum operators. We replace the inner problem (linear maximization over a convex set) with its Lagrangian dual problem and obtain:

$$\begin{aligned} \sup_x \{(A^T w)^T x : g(x) \leq \rho\} &= \inf_{\lambda \geq 0} \left\{ \sup_x \{(A^T w)^T x - \lambda g(x)\} + \lambda \rho \right\} \\ &= \inf_{\lambda > 0} \left\{ \lambda \sup_x \left\{ \frac{(A^T w)^T}{\lambda} x - g(x) \right\} + \lambda \rho \right\} \end{aligned} \quad (5a)$$

$$= \inf_{\lambda \geq 0} \left\{ \lambda g^* \left( \frac{A^T w}{\lambda} \right) + \lambda \rho \right\}. \quad (5b)$$

We substitute (5b) into (4b) to conclude:

$$\begin{aligned} \forall x \in U : f(Ax + b) \leq \tau \\ \iff \sup_{w \in \text{dom } f^*} \left\{ \inf_{\lambda \geq 0} \left\{ \lambda \rho + \lambda g^* \left( \frac{A^T w}{\lambda} \right) \right\} + b^T w - f^*(w) \right\} \leq \tau \end{aligned} \quad (6a)$$

$$\iff \forall w \in \text{dom } f^*, \exists \lambda \geq 0 : \lambda \rho + \lambda g^* \left( \frac{A^T w}{\lambda} \right) + b^T w - f^*(w) \leq \tau. \quad (6b)$$

Minimizing  $\tau$  over (6b) gives problem (2). The optimal solution  $\bar{x}$  of problem (1) can be retrieved from the optimal solution of ARO problem (2). Firstly, notice that  $\bar{x}$  solves the outer problem of (4a) and the inner problem of (4b). Let  $(\bar{\lambda}, \bar{w})$  denote the solution of the optimization problem (6a), where  $\bar{\lambda}$  is a function of  $\bar{w}$  due to the inner problem. By the equivalence introduced above,  $\bar{w}$  solves problem (4b), hence we can retrieve  $\bar{x}$  from the inner problem of (4b), i.e.,:

$$\bar{x} \in \arg \sup \{(A^T \bar{w})^T x : g(x) \leq \rho\}, \quad (7)$$

which is a convex optimization problem.  $\square$

REMARK 1. If  $g$  is differentiable and the inverse of its gradient ( $\nabla^{-1}g(\cdot)$ ) exists, then we can analytically obtain the solution as  $\bar{x} = \nabla^{-1}g\left(\frac{A^\top \bar{w}}{\lambda}\right)$  which follows from (5a). ■ In the ARO reformulation (2) the adjustable variable  $\lambda$  appears nonlinearly, hence this is also a difficult problem. However, there is only one adjustable variable, and it appears in a single (semi-infinite) constraint. In the following, we consider three cases where one can derive explicit expressions for  $\lambda$  (e.g., the analytic worst case for  $\lambda$ , as a function of  $w$ ), which circumvents the nonlinearity. In these examples we do not show how to derive the convex conjugate of various  $g(x)$  functions, but these can be found in, e.g., (Boyd and Vandenberghe 2004).

COROLLARY 1 (2-NORM CONSTRAINT). Let  $g(x) := \|x - a\|_2$  in problem (1), where  $a \in \mathbb{R}^n$  is a parameter. Then, the global optimum value  $\tau^*$  of this problem is

$$\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_2 + a^\top A^\top w + b^\top w - f^*(w). \quad (8)$$

Furthermore, if the domain of  $f^*$  consists of linear inequalities, an upper bound value of  $\tau^*$  can be found by solving a convex relaxation of (8). A corresponding lower bound solution of problem (1) is  $x^* = (A^\top w^*) \frac{\rho}{\|A^\top w^*\|_2} + a$ , where  $w^* \in \text{dom } f^*$  is the upper bound solution.

*Proof* Let the feasible set of problem (1) be defined as:

$$U_1 = \left\{ x \in \mathbb{R}^n : g(x) = \frac{1}{2} \|x - a\|_2^2 \leq \frac{1}{2} \rho^2 \right\}.$$

The conjugate of the squared norm is  $g^*(z) = \frac{1}{2} \|z\|_2^2 + z^\top a$ , hence, minimizing  $\tau$  over constraint (6) is minimizing  $\tau$  subject to:

$$\begin{aligned} & \sup_{w \in \text{dom } f^*} \left\{ \inf_{\lambda \geq 0} \left\{ \frac{1}{2} \rho^2 \lambda + \lambda g^* \left( \frac{A^\top w}{\lambda} \right) \right\} + b^\top w - f^*(w) \right\} \leq \tau \\ \iff & \sup_{w \in \text{dom } f^*} \left\{ \min_{\lambda \geq 0} \left\{ \frac{1}{2} \rho^2 \lambda + \frac{1}{2} \lambda \left\| \frac{A^\top w}{\lambda} \right\|_2^2 \right\} + a^\top A^\top w + b^\top w - f^*(w) \right\} \leq \tau \\ \iff & \sup_{w \in \text{dom } f^*} \left\{ \rho \|A^\top w\|_2 + a^\top A^\top w + b^\top w - f^*(w) \right\} \leq \tau, \end{aligned}$$

where the last step holds since the inner minimization problem is a convex problem with the optimal decision rule  $\bar{\lambda} = \rho^{-1} \|A^\top w\|_2$ . Therefore, for the feasible set  $U_1$ , problem (1) is equivalent to the following optimization problem:

$$\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_2 + a^\top A^\top w + b^\top w - f^*(w). \quad (9)$$

Notice that at the original problem (1) with  $U = U_1$ , the constraint is convex and the convexity of the objective function  $f(Ax + b)$  makes the problem non-convex, whereas in problem (9) maximizing  $-f^*(w)$  is fine and the 2-norm makes the problem non-convex.

Although we conclude that problem (9) is non-convex, there exist strong approximation methods, and an example is when the domain of  $f^*$  consists of linear inequalities, i.e.,

$$\text{dom } f^* = \{w : \alpha_i^\top w \leq \beta_i, i = 1, \dots, d\},$$

for  $\alpha_i \in \mathbb{R}^m$ ,  $\beta_i \in \mathbb{R}$ . In this setting, the only difficult part is maximizing the sum of a convex function and a concave function over linear constraints. To be able to approximate problem (9) with a convex problem we use the reformulation-linearization technique (RLT), whose details can be found in (Sherali and Adams 2013). We also tighten the RLT relaxation by using a ‘positive semi-definite cut’ (Sherali and Fraticelli 2002) and obtain (derivation is in Appendix A of the Online Supplement):

$$\begin{aligned} \sup_{V \in \mathbb{S}^{m \times m}, w \in \mathbb{R}^m} \quad & \rho \sqrt{\text{tr}(A^\top V A)} + a^\top A^\top w + b^\top w - f^*(w) \\ \text{s. t.} \quad & \alpha_i^\top w - \beta_i \leq 0, \quad i = 1, \dots, d \\ & \alpha_i^\top V \alpha_j - (\beta_i \alpha_j + \beta_j \alpha_i)^\top w + \beta_i \beta_j \geq 0, \quad i \leq j = 1, \dots, d \end{aligned} \quad (10)$$

$$\begin{pmatrix} V & w \\ w^\top & 1 \end{pmatrix} \succeq 0,$$

where  $\text{tr}(\cdot)$  denotes the trace operator. For instance, suppose  $f$  is the log-sum-exp function. Since  $f^*(w)$  is the negative entropy of  $w$  in its domain (standard  $m$ -dimensional simplex), problem (10) is an exponential-cone representable problem. It can further be proved that for this case the semi-definite constraint is redundant, and the optimal value of  $V$  is the diagonal matrix  $\text{Diag}(w)$  (Selvi et al. 2020). So the only variable is  $w \in \mathbb{R}^m$  and we can solve problem (10) with the exponential cone solver of MOSEK (MOSEK ApS 2019b).

Going back to the general setting of  $f$ , it is straightforward to see that problem (10) upper bounds the optimal objective value of problem (9) since an optimal solution  $\bar{w}$  in problem (9) is feasible in problem (10) by taking  $V = \bar{w} \bar{w}^\top$ . We can also obtain a lower bound on problem (9) by using the upper bound solution. Solving  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : x \in U_1\}$ , the optimal  $\bar{x}$  value can be recovered by  $\bar{x} = \nabla^{-1} g \left( \frac{A^\top \bar{w}}{\lambda} \right) = (A^\top \bar{w}) \frac{\rho}{\|A^\top \bar{w}\|_2} + a$ . Since we approximate  $\bar{w}$  of problem (9) with  $w^*$  in problem (10), a lower bounding approximation of



the optimal solution  $\bar{x}$  is  $x^* = (A^\top w^*) \frac{\rho}{\|A^\top w^*\|_2} + a$ . Moreover, the constraint of the original convex maximization problem is indeed tight for  $x^*$ , i.e.,  $\frac{1}{2}\|x^* - a\|_2^2 = \frac{1}{2}\rho^2$ . This shows us that this method gives us an extreme point lower bound solution, which is desirable as otherwise  $x^*$  cannot even be locally optimal, i.e., every local and global solution of problem (1) is at an extreme point of the feasible set.  $\square$

Numerical experiments of Corollary 1 are in Section 4.1

**COROLLARY 2 (BOX CONSTRAINTS).** Let  $g(x) := \|x - a\|_\infty$  in problem (1), where  $a \in \mathbb{R}^n$  is a parameter. Then, the global optimum value  $\tau^*$  of this problem is

$$\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_1 + a^\top A^\top w + b^\top w - f^*(w), \quad (11)$$

which can be solved via mixed-integer convex optimization. Furthermore, the optimal solution  $\bar{x}$  of problem (1) that attains  $\tau^*$  can be found by solving the linear optimization problem  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : \|x - a\|_\infty \leq \rho\}$ , where  $\bar{w}$  is the solution of (11).

*Proof* The feasible region defined by box constraints is:

$$U_2 = \{x \in \mathbb{R}^n : g(x) = \|x - a\|_\infty \leq \rho\}.$$

Although box constraints are actually a collection of multiple constraints, by using the  $\infty$ -norm we can apply our theorem for a single constraint. Using the conjugate

$$g^*(z) = \begin{cases} z^\top a & \text{if } \|z\|_1 \leq 1 \\ \infty & \text{otherwise,} \end{cases}$$

in (6), we obtain that problem (1) is equivalent to minimizing  $\tau$  over:

$$\begin{aligned} & \sup_{w \in \text{dom } f^*} \left\{ \min_{\lambda \geq 0} \{ \lambda \rho + a^\top A^\top w : \|A^\top w\|_1 \leq \lambda \} + b^\top w - f^*(w) \right\} \leq \tau \\ \iff & \sup_{w \in \text{dom } f^*} \{ \rho \|A^\top w\|_1 + a^\top A^\top w + b^\top w - f^*(w) \} \leq \tau, \end{aligned}$$

since the minimizer of the inner problem is  $\bar{\lambda} = \|A^\top w\|_1$ . This also shows that the piecewise linear decision rule (LDR) is optimal for this problem, as  $\bar{\lambda}$  has the structure of a piecewise LDR. We can conclude that if the uncertainty set is  $U_2$ , the optimal value of problem (1) is given by:

$$\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_1 + a^\top A^\top w + b^\top w - f^*(w). \quad (12)$$

Notice that (12) is still a hard problem due to the convexity of  $\rho \|A^\top w\|_1$ . However, this 1-norm can be represented by linear terms using extra binary variables for absolute values (see, e.g., (Löfberg 2016)). There are many efficient solvers which can solve the resulting mixed-integer convex optimization problem. For example, if  $f$  is a log-sum-exp function, then  $f^*$  is the negative entropy (with linear domain constraints), hence problem (12) will be a mixed-integer exponential cone representable problem and MOSEK can efficiently solve these type of problems.

From Theorem 1 it follows that the optimal solution  $\bar{x}$  of the original problem that attains the value  $\tau^*$  can be retrieved by solving  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : x \in U_2\}$  (as we cannot derive a closed-form solution due to the  $\infty$ -norm).  $\square$

Numerical experiments of Corollary 2 can be found in Section 4.2.

**COROLLARY 3 ( $p$ -NORM CONSTRAINT).** Let  $g(x) := \|x - a\|_p$  in problem (1), where  $p \geq 1$  and  $a \in \mathbb{R}^n$  is a parameter. Then, the global optimum value  $\tau^*$  of this problem is

$$\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_q + a^\top A^\top w + b^\top w - f^*(w), \quad (13)$$

where  $q$  is given by  $1/q + 1/p = 1$ . The solution  $\bar{x}$  that attains this value in problem (1) is obtained by solving  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : \|x - a\|_p \leq \rho\}$ , where  $\bar{w}$  is the solution of (13).

*Proof* The dual norm of the  $p$ -norm is the  $q$ -norm, where  $q$  is given by  $\frac{1}{q} + \frac{1}{p} = 1$ . The rest of this proof is analogous to the proof of Corollary 2, hence we can conclude that the optimal objective value of problem (1) is  $\tau^* = \sup_{w \in \text{dom } f^*} \rho \|A^\top w\|_q + a^\top A^\top w + b^\top w - f^*(w)$ . The solution  $\bar{x}$  can be found by similar arguments as in Corollary 2.  $\square$

Corollary 3 generalizes the previous two corollaries, i.e.,  $p = 2, \infty$ . Solving problem (13) is in general difficult, however the non-convexity of this problem is not due to  $f^*(w)$ , but the dual norm of the original  $g(x)$ . This may in turn significantly simplify solving the convex maximization problem. In Corollary 1 we showed cases where we can efficiently approximate problem (13) for  $p = 2$  and obtain the corresponding lower bound solution, and in Corollary 2 we showed how to globally solve problem (13) by mixed-integer convex optimization, along with the corresponding solution of problem 1 for  $p = \infty$ .

### 3. ARO Reformulation of Multiple-Constrained Convex Maximization

In this section, we first extend Theorem 1 to convex maximization problems with multiple convex constraints. Then, we give tractable upper and lower bound relaxation problems for the case of multiple linear constraints.

### 3.1. Convex Maximization over Multiple Convex Constraints

Let  $f : \mathbb{R}^m \mapsto \mathbb{R}$  be a closed convex objective function, and  $g_j : \mathbb{R}^n \mapsto \mathbb{R}, j = 1, \dots, q$  be closed convex functions. We assume that  $\exists \bar{x}_j \in \mathbb{R}^n : g_j(\bar{x}_j) < \rho_j$  for  $j = 1, \dots, q$ . Similarly to the previous setting, let  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ . We consider the following convex maximization problem with multiple convex constraints:

$$\begin{aligned} & \max_{x \in \mathbb{R}^n} f(Ax + b) \\ & \text{s. t. } g_j(x) \leq \rho_j, \quad j = 1, \dots, q. \end{aligned} \tag{14}$$

Let  $U = \{x \in \mathbb{R}^n : g_i(x) \leq \rho_i, i = 1, \dots, q\}$  denote the feasible set of problem (14).

**THEOREM 2.** *The optimal objective value of problem (14) is equal to the optimal objective value of the following adjustable robust optimization problem:*

$$\begin{aligned} & \inf_{\tau} \tau \\ & \text{s. t. } \forall w \in \text{dom } f^*, \exists \lambda \in \mathbb{R}_+^q, z \in \mathbb{R}^{n \times q} : \begin{cases} \sum_{j=1}^q \lambda_j \rho_j + \sum_{i=1}^q \lambda_i g_i^* \left( \frac{z_i}{\lambda_i} \right) + b^\top w - f^*(w) \leq \tau \\ \sum_{j=1}^q z_j = A^\top w, \end{cases} \end{aligned} \tag{15}$$

where  $z_j$  denotes the  $j$ -th column of  $z$ , and  $\mathbb{R}_+$  is the set of nonnegative real numbers. The optimal solution  $\bar{x}$  that attains this value satisfies  $\bar{x} \in \arg \sup \{(A^\top \bar{w})^\top x : x \in U\}$  with  $\bar{w} \in \text{dom } f^*$  being the parameter realization where the first constraint of (15) is tight at the optimal solution.

*Proof* From the proof of Theorem 1 we see that problem (14) can be written as the minimization of  $\tau$  over:

$$\sup_{w \in \text{dom } f^*} \left\{ \sup_{x \in U} \{(A^\top w)^\top x\} + b^\top w - f^*(w) \right\} \leq \tau. \tag{16}$$

Consider the inner problem in constraint (16). Taking its Lagrangian dual problem gives:

$$\begin{aligned} & \sup_{x \in U} \{(A^\top w)^\top x\} \\ & = \inf_{\lambda \in \mathbb{R}_+^q} \left\{ \sup_{x \in \mathbb{R}^n} \left\{ (A^\top w)^\top x - \sum_{i=1}^q \lambda_i g_i(x) \right\} + \sum_{j=1}^q \lambda_j \rho_j \right\} \\ & = \inf_{\lambda \in \mathbb{R}_+^q} \left\{ \left( \sum_{i=1}^q \lambda_i g_i \right)^* (A^\top w) + \sum_{j=1}^q \lambda_j \rho_j \right\}. \end{aligned} \tag{17}$$

To simplify  $\left(\sum_{i=1}^q \lambda_i g_i\right)^*$ , we use the fact that the conjugate of sum of convex functions can be written as the *infimal convolution* of these functions (Rockafellar 1970). This equivalence gives:

$$\begin{aligned} \left(\sum_{i=1}^q \lambda_i g_i\right)^*(A^\top w) &= \inf_{z_1, \dots, z_q \in \mathbb{R}^n} \left\{ \sum_{i=1}^q (\lambda_i g_i)^*(z_i) : \sum_{j=1}^q z_j = A^\top w \right\} \\ &= \inf_{z_1, \dots, z_q \in \mathbb{R}^n} \left\{ \sum_{i=1}^q \lambda_i (g_i)^*\left(\frac{z_i}{\lambda_i}\right) : \sum_{j=1}^q z_j = A^\top w \right\}, \end{aligned}$$

where the last step holds since  $\lambda_i g_i^*\left(\frac{z_i}{\lambda_i}\right) = \lambda_i \sup_x \left\{ \frac{z_i^\top x}{\lambda_i} - g_i(x) \right\} = (\lambda_i g_i)^*(z_i)$ . Therefore, (17) becomes:

$$\inf_{\lambda \in \mathbb{R}_+^q} \left\{ \sum_{j=1}^q \lambda_j \rho_j + \inf_{z_1, \dots, z_q \in \mathbb{R}^n} \left\{ \sum_{i=1}^q \lambda_i g_i^*\left(\frac{z_i}{\lambda_i}\right) : \sum_{j=1}^q z_j = A^\top w \right\} \right\},$$

and so constraint (16) reduces to:

$$\sup_{w \in \text{dom } f^*} \left\{ \inf_{\substack{z \in \mathbb{R}^{n \times q} \\ \lambda \in \mathbb{R}_+^q}} \left\{ \sum_{j=1}^q \lambda_j \rho_j + \sum_{i=1}^q \lambda_i g_i^*\left(\frac{z_i}{\lambda_i}\right) + b^\top w - f^*(w) : \sum_{j=1}^q z_j = A^\top w \right\} \right\} \leq \tau. \quad (18)$$

Minimizing  $\tau$  over (18) gives us the global optimum value of problem (14). The optimal solution  $\bar{x}$  of the original problem can be retrieved by solving  $\bar{x} \in \text{argsup} \{(A^\top \bar{w})^\top x : x \in U\}$  where  $\bar{w} \in \text{dom } f^*$  maximizes the left-hand side of (18) (the reasoning is identical to Theorem 1, i.e., such  $\bar{w}$  maximizes (16) equivalently).  $\square$

One way to approximate the ARO problem (15) is to use static variables for the adjustable variables. Assume without loss of generality that in problem (14) we have  $x \in \mathbb{R}_+^n$ . It is possible to verify that this problem is equivalent to problem (15) with the last constraint being  $\sum_{j=1}^q z_j \geq A^\top w$  (instead of equality). By relaxing the adjustable variables to static variables, we obtain a tractable approximation of the problem. Hence, the following convex optimization problem returns an upper bound to the original problem:

$$\begin{aligned} &\inf_{\tau \in \mathbb{R}, \lambda \in \mathbb{R}_+^q, z \in \mathbb{R}^{n \times q}} \tau \\ \text{s. t.} \quad &\sum_{j=1}^q \lambda_j \rho_j + \sum_{i=1}^q \lambda_i g_i^*\left(\frac{z_i}{\lambda_i}\right) + \sup_{w \in \text{dom } f^*} \{b^\top w - f^*(w)\} \leq \tau \quad (19) \\ &\sup_{w \in \text{dom } f^*} \{A_i^\top w\} \leq \sum_{j=1}^q z_{ij}, \quad i = 1, \dots, n. \end{aligned}$$

Moreover, we know that solving  $\arg \sup_{x \in \mathbb{R}_+^n} \{(A^\top \bar{w})^\top x - \sum_{i=1}^q \bar{\lambda}_i g_i(x)\}$  gives us the global optimum solution, where  $(\bar{\lambda}, \bar{w})$  solves the optimization problem of (18). Since we used a safe-approximation for this problem, we can use the  $\lambda$  value which solves problem (19) as an approximation of  $\bar{\lambda}$ , and try  $n + 1$  many scenarios for  $w$  solving each supremization problem in (19), in order to obtain a lower bound  $\bar{x}$ . Such a lower bound approach is thoroughly described in Section 3.2 where we approximate the problem of maximizing a convex function over linear constraints.

When the constraints are linear, problem (15) is linear in the adjustable variables, hence this deserves a separate treatment which is done in the next subsection.

### 3.2. Convex Maximization over a Polyhedron

In this subsection, we consider problem (14) in which the feasible set is a polyhedron with a nonempty interior. Application of Theorem 2 yields an attractive adjustable robust linear optimization reformulation, for which efficient approximations exist in the literature. We first illustrate how the problem can be approximated with lower and upper bounds. Then, we consider special cases of the objective function  $f$ : quadratic, log-sum-exp (geometric), and sum-of-max-terms.

Formally, we work on the following problem for  $D \in \mathbb{R}^{q \times n}$  and  $d \in \mathbb{R}^n$ :

$$\begin{aligned} \max_{x \in \mathbb{R}_+^n} & f(Ax + b) \\ \text{s. t. } & Dx \leq d, \end{aligned} \tag{20}$$

which is a special case of problem (14) with

$$U = \{x \in \mathbb{R}_+^n : Dx \leq d\}. \tag{21}$$

Let  $D_{(j)}$  denote the  $j$ -th row of  $D$ . Then, the  $j$ -th constraint of  $U$  is given by

$$g_j(x) \leq \rho_j \iff D_{(j)}x \leq d_j.$$

By Theorem 2, the optimal objective values of problem (20) and the following problem are equal:

$$\inf_{\tau \in \mathbb{R}} \tau \text{ s. t. } \forall w \in \text{dom } f^*, \exists \lambda \in \mathbb{R}_+^q, z \in \mathbb{R}^{n \times q} : \begin{cases} d^\top \lambda + b^\top w - f^*(w) \leq \tau \\ z_i \leq D_{(i)} \lambda_i, \quad i = 1, \dots, q \\ \sum_{j=1}^q z_j = A^\top w, \end{cases}$$

Constraints  $\sum_{j=1}^q z_j = A^\top w$  and  $z_i \leq D_{(i)} \lambda_i$ ,  $i = 1, \dots, q$  together can be written as  $A^\top w \leq D^\top \lambda$ . Therefore, problem (20) has the same optimal objective value as the ARO problem:

$$\inf_{\tau \in \mathbb{R}} \tau \text{ s. t. } \forall w \in \text{dom } f^*, \exists \lambda \in \mathbb{R}^q : \begin{cases} d^\top \lambda + b^\top w - f^*(w) \leq \tau \\ D^\top \lambda \geq A^\top w \\ \lambda \geq \mathbf{0}. \end{cases} \quad (22)$$

Notice that the final problem is a linear ARO problem with fixed recourse (linearity is obtained by lifting the  $-f^*(w)$  term to the uncertainty set). There are many possible methods one can use to solve such a problem, for example, one can solve this problem to optimality by eliminating the adjustable variables via Fourier-Motzkin Elimination for ARO (Zhen et al. 2018), which is efficiently applicable for small-sized problems. We refer to Yanıkođlu et al. (2019) for a survey of alternative methods to solve this linear ARO problem. In the remaining of this section we show how to derive tractable problems to find upper and lower bounds on the optimal objective value of problem (20).

In Appendix B of the Online Supplement it is shown that by using linear decision rules, one obtains a (tractable) safe approximation of the constraints of ARO problem (22) as:

$$\inf_{u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times m}, r \in \mathbb{R}^q, \tau \in \mathbb{R}} \tau \text{ s. t. } \begin{cases} d^\top (u + Vw + rw_0) + b^\top w + w_0 - \tau \leq 0, & \forall (w_0 \ w^\top)^\top \in W \quad (23a) \\ -D^\top (u + Vw + rw_0) + A^\top w \leq \mathbf{0}, & \forall (w_0 \ w^\top)^\top \in W \quad (23b) \\ -(u + Vw + rw_0) \leq \mathbf{0} & \forall (w_0 \ w^\top)^\top \in W, \quad (23c) \end{cases}$$

with  $W = \{(w_0 \ w^\top)^\top \in \mathbb{R}^{m+1} : w_0 + f^*(w) \leq 0\}$ , which is used to prove the following result.

**THEOREM 3.** *The optimal objective value of the following problem is an upper bound to the optimal objective value of problem (20):*

$$\inf_{u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times m}, r \in \mathbb{R}^q, \tau \in \mathbb{R}} \tau \text{ s. t. } \begin{cases} d^\top u + (1 + d^\top r) f\left(\frac{V^\top d + b}{1 + d^\top r}\right) \leq \tau & (24a) \\ 1 + d^\top r \geq 0 \\ -D_i^\top u + (-D_i^\top r) f\left(\frac{A_i - V^\top D_i}{-D_i^\top r}\right) \leq 0 & i = 1, \dots, n \quad (24b) \\ -D_i^\top r \geq 0 \\ -u_i + (-r_i) f\left(\frac{V_{(i)}^\top}{r_i}\right) \leq 0 & i = 1, \dots, q \quad (24c) \\ -r_i \geq 0. \end{cases}$$

Here,  $V_{(i)}$  stands for the  $i$ -th row of  $V$  where  $A_i, D_i$  are the  $i$ -th columns of  $A, D$ .

REMARK 2. Problem (24) is a convex optimization problem whose complexity depends on the perspective of  $f$ . If  $f$  is positively homogeneous, the perspective function is  $z_0 f(\frac{z}{z_0}) = f(z)$  which in turn makes problem (24) easier (without variable  $r$ ). ■

REMARK 3. When problem (24) is hard to solve, the adversarial approach (Bienstock and Özbay 2008) could be a valuable alternative, which avoids directly solving this problem. This approach takes the equivalent safe approximation problem (23), replaces  $W$  with a finite set  $\overline{W}$ , and solves the resulting linear optimization problem (LP). Then, the  $(w_0 \ w^\top)^\top \in W$  values that violate each of the constraints the most are added to set  $\overline{W}$ . Iterating this procedure until there is no violation guarantees optimality at termination. Obviously, this approach does not use perspective functions, but if the number of LPs solved is large, then this also becomes a hard problem. ■

The solution of the upper bound problem can be used to obtain a (potentially good) lower bound for problem (20) by using what was proposed for two-stage fixed-recourse robust constraints by Hadjiyiannis et al. (2011) and extended by Zhen et al. (2017). Theorem 2 states that the global optimum solution of problem (20) can be obtained by solving  $\bar{x} \in \arg \sup\{(A^\top \bar{w})^\top x : x \in U\}$  where  $\bar{w} \in \text{dom } f^*$  is the scenario where the first constraint of (22) is tight. However, finding  $\bar{w}$  is as hard as solving (22) (see the proof of Theorem 2), while for any  $w \in \text{dom } f^*$ , solving  $\arg \sup\{(A^\top w)^\top x : x \in U\}$  gives us a feasible (lower bound) solution of problem (20). Hence, we generate a finite set  $\overline{W}$ , which will be used to obtain good lower bound solutions, i.e., for each  $w \in \overline{W}$  we solve:

$$x \in \arg \sup_{x \in U} \{(A^\top w)^\top x\}, \quad (25)$$

and the best solution is the one giving the best objective  $f(Ax + b)$ . The set  $\overline{W}$  can be generated in multiple ways, and we use our upper bound solution for this purpose, namely  $(\hat{u}, \hat{r}, \hat{V}, \hat{\tau})$  standing for optimal  $(u, r, V, \tau)$  values in (24). We plug this solution in the safe approximation (23) of the ARO problem (22) that is equivalent to the original convex maximization problem. Then, we generate elements of  $\overline{W}$  as the worst-case parameter realizations in each constraint of (23), independently. For example, the first element of  $\overline{W}$  is obtained from the first constraint (23a) as:

$$\overline{W}^1 \in \arg \sup_{w \in \text{dom } f^*} \left\{ \sup_{w_0 \leq -f^*(w)} d^\top (\hat{u} + \hat{V}w + \hat{r}w_0) + b^\top w + w_0 - \hat{\tau} \right\}$$

$$\begin{aligned}
&= \arg \sup_{w \in \text{dom } f^*} \left\{ \sup_{w_0 \leq -f^*(w)} (1 + d^\top \hat{r})w_0 + (d^\top \hat{V} + b^\top)w + d^\top \hat{u} - \hat{\tau} \right\} \\
&= \arg \sup_{w \in \text{dom } f^*} \left\{ -(1 + d^\top \hat{r})f^*(w) + (d^\top \hat{V} + b^\top)w + d^\top \hat{u} - \hat{\tau} \right\},
\end{aligned}$$

We thus construct  $\bar{W}$  as  $\bar{W} = \bar{W}^1 \cup [\bigcup_{i=1}^n \bar{W}_i^2] \cup [\bigcup_{i=1}^q \bar{W}_i^3]$  with:

$$\bar{W}^1 \in \arg \sup_{w \in \text{dom } f^*} \left\{ -(1 + d^\top r)f^*(w) + (d^\top \hat{V} + b^\top)w + d^\top \hat{u} - \hat{\tau} \right\}, \quad (26)$$

$$\bar{W}_i^2 \in \arg \sup_{w \in \text{dom } f^*} \left\{ (D_i^\top r)f^*(w) + (A_i^\top - D_i^\top \hat{V})w - D_i^\top \hat{u} \right\}, \quad i = 1, \dots, n \quad (27)$$

$$\bar{W}_i^3 \in \arg \sup_{w \in \text{dom } f^*} \left\{ -\hat{u}_i - \hat{V}_i w + \hat{r}_i f^*(w) \right\}, \quad i = 1, \dots, q. \quad (28)$$

In the above notation  $\bar{W}$  is a finite set of  $n + q + 1$  many  $w$  scenarios. With this approach, we aim to represent  $\bar{w}$  (the worst-case parameter realization of the original ARO problem) as accurately as possible by plugging the optimal LDR from the upper bound relaxation problem (24) and then collecting the worst-case parameter realization in each constraint of the safe approximation. We emphasize that the solution of the upper bound relaxation is directly being used in the process of obtaining a lower bound. Hence, the upper bound problem is not only being solved to get an upper bound value, but more importantly also to obtain a good solution to the original problem. The upper/lower bound approximation scheme is summarized in Appendix C of the Online Supplement.

We next derive the approximations for specific problems, namely convex quadratic maximization, convex log-sum-exp (geometric) maximization, and convex sum-of-max-linear-terms maximization. The results are shared in Tables 1, 2, and 3. Complete derivations can be found in Appendix D of the Online Supplement. In summary, we see that the upper bound approximation of the convex quadratic maximization problem is found by solving a second-order cone optimization problem, and from the solution of this problem the lower bound scenarios can be collected analytically. For geometric maximization, the upper bound problem is a convex exponential cone optimization problem, and the lower bound scenarios can also be collected by solving multiple exponential cone optimization problems. Finally, for sum-of-max-linear-terms maximization, the upper bound problem is a linear optimization problem, and the lower bound scenarios can be found analytically. All of the original problems are known to be very hard problems, while the approximation problems are mainstream convex optimization problems and there exist many powerful solvers to solve such problems. In the numerical experiments, we use Mosek as a conic optimization solver and CPLEX (IBM ILOG CPLEX 2014) as a linear optimization solver.



Problem	Parameters and Assumptions	Convex Maximization Formulation
Quadratic Maximization	$g(x) = x^\top Qx + \ell^\top x$ , convex quadratic function $Q \succeq 0$ , positive semi-definite matrix $\ell \in \mathbb{R}^n$ , linear coefficients vector	$\max_{x \in \mathbb{R}_+^n} g(x) = x^\top Qx + \ell^\top x$ s. t. $Dx \leq d$
Geometric Maximization	$f(z) = \log(\sum_{i=1}^m \exp(z_i))$ , log-sum-exp function $A \in \mathbb{R}^{m \times n}$ , linear coefficients matrix $b \in \mathbb{R}^m$ , linear constants vector	$\max_{x \in \mathbb{R}_+^n} f(Ax + b) = \log(\sum_{i=1}^m \exp(A_{(i)}x + b_i))$ s. t. $Dx \leq d$
Sum-of-Max-Linear-Terms Optimization	$f(z) = \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{z_j\}$ , sum-of-max-terms function $K > 0$ , number of max-terms $\mathcal{I}_k \subseteq \{1, \dots, m\}$ , indexes for $k$ -th max term $\mathcal{I}_k \cap \mathcal{I}_\ell = \emptyset$ , for $k \neq \ell$ , w.l.o.g. $\cup_{k=1}^K \mathcal{I}_k = \{1, \dots, m\}$ , w.l.o.g. $A \in \mathbb{R}^{m \times n}$ , linear coefficients matrix $b \in \mathbb{R}^m$ , linear constants vector	$\max_{x \in \mathbb{R}_+^n} f(Ax + b) = \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{(j)}x + b_j\}$ s. t. $Dx \leq d$ Note: By using logical programming one can reformulate as mixed integer optimization

**Table 1** Quadratic, Geometric, and Sum-of-Max-Linear-Terms Maximization problems. In each case we introduce the problem setting, and share the main convex maximization problem that we are interested in solving.

All problems have the same linear constraints  $Dx \leq d$  for  $D \in \mathbb{R}^{q \times n}$ ,  $d \in \mathbb{R}^q$ .

Problem	Upper Bound Problem	Problem Type and Variables
Quadratic Maximization	$\inf \tau$ s. t. $d^\top u + \bar{v}^\top d - (1 + \tau)/2 + \left\  \begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + (1 - \tau)/2 \end{pmatrix} \right\ _2 \leq 0$ $-D_i^\top u + \frac{\ell_i}{2} - \bar{v}^\top D_i + \left\  \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - \hat{v}^\top D_i \end{pmatrix} \right\ _2 \leq 0, \quad i = 1, \dots, n$ $-u_i - \bar{v}_i + \left\  \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \end{pmatrix} \right\ _2 \leq 0, \quad i = 1, \dots, q$	Second-order Cone Optimization Problem $\tau \in \mathbb{R}, u \in \mathbb{R}^n, \bar{v} \in \mathbb{R}^q, \hat{v} \in \mathbb{R}^q, \tilde{V} \in \mathbb{R}^{q \times m}$ , $L$ is obtained by decomposition $Q = L^\top L$
Geometric Maximization	$\inf \tau$ s. t. $1 + d^\top r \geq \sum_{j=1}^m z_j^{(1)}$ $(z_j^{(1)}, 1 + d^\top r, (V_j^\top d + b_j - \tau + d^\top u)) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m$ , $-D_i^\top r \geq \sum_{j=1}^m z_{ij}^{(2)}, \quad i = 1, \dots, n$ $(z_{ij}^{(2)}, -D_i^\top r, (A_{i,(j)} - V_j^\top D_i - D_i^\top u)) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m, \quad i = 1, \dots, n$ $-r_i \geq \sum_{j=1}^m z_{ij}^{(3)}, \quad i = 1, \dots, q$ $(z_{ij}^{(3)}, -r_i, (-V_{j,(i)} - u_i)) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m, \quad i = 1, \dots, q$	Exponential Cone Optimization Problem $r \in \mathbb{R}^q, u \in \mathbb{R}^n, V \in \mathbb{R}^{q \times m}, \tau \in \mathbb{R}$ , $z^{(1)} \in \mathbb{R}^m, z^{(2)} \in \mathbb{R}^{n \times m}, z^{(3)} \in \mathbb{R}^{q \times m}$ , $\mathcal{K}_{\text{exp}}$ denotes the exponential cone
Sum-of-Max-Linear-Terms Optimization	$\inf \tau$ s. t. $d^\top u + \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{V_j^\top d + b_j\} \leq \tau$ $-D_i^\top u + \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{i,(j)} - V_j^\top D_i\} \leq 0, \quad i = 1, \dots, n$ $-u_i + \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{V_{j,(i)}\} \leq 0, \quad i = 1, \dots, q$	Linear Optimization Problem $u \in \mathbb{R}^n, V \in \mathbb{R}^{q \times m}, \tau \in \mathbb{R}$ , for linearity use auxiliary variables

**Table 2** For each of the three problems, we show the upper bound approximation problems. These problems are special forms of upper bound problem (24) of Theorem 3.

Problem	Lower Bound Scenarios	Problem Type
Quadratic Maximization	$\bar{W}^1 = \left[ h \left( \begin{pmatrix} \tilde{V}^\top d \\ d^\top \hat{v} + (1-\tau)/2 \\ 1 \end{pmatrix} \right) \right]$ $\bar{W}_i^2 = \left[ h \left( \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - D_i^\top \hat{v} \\ 1 \end{pmatrix} \right) \right], \quad i = 1, \dots, n$ $\bar{W}_i^3 = \left[ h \left( \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \\ 1 \end{pmatrix} \right) \right], \quad i = 1, \dots, q,$ <p>where <math>h(\cdot)</math> normalizes its input.</p>	Analytic Scenarios
Geometric Maximization	$\bar{W}^1 \in \arg \sup_{w, t \in \mathbb{R}^m} (1 + d^\top r) (\sum_{j=1}^m t_j) + (d^\top V + b^\top) w + d^\top u - \tau$ <p>s. t. <math>(1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m</math>  <math>\sum_{j=1}^m w_j = 1</math></p> $\bar{W}_i^2 \in \arg \sup_{w, t \in \mathbb{R}^m} (-D_i^\top r) (\sum_{j=1}^m t_j) + (A_i^\top - D_i^\top V) w - D_i^\top u$ <p>s. t. <math>(1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m</math> <span style="float: right;"><math>i = 1, \dots, n</math></span>  <math>\sum_{j=1}^m w_j = 1</math></p> $\bar{W}_i^3 \in \arg \sup_{w, t \in \mathbb{R}^m} (-r_i) (\sum_{j=1}^m t_j) + (-V_{(i)}) w - u_i$ <p>s. t. <math>(1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m</math> <span style="float: right;"><math>i = 1, \dots, q</math></span>  <math>\sum_{j=1}^m w_j = 1</math></p>	Each scenario collection is an exponential cone problem
Sum-of-Max-Linear-Terms Optimization	$\bar{W}^1 \in \arg \sup_w d^\top (u + Vw) + b^\top w - \tau$ <p>s. t. <math>w_j \geq 0, \quad j = 1, \dots, m</math>  <math>\sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K</math></p> $\bar{W}_i^2 \in \arg \sup_w A_i^\top w - D_i^\top (u + Vw)$ <p>s. t. <math>w_j \geq 0, \quad j = 1, \dots, m</math> <span style="float: right;"><math>i = 1, \dots, n</math></span>  <math>\sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K</math></p> $\bar{W}_i^3 \in \arg \sup_w -u_i - V_{(i)} w$ <p>s. t. <math>w_j \geq 0, \quad j = 1, \dots, m</math> <span style="float: right;"><math>i = 1, \dots, q</math></span>  <math>\sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K</math></p>	Each optimization problem can be split to $K$ optimization problems that are analytically solvable

**Table 3** For each of the three problems, we share the lower bound scenario collection steps. As discussed before, these scenarios will help us to find a lower bound solution by solving linear optimization problems corresponding to each scenario.

## 4. Numerical Experiments

In this section we present numerical experiments to support the theory developed in this paper. We use YALMIP through MATLAB 2018b (MATLAB 2018) to call various solvers, and report the solver times below (we do not reflect the problem formulation times in YALMIP). Numerical experiments are obtained by using a standard personal computer with an 8-th Generation Intel(R) Core(TM) i7-8750H processor. The details of the data and codes of these experiments are shared in Appendix G of the Online Supplement.

### 4.1. Log-Sum-Exp Maximization over a 2-norm Constraint

Consider problem (1) with  $g(x) := \|x - a\|_2$ . In Section 2 under Corollary 1 we discussed that for  $f(z) = \log(\sum_{i=1}^m \exp(z_i))$ , the upper bound approximation problem is exponential-cone representable, and we can analytically obtain the lower bound solution immediately. The approximation problem (10) is solved by MOSEK.

To benchmark our approximation, we used general-purpose nonlinear optimization solvers Artelys Knitro (Byrd et al. 2006), IPOPT (Wächter and Biegler 2006), and BMIBNB of YALMIP (Löfberg 2004). Knitro appeared to yield the best result, so we report Knitro as a benchmark. Although Knitro does not guarantee global optimality it can find the global optimum value faster and more often compared to the other solvers we tried. We also compare our approximation with the global optimization solver BARON. The results can be found in Table 4.

Problem		Knitro		Knitro Multi-Start			BARON		Approximation		
		value	time	value	start	time	value	time	upper	lower	time
# 1	( $n = 5, m = 5$ )	9.7890	0.05	-	-	-	9.7890	0.40	10.0186	9.7816	0.22
# 2	( $n = 20, m = 15$ )	54.0329	0.18	55.8913	2	0.43	25.0731	1,800.00	56.0521	55.7287	0.22
# 3	( $n = 100, m = 120$ )	241.1606	1.61	-	-	-	NA	1,800.00	241.1606	241.1606	0.23
# 4	( $n = 20, m = 40$ )	156.6875	0.40	179.1224	5	5.52	NA	1,800.00	179.1224	179.1224	0.24
# 5	( $n = 50, m = 100$ )	324.8785	1.17	370.9066	3	6.55	NA	1,800.00	370.9066	370.9066	0.27
# 6	( $n = 100, m = 100$ )	428.7200	2.37	472.0475	2	8.86	NA	1,800.00	472.0475	472.0475	0.25
# 7	( $n = 200, m = 30$ )	551.3160	0.68	570.8467	20	18.45	NA	1,800.00	570.8467	570.8467	0.23
# 8	( $n = 400, m = 80$ )	570.7248	3.51	601.6886	4	14.35	NA	1,800.00	601.6886	601.6886	0.25
# 9	( $n = 50, m = 20$ )	error	0.62	-	-	-	error	1,800.00	890.7722	890.7722	0.24
# 10	( $n = 10,000, m = 100$ )	NA	1,800.00	-	-	-	-	-	456.6373	455.5054	0.29
# 11	( $n = 1,000, m = 1,000$ )	156.4414	1311.00	-	-	-	-	-	178.4665	141.9482	0.59
# 12	( $n = 2,000, m = 700$ )	238.4521	1,800.00	-	-	-	-	-	325.0580	311.7564	0.51

**Table 4 Comparison of our approximation method with Knitro and BARON for the 2-norm constrained log-sum-exp maximization problem. The first column gives the problem numbers, and gives the dimension of these problems (recall the decision variable  $x$  is an  $n$ -dimensional vector and the function  $f(Ax + b)$  takes  $m$ -dimensional vector input, hence  $A$  is a matrix of size  $m \times n$ ). The ‘value’ column stands for the best objective value computed by the corresponding solver, and ‘time’ stands for how many seconds it took for the solver to compute this. ‘start’ means the minimum number of starting points for Knitro to find the corresponding value. The column ‘upper’ is our upper bound approximation, and ‘lower’ is the lower bound approximation. We allow 30-minutes for solvers, hence ‘NA’ means that the solver cannot find any feasible point in the time limit, otherwise the best solution found is written. The entry ‘error’ means that there are numerical errors and the solver terminates.**

These results show that BARON can only find the global optimum value (and guarantee optimality) in the first problem, which is the smallest one. For the other problems BARON is not usable, which is mainly due to the fact that the log-sum-exp function is highly nonlinear as well as evaluating it is a hard task due to the exponential-terms. Moreover, we see that our approximation method (except Problems 1 and 2) is faster than Knitro without multi-start which is designed to find a local optimum. For Problems 4-7 Knitro cannot find the global optimum without using the multi-start option (which needs manual tuning) and there is no guarantee of global optimality at the end, while our method solves the problem with a global optimality guarantee (upper bounds are equal to lower bounds). For Problem 1 and Problem 2 we cannot find the global optimum value, which shows our method does not necessarily find the global optimum in all of the cases. Problem 9 is a problem where the exponential summands in the logarithm operator get very large and the computers accept these terms as infinity, so the solvers cannot compute any solution, and they do not automatically scale the problem. However, since our upper bound approximation does not compute the log-sum-exp function directly, it does not suffer from such numerical issues. To compute a lower bound we need function evaluations and to address the numerical issues we automatically scale the problem. Problem 10 has 10,000 variables, so none of the solvers can find an initial solution within 1,800 seconds, whereas our method finds the upper and lower bounds within 0.29 seconds. Problem 11 has  $m = 1,000$  and  $n = 1,000$ . This problem has a larger approximation gap, which may be a result of the increasing dimension (recall that the proposed approximation problem has  $m$  variables). Finally, Problem 12 has a large dimension in terms of  $m$  and  $n$ . Knitro returns a feasible solution within 1,800 seconds but it is not a local optimum (large  $n$  increases the dimension of the main problem). We also have a gap between the upper and lower bound approximations in this problem.

#### 4.2. Log-Sum-Exp Maximization over an $\infty$ -norm Constraint

We consider problem (1) with  $g(x) := \|x - a\|_\infty$ . In Corollary 2 we discussed that we can find the global optimum value of problem (1) by solving a mixed-integer convex optimization problem. In light of our findings, when  $f$  is a log-sum-exp function, we can represent this problem as a mixed-integer exponential-cone optimization problem and MOSEK can solve these types of problems efficiently. We use the same solvers as in the previous subsection as benchmarks. In this problem, we find the exact global optimum value, i.e., we do not approximate this value. The solutions that attain the global optimum values are also

efficiently retrieved, as problem (7) is maximizing a linear function over box constraints. The results are presented in Table 5.

Problem		Knitro		Knitro Multi-Start			BARON		Exact	
		value	time	value	start	time	value	time	value	time
# 1	$(n = 5, m = 5)$	9.10	0.03	15.00	2	0.04	15.00	0.06	15.00	0.28
# 2	$(n = 20, m = 5)$	46.34	0.04	54.50	3	0.11	23.50	1,800.00	54.50	0.39
# 3	$(n = 50, m = 20)$	NA	0.20	-	-	-	NA	1,800.00	1,723.30	48.00
# 4	$(n = 180, m = 20)$	54.39	0.50	57.70	15	6.17	NA	1,800.00	57.70	149.80
# 5	$(n = 30, m = 300)$	35.76	0.50	39.62	30	33.91	NA	1,800.00	39.62	18.37

**Table 5** Comparison of our exact method with Knitro and BARON for  $\infty$ -norm constrained log-sum-exp maximization. The meanings of common columns are the same as Table 4, and similarly time is in seconds. Here, ‘Exact’ section stands for our method which computes exactly the global optimum value as shown in Corollary 2.

In general, our method’s computation time is higher compared to the previous problem, mainly because we solve a mixed-integer exponential cone optimization problem. However, contrary to Knitro (even with multi-start), our method provides a global optimality guarantee. Similarly to the previous example, BARON is not scalable for larger problems.

### 4.3. Convex Quadratic Maximization over Linear Constraints

We consider the first problem type from Table 1, e.g., maximizing a convex quadratic function with respect to linear constraints. The state-of-the-art solver for convex quadratic maximization is CPLEX (version 12.6 onwards), which uses a branch-and-bound method based on McCormick relaxations and SDP cuts. The algorithm terminates at a global optimum, but as for any branch and bound algorithm, this may take an exponential number of steps (Boyd and Mattingley 2007). On the one hand, our upper bound approximation method solves a second-order cone optimization problem, so we use MOSEK for this purpose. On the other hand, our lower bound approximation method solves multiple linear optimization problems after collecting lower bound scenarios analytically, hence we use CPLEX for this purpose. The results are given in Table 6.

Problems 1-3 are small-sized problems where we see that CPLEX finds the global optimum faster than our approximation method’s total time. In Problems 4-5 CPLEX finds the global optimum slower than our approximation, while with a time limit of the time our

Problem	CPLEX			Upper Bound		Lower Bound	
	value	time	restricted	value	time	value	time
# 1 ( $n = 20, q = 10$ )	394.75	0.08	394.75	702.05	0.43	394.75	0.16
# 2 ( $n = 20, q = 10$ )	884.75	0.10	884.75	1,192.10	0.43	884.75	0.15
# 3 ( $n = 10, q = 15$ )	4,674.70	0.15	4,674.70	4,753.10	0.01	4,674.70	0.17
# 4 ( $n = 50, q = 62$ )	175,710.00	0.81	175,710.00	177,640.00	0.29	175,710.00	0.02
# 5 ( $n = 100, q = 130$ )	692,610.00	16.50	692,610.00	707,520.00	2.30	692,610.00	1.40
# 6 ( $n = 200, q = 240$ )	6,020,800.00	466.00	6,020,800.00	6,100,200.00	25.00	6,020,800.00	3.40
# 7 ( $n = 240, q = 280$ )	7,303.00	3,600.00	7,303.00	1,961,700.00	31.21	1,855,700.00	4.18

**Table 6** Comparison of our approximation method and CPLEX with multiple linear constrained convex quadratic maximization. Here the column names that are used in the previous tables have the same meaning, and time is similarly in seconds. Recall that  $q$  is the number of (linear) constraints. The ‘Upper Bound’ section is our proposed upper bound, and ‘Lower Bound’ is the proposed lower bound. In ‘restricted’ column of CPLEX, we are giving a time limit of the total time needed by our approximation method (upper + lower) and record the best CPLEX can find. Note that when the ‘restricted’ case finds the same value as its ‘value’ (e.g., the unlimited time case), it means CPLEX finds the global optimum solution but cannot guarantee global optimality yet.

approximation takes CPLEX still finds the global optimum, but cannot guarantee global optimality. A similar argument holds for Problem 6, but the run-time of CPLEX increases significantly. Finally in Problem 7, CPLEX cannot find the global optimum solution in an hour (3,600 seconds). In general, we see that our lower bound values are all equal to the global optimum value (except for Problem 7) with a reasonable run-time, however we cannot guarantee global optimality or give tight upper bounds, unlike in our other experiments.

#### 4.4. Log-Sum-Exp Maximization over Linear Constraints

We consider the second problem type from Table 1, where we are interested in maximizing a log-sum-exp function over linear constraints. We use the state-of-the-art general purpose global optimization solver BARON as the main solver. We also use Knitro as a local optimization solver. Since our upper and lower bound problems are exponential cone representable problems, we use MOSEK solver for solving this problem. The results can be found in Table 7.

Except for Problems 2 and 6, we find global optimum upper and lower bounds, enabling us to guarantee global optimality. For Problems 2 and 6 we obtain tight upper bounds, and our lower bounds are globally optimal. Our upper bound approximation problem works

Problem	Knitro		Knitro Multi-Start			Baron		Upper Bound		Lower Bound	
	value	time	value	start	time	value	time	value	time	value	time
# 1 (size= 10)	16.8763	0.05	35.2008	3	0.10	35.2008	0.32	35.2008	0.05	35.2008	0.01
# 1 (size= 40)	248.7589	0.29	-	-	-	248.7589	0.36	248.7589	1.13	248.7589	0.01
# 1 (size= 60)	282.6632	0.48	386.0733	5	1.95	384.6207	114.38*	386.0733	2.95	386.0733	0.01
# 1 (size= 100)	667.0963	2.19	676.8081	20	37.77	547.8524	10.94*	676.8080	9.98	676.8080	0.01
# 2 (size= 20)	64.5087	0.10	64.8926	10	0.43	64.6196	0.28*	65.2268	0.37	64.8926	0.30
# 3 (size= 50)	102.5824	0.32	145.3380	50	11.24	144.1676	0.64*	145.3380	9.73	145.3380	0.01
# 4 (size= 100)	166.8399	1.85	176.1074	50	77.05	166.8399	3.17*	176.1074	241.57	176.1074	0.01
# 5 (size= 10)	34.8293	0.05	45.0356	3	0.10	45.0356	1,000.00	45.0356	0.18	45.0356	0.01
# 6 (size= 30)	74.8013	0.17	76.0362	15	1.86	76.0362	0.72*	76.1283	14.71	76.0362	4.37

**Table 7 Comparison of our approximation method with Knitro and BARON. Size of the problem means  $n = q = m$  and these are equal to the given size, i.e., the number of variables, number of constraints, and input dimension of the log-sum-exp function are equal. A star (\*) next to the solution-time of BARON means the solver encountered numerical issues and it returns the best solution in the reported time.**

fast, however it is an exponential cone problem with  $2q + qm$  variables in total. Depending on the size of the problem, our approximation may get slower as shown in Problem 4. However, in such large-scale problems Knitro also needs a lot of multiple starting points, and as it cannot find an upper bound to the solution obtained, it cannot guarantee optimality. Knitro needs multi-start to find the global optimum in all of the problems except for Problem 1 with  $n = 40$ . Also, it should be noted that BARON cannot guarantee global optimality in several cases. This is because BARON has numerical difficulties in these problems.

#### 4.5. Sum-of-Max-Linear-Terms Maximization over Linear Constraints

Finally, we consider the third problem type from Table 1. For this problem, we solve our approximation problems (which are all linear problems) with CPLEX. We also compare our approximations with directly solving this convex maximization problem via GUROBI version 9.0 (Gurobi Optimization 2018). Although GUROBI does not solve sum-of-max-linear-terms maximization problems explicitly, YALMIP gives the mixed integer linear optimization problem reformulation of this problem successfully (by applying logical programming). The reason we chose GUROBI as a benchmark solver is because it gives a better performance than CPLEX in our numerical experiments. Therefore, we compare

our linear approximation problems to the performance of GUROBI on mixed integer optimization to solve the problem to global optimality. Recall that mixed integer optimization is NP-hard (Nemhauser and Wolsey 1988). The results are given in Table 8.

Problem	GUROBI			Upper Bound		Lower Bound	
	value	time	restricted	value	time	value	time
# 1 ( $n = 5,  \mathcal{I}_k  = 5, K = 1$ )	23.29	0.01	23.29	23.29	0.01	23.29	0.07
# 2 ( $n = 5,  \mathcal{I}_k  = 5, K = 10$ )	233.94	0.10	233.94	233.94	0.01	233.94	0.08
# 3 ( $n = 20,  \mathcal{I}_k  = 10, K = 10$ )	1,081.60	7.18	1,055.20	1,169.40	0.20	1,053.10	0.62
# 4 ( $n = 30,  \mathcal{I}_k  = 20, K = 20$ )	4,105.10	1,000.00*	3,970.80	4,499.10	0.72	3,976.00	1.71
# 5 ( $n = 100,  \mathcal{I}_k  = 40, K = 30$ )	22,224.00	1,000.00*	21,129.90	27,939.00	16.69	21,751.00	4.46
# 6 ( $n = 200,  \mathcal{I}_k  = 50, K = 50$ )	54,213.00	1,000.00*	54,213.00	83,466.00	146.22	61,794.00	4.46
# 7 ( $n = 10,  \mathcal{I}_k  = 5, K = 2$ )	113.71	0.08	113.71	113.71	0.01	113.71	0.15
# 8 ( $n = 10,  \mathcal{I}_k  = 50, K = 50$ )	479.50	1,000.00*	479.50	3,052.90	1.37	3,052.90	0.27
# 9 ( $n = 30,  \mathcal{I}_k  = 50, K = 50$ )	1,877.30	1,000.00*	539.10	2,894.60	42.38	2,783.50	1.27
# 10 ( $n = 50,  \mathcal{I}_k  = 60, K = 60$ )	1,673.70	1,000.00*	1,673.70	3,323.50	356.23	2,997.40	0.57
# 11 ( $n = 20,  \mathcal{I}_k  = 10, K = 10$ )	3,002.40	93.85	2,401.10	3,032.00	0.47	3,002.40	0.55
# 12 ( $n = 20,  \mathcal{I}_k  = 50, K = 10$ )	3,114.20	1,000.00*	96.80	3,452.20	0.43	3,349.00	0.39
# 13 ( $n = 20,  \mathcal{I}_k  = 100, K = 50$ )	1,091.80	1,000.00*	489.60	17,568.00	10.06	17,062.00	1.13

**Table 8 Comparison of our approximation method with GUROBI for multiple linear constrained sum-of-max-linear-terms maximization. The column descriptions are the same as the previous tables. The size of the problem is defined by  $n$  (number of variables),  $\mathcal{I}_k$  (number of elements in the set of each max-term), and  $K$  (number of max-terms). GUROBI solves each problem’s mixed integer linear optimization reformulation, hence for large sized problems this can take a vast amount of time. Therefore we give 1,000 seconds time limits for each run of GUROBI. The ‘value’ column of GUROBI shows the value computed by GUROBI within 1,000 seconds, thus if the time is 1,000\*, it means GUROBI cannot compute the global optimum solution within the time limit and returns the best solution found. The ‘restricted’ column gives the best value GUROBI can compute within a limitation of the time that it takes for our method to find the upper and lower bound values.**

Our method does not necessarily yield a global optimum contrary to solving the mixed-integer optimization reformulation of the problem, however, in large problems our method converges to a lower bound solution very swiftly. On the other hand, GUROBI suffers severely from the curse of dimensionality in mixed integer optimization. Therefore, we see that in bigger problems our lower bound quality is better than the best solution found by GUROBI, while in small-sized problems the solver converges to global optima very



quickly. Our method particularly stands out in terms of the speed. We find the global optimum (and certify optimality) in Problems 1,2,7,8. Note that in Problem 1 the global optimality of our approximation is not a coincidence because in this problem we have  $K = 1$ , and we know linear decision rules are optimal in this case (Ardestani-Jaafari and Delage 2016). Moreover, GUROBI cannot find the global optimum in the allowed 1,000 seconds for Problems 4,5,6,8,9,10,12,13. Except for Problems 3,4,5, our lower bound solutions are better than the ones found by GUROBI, or equal with a faster computation time. In Problem 3, GUROBI finds a better value than our lower bound in the same time (time restricted case). In the largest problems, namely 6 and 13, the best solutions found by GUROBI in 1,000 seconds are considerably lower than our lower bound solutions, where our method computes the upper and lower bounds within 150 and 11 seconds, respectively.

## 5. Conclusions

Maximizing a convex function over convex constraints is known to be a hard problem even in its simplest cases. One can either try to solve the problem globally, or aim to obtain a good lower bound solution. We show that, as the size of the convex maximization problem gets larger, the local optimization solvers return solutions with large global optimality gaps, and in many cases the global optimization solvers cannot terminate.

In this work, we show how to use adjustable robust optimization techniques to solve the convex maximization problem. The main idea is to transfer the difficulty of the main non-convex problem to the nonlinearity of an equivalent (convex) ARO problem. Exploiting the rich ARO literature gives us strong methods to tightly approximate the convex maximization problem efficiently. More specifically, we show convex maximization problems whose ARO reformulations can be simplified by eliminating the adjustable variables, or problems where the adjustable variables can be restricted by using decision rules. The subsequent applications of robust optimization techniques approximate the resulting problem. Furthermore, we explicitly derive tractable upper and lower bound approximation problems of some well-known convex maximization problems. By using similar techniques, we also show a class of convex maximization problems which can be reformulated as mixed-integer convex optimization problems; although the latter is still NP-hard, the existence of powerful solvers makes these reformulations more practical to solve.

Since we approximate the equivalent ARO problem of the convex maximization problem, the gap between the upper and lower bounds we propose can be explained by the ARO

theory. For instance, in the problems where linear decision rules are used to restrict the adjustable variables, the gap of the approximations can be explained by the performance of the linear decision rules. This theoretical bridge allows us to guarantee global optimality of the proposed approximation in some (easy) convex maximization problems whose ARO reformulations admit linear decision rules optimally.

The numerical experiments demonstrate the efficiency and strength of the proposed approximation. To be more specific, maximizing the log-sum-exp function over a ball constraint is approximated in less than half a second, where most of the problems are approximated without any gap. If the constraints are instead box constraints, the problem can be represented as a mixed-integer exponential cone optimization problem, and we can solve this problem to global optimality while the global optimization solvers fail to solve the original problem in most cases. We explicitly work with the maximization of convex quadratic, log-sum-exp, and sum-of-max-terms functions over linear constraints. Regarding the first case, we show that almost optimal lower bounds can be obtained very swiftly even for large problems where global optimization solvers cannot find a solution better than the starting point. For the second case, we find global lower and upper bounds in almost all problems, with the exceptions being approximated tightly, even for problems where the solvers face numerical difficulties due to the nature of geometric maximization. For the third case, as both of our upper and lower bound approximations are obtained via linear optimization, we find tight approximations swiftly, while the solvers are successful in small-sized problems but fail termination in large problems.

There are two directions for future work: application and theory. For the former one, the proposed approximation methods can be applied to one of the numerous real-life convex maximization problems. Applying these approximations in the convex maximization step of the convex-concave method can be useful for DC programming. Another implementation can be integrating our methodology with the global optimization solvers, where the solvers can first obtain a good lower bound by using our solution (along with the corresponding upper bound), and then iterate to find a globally optimal solution. As for theory, one can use piecewise-linear decision rules or nonlinear decision rules to restrict the adjustable variables in the proposed ARO reformulation. We do not give theoretical guarantees of the approximation gaps, hence developing such guarantees is essential. Furthermore, one can develop an algorithm by dividing the feasible region of the convex maximization problem into parts and solve our approximations in each part, which may give tighter bounds.

## Acknowledgments

The research of Aras Selvi was funded by Center of Economic and Business Research (CentER) of Tilburg University during his time there. Some of this work was done when Dick den Hertog was with Tilburg University. The authors would like to thank the editors and the anonymous reviewers for their valuable input.

## References

- Andrianova AA, Korepanova AA, Halilova IF (2016) One algorithm for branch and bound method for solving concave optimization problem. *IOP Conference Series: Materials Science and Engineering* 158:012005, URL <http://dx.doi.org/10.1088/1757-899x/158/1/012005>.
- Ardestani-Jaafari A, Delage E (2016) Robust optimization of sums of piecewise linear functions with application to inventory problems. *Operations Research* 64(2):474–494.
- Audet C, Hansen P, Savard G (2005) *Essays and surveys in global optimization*. GERAD (Springer).
- Benson HP (1995) Concave minimization: theory, applications and algorithms. *Handbook of global optimization*, 43–148 (Springer).
- Bienstock D, Özbay N (2008) Computing robust basestock levels. *Discrete Optimization* 5(2):389–414.
- Boyd S, Kim SJ, Vandenberghe L, Hassibi A (2007) A tutorial on geometric programming. *Optimization and Engineering* 8(1):67–127.
- Boyd S, Mattingley J (2007) Branch and bound methods. [https://stanford.edu/class/ee364b/lectures/bb\\_notes.pdf](https://stanford.edu/class/ee364b/lectures/bb_notes.pdf).
- Boyd S, Vandenberghe L (2004) *Convex optimization* (Cambridge University Press).
- Byrd RH, Nocedal J, Waltz RA (2006) Knitro: An integrated package for nonlinear optimization. *Large-scale nonlinear optimization*, 35–59 (Springer).
- Duffin RJ (1967) *Geometric programming - theory and application* (SIAM).
- Enkhbat R, Barsbold B, Kamada M (2006) A numerical approach for solving some convex maximization problems. *Journal of Global Optimization* 35(1):85–101.
- Goemans MX, Williamson DP (1994) .879-approximation algorithms for MAX CUT and MAX 2SAT. *Proceedings of the Twenty-sixth Annual ACM Symposium on Theory of Computing*, 422–431 (New York, NY, USA), URL <http://dx.doi.org/10.1145/195058.195216>.
- Gurobi Optimization L (2018) Gurobi optimizer reference manual. URL <http://www.gurobi.com>.
- Hadjiyiannis MJ, Goulart PJ, Kuhn D (2011) A scenario approach for estimating the suboptimality of linear decision rules in two-stage robust optimization. *2011 50th IEEE Conference on Decision and Control and European Control Conference*, 7386–7391 (IEEE).
- Horst R, Phong TQ, Thoai NV, de Vries J (1991) On solving a DC programming problem by a sequence of linear programs. *Journal of Global Optimization* 1(2):183–203.

- Horst R, Thoai NV (1999) DC programming: overview. *Journal of Optimization Theory and Applications* 103(1):1–43.
- IBM ILOG CPLEX (2014) V12.6: User’s Manual for CPLEX. [https://www.ibm.com/support/knowledgecenter/SSSA5P\\_12.6.2/ilog.odms.studio.help/pdf/usrcplex.pdf](https://www.ibm.com/support/knowledgecenter/SSSA5P_12.6.2/ilog.odms.studio.help/pdf/usrcplex.pdf).
- Le Thi HA, Pham Dinh T (2018) DC programming and DCA: thirty years of developments. *Mathematical Programming* 169(1):5–68.
- Lipp T, Boyd S (2016) Variations and extension of the convex–concave procedure. *Optimization and Engineering* 17(2):263–287.
- Löfberg J (2004) YALMIP: A toolbox for modeling and optimization in MATLAB. *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No. 04CH37508)*, 284–289 (IEEE).
- Löfberg J (2016) YALMIP tutorial: Nonlinear operators - integer models. <https://yalmip.github.io/tutorial/nonlinearoperatorsmixedinteger/>.
- Mangasarian OL (1996) *Machine learning via polyhedral concave minimization*. (Heidelberg: Physica-Verlag HD).
- Mangasarian OL (2015) Unsupervised classification via convex absolute value inequalities. *Optimization* 64(1):81–86.
- MATLAB (2018) *version 9.5.0 (R2018b)* (Natick, Massachusetts: The MathWorks Inc.).
- MOSEK ApS (2019a) Modeling cookbook. <https://docs.mosek.com/MOSEKModelingCookbook-letter.pdf>.
- MOSEK ApS (2019b) *The MOSEK optimization toolbox for MATLAB manual. Version 9.0*. Available at <http://docs.mosek.com/9.0/toolbox/index.html>.
- Nemhauser GL, Wolsey LA (1988) *Integer and combinatorial optimization* (New York, NY, USA: Wiley-Interscience), ISBN 0-471-82819-X.
- Pardalos PM, Rosen JB (1986) Methods for global concave minimization: A bibliographic survey. *SIAM Review* 28(3):367–379.
- Pardalos PM, Schnitger G (1988) Checking local optimality in constrained quadratic programming is NP-hard. *Operations Research Letters* 7(1):33–35.
- Rebennack S, Nahapetyan A, Pardalos PM (2009) Bilinear modeling solution approach for fixed charge network flow problems. *Optimization Letters* 3(3):347–355.
- Rockafellar RT (1970) *Convex analysis*. Princeton Mathematical Series (Princeton, N. J.: Princeton University Press).
- Roos E, Den Hertog D, Ben-Tal A, De Ruiter F, Zhen J (2018) Tractable approximation of hard uncertain optimization problems. *Preprint 6679: Optimization Online* URL [http://www.optimization-online.org/DB\\_HTML/2018/06/6679.html](http://www.optimization-online.org/DB_HTML/2018/06/6679.html).

- Selvi A, Den Hertog D, Wiesemann W (2020) A reformulation-linearization technique for optimization over simplices. *Preprint 8098: Optimization Online* URL [http://www.optimization-online.org/DB\\_HTML/2020/11/8098.html](http://www.optimization-online.org/DB_HTML/2020/11/8098.html).
- Serrano SA (2015) *Algorithms for unsymmetric cone optimization and an implementation for problems with the exponential cone*. Ph.D. thesis, Stanford University, Institute for Computational and Mathematical Engineering.
- Sherali HD, Adams WP (2013) *A reformulation-linearization technique for solving discrete and continuous nonconvex problems*, volume 31 (Springer).
- Sherali HD, Fraticelli BM (2002) Enhancing RLT relaxations via a new class of semidefinite cuts. *Journal of Global Optimization* 22(1-4):233–261.
- Tuy H (1964) Concave programming under linear constraints. *Soviet Mathematics Doklady* 5:1437–1440.
- Tuy H (1986) A general deterministic approach to global optimization via DC programming. *North-Holland mathematics studies*, volume 129, 273–303 (Elsevier).
- Tuy H, Horst R (1988) Convergence and restart in branch-and-bound algorithms for global optimization. application to concave minimization and DC optimization problems. *Mathematical Programming* 41(1-3):161–183.
- Wächter A, Biegler LT (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming* 106(1):25–57.
- Yanıkoglu İ, Gorissen BL, den Hertog D (2019) A survey of adjustable robust optimization. *European Journal of Operational Research* 277(3):799–813.
- Zass R, Shashua A (2007) Nonnegative sparse PCA. Schölkopf B, Platt JC, Hoffman T, eds., *Advances in Neural Information Processing Systems 19*, 1561–1568 (MIT Press), URL <http://papers.nips.cc/paper/3104-nonnegative-sparse-pca.pdf>.
- Zhen J, De Ruiter F, Den Hertog D (2017) Robust optimization for models with uncertain SOC and SDP constraints. *Preprint 6371: Optimization Online* URL [http://www.optimization-online.org/DB\\_HTML/2017/12/6371.html](http://www.optimization-online.org/DB_HTML/2017/12/6371.html).
- Zhen J, Den Hertog D, Sim M (2018) Adjustable robust optimization via Fourier–Motzkin Elimination. *Operations Research* 66(4):1086–1100.
- Zwart PB (1974) Global maximization of a convex function with linear inequality constraints. *Operations Research* 22(3):602–609.

## Online Supplement: Appendices

### Appendix A: RLT Relaxation of Problem (8)

Suppose we have a symmetric matrix variable  $V \in \mathbb{S}^{m \times m}$  such that  $V = ww^\top$ . The following can be applied to replace the convex term  $\|A^\top w\|_2$ :

$$\|A^\top w\|_2 = \sqrt{w^\top A A^\top w} = \sqrt{\text{tr}(w^\top A A^\top w)} = \sqrt{\text{tr}(A^\top w w^\top A)}.$$

Therefore, we use the following concave reformulation of  $\|A^\top w\|_2$ :

$$\|A^\top w\|_2 = \sqrt{\text{tr}(A^\top V A)}$$

This concave reformulation is of course based on the assumption  $V = ww^\top$ . Moreover, we use the main idea of RLT and multiply each of the original constraints  $\alpha_i^\top w - \beta_i \leq 0$ ,  $\alpha_j^\top w - \beta_j \leq 0$  to obtain:

$$(\alpha_i^\top w - \beta_i)(\alpha_j^\top w - \beta_j) \geq 0$$

$$\iff \alpha_i^\top w w^\top \alpha_j - (\beta_i \alpha_j + \beta_j \alpha_i)^\top w + \beta_i \beta_j \geq 0 \quad (29)$$

$$\iff \alpha_i^\top V \alpha_j - (\beta_i \alpha_j + \beta_j \alpha_i)^\top w + \beta_i \beta_j \geq 0. \quad (30)$$

Although  $V = ww^\top$  is assumed, it is a non-convex constraint, so we relax it as:

$$V \succeq ww^\top \iff \begin{pmatrix} V & w \\ w^\top & 1 \end{pmatrix} \succeq 0. \quad (31)$$

Thus, problem (8) is relaxed by the following convex optimization problem:

$$\begin{aligned} & \sup_{V \in \mathbb{S}^{m \times m}, w \in \mathbb{R}^m} \rho \sqrt{\text{tr}(A^\top V A)} + a^\top A^\top w + b^\top w - f^*(w) \\ & \text{s. t.} \quad \alpha_i^\top w - \beta_i \leq 0, \quad i = 1, \dots, d \\ & \quad \alpha_i^\top V \alpha_j - (\beta_i \alpha_j + \beta_j \alpha_i)^\top w + \beta_i \beta_j \geq 0, \quad i \leq j = 1, \dots, d \end{aligned} \quad (32)$$

$$\begin{pmatrix} V & w \\ w^\top & 1 \end{pmatrix} \succeq 0,$$

which concludes the proof.

### Appendix B: Proof of Theorem 3

We showed that problem (20) can be represented as ARO problem (22). As shown by Roos et al. (2018), we can lift the nonlinear term  $f^*(w)$  to the uncertainty set by introducing an auxiliary uncertain parameter  $w_0$ . Hence, the set of constraints of the ARO problem is equivalent to:

$$\forall \begin{pmatrix} w_0 \\ w \end{pmatrix} \in W, \exists \lambda \in \mathbb{R}^q : \begin{cases} d^\top \lambda + b^\top w + w_0 \leq \tau & (33a) \\ D^\top \lambda \geq A^\top w & (33b) \\ \lambda \geq \mathbf{0}, & (33c) \end{cases}$$

where we define the new uncertainty set as

$$W = \left\{ \begin{pmatrix} w_0 \\ w \end{pmatrix} \in \mathbb{R}^{m+1} : w_0 + f^*(w) \leq 0 \right\}. \quad (34)$$

A safe approximation of the constraint set is obtained by using a linear decision rule for the adjustable variable:

$$\lambda = u + Vw + rw_0,$$

where  $u \in \mathbb{R}^q$ ,  $V \in \mathbb{R}^{q \times m}$  and  $r \in \mathbb{R}^q$ . Substituting this LDR in (33a) leads to

$$\begin{aligned}
d^\top \lambda + b^\top w + w_0 &\leq \tau & \forall (w_0 \ w^\top)^\top \in W \\
\iff d^\top (u + Vw + rw_0) + b^\top w + w_0 &\leq \tau & \forall (w_0 \ w^\top)^\top \in W \\
\iff d^\top u + \begin{pmatrix} w_0 \\ w \end{pmatrix}^\top \begin{pmatrix} 1 + d^\top r \\ V^\top d + b \end{pmatrix} &\leq \tau & \forall (w_0 \ w^\top)^\top \in W \\
\iff d^\top u + \delta^* \left( \begin{pmatrix} 1 + d^\top r \\ V^\top d + b \end{pmatrix} \middle| W \right) &\leq \tau. & (35)
\end{aligned}$$

To be able to find the tractable robust counterpart of (35), we derive the support function of the new uncertainty set  $W$ , which is

$$\begin{aligned}
\delta^* \left( \begin{pmatrix} z_0 \\ z \end{pmatrix} \middle| W \right) &= \sup_{(w_0, w)^\top \in W} \{z_0 w_0 + z^\top w\} \\
&= \begin{cases} \sup_{w \in \mathbb{R}^m} \{z^\top w - z_0 f^*(w)\} & \text{if } z_0 > 0 \\ \sup_{w \in \text{dom } f^*} \{z^\top w\} & \text{if } z_0 = 0 \\ +\infty & \text{otherwise} \end{cases} \\
&= \begin{cases} z_0 f \left( \frac{z}{z_0} \right) & \text{if } z_0 \geq 0 \\ +\infty & \text{otherwise.} \end{cases} & (36)
\end{aligned}$$

Above, for the case of  $z_0 = 0$ , we use the property  $\sup_{w \in \text{dom } f^*} \{z^\top w\} = \delta^*(z | \text{dom } f^*) = \lim_{z_0 \downarrow 0} z_0 f \left( \frac{z}{z_0} \right)$ , and the result follows since for  $z_0 = 0$  we have the understanding of  $\lim_{z_0 \downarrow 0} z_0 f \left( \frac{z}{z_0} \right)$  for the perspective (Rockafellar 1970). By substituting (36) into (35) we obtain:

$$\begin{aligned}
d^\top u + \delta^* \left( \begin{pmatrix} 1 + d^\top r \\ V^\top d + b \end{pmatrix} \middle| W \right) &\leq \tau \\
\iff \begin{cases} d^\top u + (1 + d^\top r) f \left( \frac{V^\top d + b}{1 + d^\top r} \right) \leq \tau \\ 1 + d^\top r \geq 0. \end{cases}
\end{aligned}$$

Hence, by using LDRs (33a) becomes exactly (24a).

Following the same steps for (33b) yields us to (24b):

$$\begin{aligned}
D^\top \lambda &\geq A^\top w & \forall (w_0 \ w^\top)^\top \in W \\
\iff D_i^\top \lambda &\geq A_i^\top w & \forall (w_0 \ w^\top)^\top \in W, \ i = 1, \dots, n \\
\iff \begin{cases} -D_i^\top u + (-D_i^\top r) f \left( \frac{A_i - V^\top D_i}{-D_i^\top r} \right) \leq 0 \\ -D_i^\top r \geq 0 \end{cases} & \quad i = 1, \dots, n.
\end{aligned}$$

Similarly (33c) becomes (24c):

$$\begin{aligned}
\lambda &\geq 0 \\
\iff -u_i - V_{(i)} w - r_i w_0 &\leq 0 & \forall (w_0 \ w^\top)^\top \in W, \ i = 1, \dots, q \\
\iff \begin{cases} -u_i + (-r_i) f \left( \frac{-V_{(i)}^\top}{-r_i} \right) \\ -r_i \geq 0 \end{cases} & \quad i = 1, \dots, q.
\end{aligned}$$

As we use an LDR for the adjustable variable, the optimal objective value of (24) is an upper bound to (20).

### Appendix C: Upper and Lower Bound Approximation of Problem (20)

We summarize the process of finding upper and lower bounds on the global optimum objective value of problem (20).

---

#### Algorithm 1: Obtaining upper and lower bounds for problem (20)

---

**input** :  $f, A, b, U$

**output**: Upper bound value  $\hat{\tau}$ , lower bound solution  $x^*$  with value  $f(Ax^* + b)$

1. Obtain the upper bound solution by solving (24), i.e.,  $(\hat{u}, \hat{V}, \hat{r}, \hat{\tau}) \in$

$$\arg \inf_{u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times m}, r \in \mathbb{R}^q, \tau \in \mathbb{R}} \tau \text{ s. t. } \begin{cases} d^\top u + (1 + d^\top r) f\left(\frac{V^\top d + b}{1 + d^\top r}\right) \leq \tau \\ 1 + d^\top r \geq 0 \\ -D_i^\top u + (-D_i^\top r) f\left(\frac{A_i - V^\top D_i}{-D_i^\top r}\right) \leq 0 & i = 1, \dots, n \\ -D_i^\top r \geq 0 \\ -u_i + (-r_i) f\left(\frac{V_{(i)}^\top}{r_i}\right) \leq 0 & i = 1, \dots, q \\ -r_i \geq 0, \end{cases}$$

or alternatively via the adversarial approach.  $\hat{\tau}$  is an upper bound value.

2. Generate a finite set of (potential) worst-case ARO scenarios by plugging the optimal LDR back in the safe approximation of the original ARO, and by collecting worst-case scenario of each constraint, i.e.,  $\overline{W} = \overline{W}^1 \cup [\bigcup_{i=1}^n \overline{W}_i^2] \cup [\bigcup_{i=1}^q \overline{W}_i^3]$  with:

$$\begin{aligned} \overline{W}^1 &\in \arg \sup_{w \in \text{dom } f^*} \left\{ -(1 + d^\top r) f^*(w) + (d^\top \hat{V} + b^\top) w + d^\top \hat{u} - \hat{\tau} \right\}, \\ \overline{W}_i^2 &\in \arg \sup_{w \in \text{dom } f^*} \left\{ (D_i^\top r) f^*(w) + (A_i^\top - D_i^\top \hat{V}) w - D_i^\top \hat{u} \right\}, & i = 1, \dots, n \\ \overline{W}_i^3 &\in \arg \sup_{w \in \text{dom } f^*} \left\{ -\hat{u}_i - \hat{V}_i w + \hat{r}_i f^*(w) \right\}, & i = 1, \dots, q. \end{aligned}$$

3. For all  $w \in \overline{W}$ , solve the linear optimization problem:

$$x \in \arg \sup_{x \in U} \{(A^\top w)^\top x\},$$

and return  $x$  that achieves the highest  $f(Ax + b)$  as the lower bound solution of problem (20).

---



## Appendix D: Complete Derivation of Specific Problems in Section 3.2

### D.1. Quadratic Optimization

Here we consider problem (20) when the objective function is a convex quadratic function. For the problem of maximizing a convex quadratic function over a polyhedron, we can find an upper bound by solving a second-order cone optimization problem, and we can find a lower bound by solving a linear optimization problem.

Consider the convex quadratic function  $g : \mathbb{R}^n \mapsto \mathbb{R}$  defined by:

$$g(x) = x^\top Qx + \ell^\top x,$$

where  $\ell \in \mathbb{R}^n$  and  $Q$  is a symmetric positive semi-definite (psd) matrix. Maximizing this function over a polyhedral set can be written as the robust optimization problem:

$$\begin{aligned} \inf \quad & \tau \\ \text{s. t.} \quad & x^\top Qx + \ell^\top x \leq \tau, \quad \forall x \in U, \end{aligned} \quad (38)$$

where  $U = \{x \in \mathbb{R}_+^n : Dx \leq d\}$  for  $D \in \mathbb{R}^{q \times n}$ ,  $d \in \mathbb{R}^q$ . We use the conic representation of the constraints of problem (38):

$$\left\| \begin{pmatrix} (1 + \ell^\top x - \tau)/2 \\ Lx \end{pmatrix} \right\|_2 - (1 - \ell^\top x + \tau)/2 \leq 0,$$

where  $L$  is the psd decomposition  $Q = L^\top L$ . Therefore, the constraint of problem (38) can be written as a robust conic constraint. Define  $f : \mathbb{R}^{m+1} \times \mathbb{R} \mapsto \mathbb{R}$  by:

$$f \begin{pmatrix} z \\ \tilde{z} \end{pmatrix} = \|z\|_2 + \tilde{z}, \quad (39)$$

with  $z \in \mathbb{R}^{m+1}$  and  $\tilde{z} \in \mathbb{R}$ . It can be verified that  $f$  is positively homogeneous and that the conjugate of this function for  $w \in \mathbb{R}^{m+1}$ ,  $\tilde{w} \in \mathbb{R}$  is:

$$f^* \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} = \begin{cases} 0 & \text{if } \tilde{w} = 1 \text{ and } \|w\|_2 \leq 1 \\ +\infty & \text{otherwise.} \end{cases}$$

Defining

$$A = \begin{bmatrix} L \\ \ell^\top/2 \\ \ell^\top/2 \end{bmatrix}, \quad b = \begin{bmatrix} \mathbf{0} \\ (1 - \tau)/2 \\ (-1 - \tau)/2 \end{bmatrix}, \quad (40)$$

it follows that:

$$f(Ax + b) = f \left( \begin{pmatrix} Lx \\ (1 + \ell^\top x - \tau)/2 \\ (\ell^\top x - 1 - \tau)/2 \end{pmatrix} \right) = \left\| \begin{pmatrix} Lx \\ (1 + \ell^\top x - \tau)/2 \end{pmatrix} \right\|_2 - (1 - \ell^\top x + \tau)/2.$$

Hence, the constraint of problem (38) is equivalent to  $f(Ax + b) \leq 0$ . Therefore, problem (38) can be rewritten as:

$$\begin{aligned} \inf \quad & \tau \\ \text{s. t.} \quad & f \left( \begin{pmatrix} Lx \\ (1 + \ell^\top x - \tau)/2 \\ (\ell^\top x - 1 - \tau)/2 \end{pmatrix} \right) \leq 0, \quad \forall x \in U. \end{aligned} \quad (41)$$

An upper bound of this problem can now be obtained by applying Theorem 3 and exploiting the positive homogeneity of  $f$  (see Appendix E). The upper bound is the optimal value of the problem:

$$\begin{aligned}
 & \inf \tau \\
 & \text{s. t. } d^\top u + \bar{v}^\top d - (1 + \tau)/2 + \left\| \begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + (1 - \tau)/2 \end{pmatrix} \right\|_2 \leq 0 \\
 & \quad -D_i^\top u + \frac{\ell_i}{2} - \bar{v}^\top D_i + \left\| \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - \hat{v}^\top D_i \end{pmatrix} \right\|_2 \leq 0, \quad i = 1, \dots, n \\
 & \quad -u_i - \bar{v}_i + \left\| \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \end{pmatrix} \right\|_2 \leq 0, \quad i = 1, \dots, q,
 \end{aligned} \tag{42}$$

in which the variables are  $\tau \in \mathbb{R}$ ,  $u \in \mathbb{R}^q$ ,  $\bar{v} \in \mathbb{R}^q$ ,  $\hat{v} \in \mathbb{R}^q$ ,  $\tilde{V} \in \mathbb{R}^{q \times m}$ .

In order to compute a lower bound, we use the optimal solution  $(\tau, u, \bar{v}, \hat{v}, \tilde{V})$  to problem (42) by obtaining a collection of worst case scenarios  $\bar{W}$  from (26), (27), and (28). These problems can be solved analytically as explained in Appendix F. This yields the scenarios:

$$\begin{aligned}
 \bar{W}^1 &= \left[ h \left( \begin{pmatrix} \tilde{V}^\top d \\ d^\top \hat{v} + (1 - \tau)/2 \\ 1 \end{pmatrix} \right) \right] \\
 \bar{W}_i^2 &= \left[ h \left( \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - D_i^\top \hat{v} \\ 1 \end{pmatrix} \right) \right] \quad i = 1, \dots, n \\
 \bar{W}_i^3 &= \left[ h \left( \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \\ 1 \end{pmatrix} \right) \right] \quad i = 1, \dots, q,
 \end{aligned} \tag{43}$$

where  $h(a) = a/\|a\|_2$  normalizes its input. Using these worst-case scenarios, the candidate solutions  $\bar{x}^{(j)}$  are obtained by solving (25), and we can substitute them in the main objective function as  $f(A\bar{x}^{(j)} + b)$  to find the best lower bound.

## D.2. Geometric Optimization

Geometric Optimization (GO) is a class of optimization problems originally introduced by Duffin (1967). A practical tutorial can be found in the work of Boyd et al. (2007). Even though it can have many representations, we focus on the GO variant where the objective is maximizing the convex log-sum-exp objective. The log-sum-exp function  $f : \mathbb{R}^m \mapsto \mathbb{R}$  is defined as

$$f(z) = \log \left( \sum_{i=1}^m \exp(z_i) \right), \tag{44}$$

and we are interested in solving problems of the following type

$$\begin{aligned}
 & \max_x f(Ax + b) = \log \left( \sum_{i=1}^m \exp(A_{(i)}x + b_i) \right) \\
 & \text{s. t. } x \in U,
 \end{aligned} \tag{45}$$

where  $U = \{x \in \mathbb{R}_+^n : Dx \leq d\}$ . This problem may appear in robust geometric optimization problems. If one applies the adversarial approach for such problems, in the step of adding worst-case uncertainty realization to the discrete uncertainty set, one will need to maximize the convex geometric function.

The conjugate  $f^* : \mathbb{R}^m \mapsto \mathbb{R}$  of the log-sum-exp function (44) is

$$f^*(w) = \begin{cases} \sum_{i=1}^m w_i \log(w_i) & \text{if } w \in \mathbb{R}_+^m \text{ and } \sum_{i=1}^m w_i = 1 \\ \infty & \text{otherwise.} \end{cases} \tag{46}$$

We observe that  $f^*(w)$  is the negative-entropy function of  $w$  on its domain, which is a standard  $m$ -dimensional simplex. It is well known that the negative entropy is a strictly convex function. Next we show that the upper bound and lower bound approximation problems (of problem (45)) are exponential cone representable. This allows one to use the power of today's conic programming solvers, e.g., Mosek's exponential cone optimization solver. We start by introducing the exponential cone, which is the following convex subset of  $\mathbb{R}^3$ :

$$\mathcal{K}_{\text{exp}} = \{(x_1, x_2, x_3) : x_1 \geq x_2 \exp(x_3/x_2), x_2 > 0\} \cup \{(x_1, 0, x_3) : x_1 \geq 0, x_3 < 0\}.$$

So, the exponential cone is the closure of the set of points which satisfy  $x_1 \geq x_2 \exp(x_3/x_2)$ ,  $x_1, x_2 > 0$ . In the next corollary we show that the upper bound problem (24) is exponential cone representable.

**COROLLARY 4 (UPPER BOUND APPROXIMATION).** Upper bound problem (24) is exponential cone representable with the following problem with variables  $r \in \mathbb{R}^q, u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times m}, \tau \in \mathbb{R}, z^{(1)} \in \mathbb{R}^m, z^{(2)} \in \mathbb{R}^{n \times m}, z^{(3)} \in \mathbb{R}^{q \times m}$ :

$$\inf \tau \quad \text{s.t.} \quad \left\{ \begin{array}{ll} 1 + d^\top r \geq \sum_{j=1}^m z_j^{(1)} & (47\text{a}) \\ \left( z_j^{(1)}, 1 + d^\top r, (V_j^\top d + b_j - \tau + d^\top u) \right) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m & \\ -D_i^\top r \geq \sum_{j=1}^m z_{ij}^{(2)} & i = 1, \dots, n \quad (47\text{b}) \\ \left( z_{ij}^{(2)}, -D_i^\top r, (A_{i,(j)} - V_j^\top D_i - D_i^\top u) \right) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m & \\ -r_i \geq \sum_{j=1}^m z_{ij}^{(3)} & i = 1, \dots, q \quad (47\text{c}) \\ \left( z_{ij}^{(3)}, -r_i, (-V_{j,(i)} - u_i) \right) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m. & \end{array} \right.$$

■

*Proof* Roos et al. (2018) show that if a function is conically representable, so is its perspective in the same cone. Log-sum-exp is an exponential cone representable function (MOSEK ApS 2019a), and we show how to represent a convex inequality system of its perspective with exponential cones. Consider the following set of constraints:

$$\begin{cases} t \geq x_0 \log(\exp(x_1/x_0) + \dots + \exp(x_m/x_0)) \\ x_0 > 0. \end{cases} \quad (48)$$

By using the proof in (Roos et al. 2018), we can write the following equivalent constraint set:

$$\begin{cases} x_0 \geq \sum_{j=1}^m z_j \\ (z_j, x_0, (x_j - t)) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m. \end{cases} \quad (49)$$

Since constraints of type (48) appear in the upper bound approximation problem (24), we can use the equivalent representation (49) in each of the constraints to obtain problem (47). □

The lower bound can also be obtained by solving exponential cone programs. The worst-case scenario collection is obtained by solving (26), (27), (28). Here  $(r, u, V, r, \tau)$  are all parameters taken from the solution of the upper bound problem. This set of problems can be formulated as exponential conic problems, which is shown in the next corollary.

COROLLARY 5 (LOWER BOUND SCENARIOS). The problems (26), (27), and (28) can be written as the following exponential cone problems:

$$\begin{aligned}
 (26) : \arg \sup_{w,t} & \left\{ (1 + d^\top r) (\sum_{j=1}^m t_j) + (d^\top V + b^\top) w + d^\top u - \tau \right\} \\
 \text{s. t.} & \quad (1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m \\
 & \quad \sum_{j=1}^m w_j = 1, \\
 (27) : \arg \sup_{w,t} & \left\{ (-D_i^\top r) (\sum_{j=1}^m t_j) + (A_i^\top - D_i^\top V) w - D_i^\top u \right\} \\
 \text{s. t.} & \quad (1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m \\
 & \quad \sum_{j=1}^m w_j = 1, \\
 (28) : \arg \sup_{w,t} & \left\{ (-r_i) (\sum_{j=1}^m t_j) + (-V_{(i)}) w - u_i \right\} \\
 \text{s. t.} & \quad (1, w_j, t_j) \in \mathcal{K}_{\text{exp}}, \quad j = 1, \dots, m \\
 & \quad \sum_{j=1}^m w_j = 1.
 \end{aligned}$$

*Proof* Serrano (2015) shows that the negative entropy function is exponential conically representable, and the following problems are equivalent (here  $w \in \mathbb{R}^m$ ):

$$\begin{aligned}
 \sup_w \{c_0(-\sum_{i=1}^m w_i \log(w_i))\} &= \sup_{w,t} \{c_0 \sum_{i=1}^m t_i\} \\
 \text{s. t. } w_i \geq 0, \quad i = 1, \dots, m. & \quad \text{s. t. } (1, w_i, t_i) \in \mathcal{K}_{\text{exp}}, \quad i = 1, \dots, m.
 \end{aligned} \tag{50}$$

The result now follows by substitution of the conjugate (46) in (26), (27), and (28), respectively and then applying equivalence (50).  $\square$

### D.3. Sum-of-Max-Linear-Terms Optimization

Formally, the sum-of-max-terms function  $f : \mathbb{R}^m \mapsto \mathbb{R}$  is written as

$$f(z) = \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{z_j\}, \tag{51}$$

where the set  $\mathcal{I}_k \subseteq \{1, \dots, m\}$  for each  $k \in \{1, \dots, K\}$ . Moreover, we can assume  $\mathcal{I}_k \cap \mathcal{I}_\ell = \emptyset$  for any  $k \neq \ell$  and  $\cup_{k=1}^K \mathcal{I}_k = \{1, \dots, m\}$  without loss of generality, since otherwise we can add components to  $z$  to make this statement hold. The sum-of-max-linear-terms function we cover at this section is represented as

$$f(Ax + b) = \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{(j)} x + b_j\},$$

which is a convex and positively homogeneous function. The main convex maximization problem we are interested in is maximizing  $f(Ax + b)$  over  $U = \{x \in \mathbb{R}_+^n : Dx \leq d\}$ , formally:

$$\begin{aligned}
 \max_x & \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{(j)} x + b_j\} \\
 \text{s. t.} & \quad x \in U.
 \end{aligned} \tag{52}$$

This problem naturally arises when one applies the adversarial approach to robust optimization problems with uncertain sum-of-max-linear-terms constraints.

The conjugate of sum-of-max-linear-terms (51) is given by Roos et al. (2018) as:

$$f^*(w) = \begin{cases} 0 & \text{if } w_i \geq 0 \forall i = 1, \dots, m, \sum_{j \in \mathcal{I}_k} w_j = 1 \\ \infty & \text{otherwise.} \end{cases}$$

The formulation of the upper bound approximation for maximizing sum-of-max-linear-terms function over a polyhedron can be greatly simplified. This is due to the fact that sum-of-max-linear-terms function is a

positively homogeneous function as well as the trick of introducing auxiliary variables which give us a linear optimization problem in return.

Since the function is a positively homogeneous function, we can write the upper bound approximation problem (24) as:

$$\begin{aligned} & \inf_{u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times m}, \tau \in \mathbb{R}} \tau \\ \text{s. t.} \quad & d^\top u + \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{V_j^\top d + b_j\} \leq \tau \\ & -D_i^\top u + \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{i,(j)} - V_j^\top D_i\} \leq 0 \quad i = 1, \dots, n \\ & -u_i - \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{V_{j,(i)}\} \leq 0 \quad i = 1, \dots, q. \end{aligned} \quad (53)$$

Problem (53) can be reformulated as a linear optimization problem by using auxiliary variables. The worst-case scenarios for computing the lower bound are obtained from problems (26), (27), and (28), which simplify to:

$$(26) : \arg \sup_{w \in \text{dom } f^*} \{d^\top (u + Vw) + b^\top w - \tau\}, \quad (54)$$

$$(27) : \arg \sup_{w \in \text{dom } f^*} \{A_i^\top w - D_i^\top (u + Vw)\}, \quad i = 1, \dots, n \quad (55)$$

$$(28) : \arg \sup_{w \in \text{dom } f^*} \{-u_i - V_{(i)} w\}, \quad i = 1, \dots, q, \quad (56)$$

where we do not have the conjugate terms since  $f$  is a homogeneous function so its conjugate takes value 0. Therefore, the worst-case scenarios of each constraint are given by the following problems:

- For (54):

$$\begin{aligned} & \sup_w d^\top (u + Vw) + b^\top w - \tau \\ \text{s. t.} \quad & w_j \geq 0, \quad j = 1, \dots, m \\ & \sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K. \end{aligned} \quad (57)$$

Recalling the only variable here is  $w$ , this is a linear optimization problem. Moreover, since we have  $\mathcal{I}_k \cap \mathcal{I}_{k'} = \emptyset$  for  $k \neq k'$ , we can separate this problem to  $K$  independent optimization problems, where each problem  $k$  is:

$$\begin{aligned} c_k &= \sup_{w \geq 0} d^\top (u + V_{\{j\}} w) + b_{\{j\}}^\top w \\ \text{s. t.} \quad & \sum_{i=1}^{|\mathcal{I}_k|} w_i = 1. \end{aligned} \quad (58)$$

Here  $y \in \mathbb{R}^{|\mathcal{I}_k|}$  is the  $w$  components corresponding to the  $k$ -th term in the sum-of-max-linear-terms function definition. Similarly,  $V_{\{j\}}, b_{\{j\}} \in \mathbb{R}^{|\mathcal{I}_k|}$  are the components of  $V, b$  corresponding to the  $k$ -th term. Notice that problem (58) is a linear optimization problem over a simplex. The optimal value will have  $y_i = 1$  for some  $i$  and  $y_{i'} = 0$  for all  $i' \neq i$ . Therefore, the solution is

$$c_k = \max_{i=1, \dots, |\mathcal{I}_k|} \{d^\top (u + V_{\{j\},i})\} + b_{\{j\},i},$$

where  $V_{\{j\},i}, b_{\{j\},i}$  represent the  $i$ -th columns of  $V_{\{j\}}$  and  $b_{\{j\}}$ , respectively. Hence, the optimal value of (57) is given by  $-\tau + \sum_{k=1}^K c_k$ . The arg max value can be retrieved easily by detecting which  $y_i$  variables took value 1; there will be exactly  $K$  ones in the result and the rest will be zeros.

- For (55), for all  $i = 1, \dots, n$ :

$$\begin{aligned} & \sup_w A_i^\top w - D_i^\top (u + Vw) \\ \text{s. t.} \quad & w_j \geq 0, \quad j = 1, \dots, m \\ & \sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K. \end{aligned}$$

Similarly, this problem can be separated to  $K$  independent linear optimization problems over simplices. The optimal solution can be found analytically.

- For (56), for all  $i = 1, \dots, q$ :

$$\begin{aligned} & \sup_w -u_i - V_{(i)}w \\ \text{s. t. } & w_j \geq 0, \quad j = 1, \dots, m \\ & \sum_{j \in \mathcal{I}_k} w_j = 1, \quad k = 1, \dots, K. \end{aligned}$$

This problem can be solved analytically once again, concluding that all of the worst-case scenario finding procedure can be solved analytically.

## Appendix E: Upper Bound Approximation of Quadratic Maximization via SOCO

We follow Theorem 3 to apply the upper bound approximation for problem (41). Because  $f$  is a positively homogeneous function, the upper bound problem (24) reduces to the following problem for the variables  $u \in \mathbb{R}^q, V \in \mathbb{R}^{q \times (m+2)}, \tau \in \mathbb{R}$ :

$$\begin{aligned} & \inf \tau \\ \text{s. t. } & d^\top u + f(V^\top d + b) \leq 0 \\ & -D_i^\top u + f(A_i - V^\top D_i) \leq 0, \quad i = 1, \dots, n \\ & -u_i + f(-V_{(i)}^\top) \leq 0, \quad i = 1, \dots, q. \end{aligned} \tag{59}$$

Since  $V$  has  $m + 2$  columns, we represent it as:

$$V = [\tilde{V} \ \hat{v} \ \bar{v}] \quad \text{where } \tilde{V} \in \mathbb{R}^{q \times m}, \hat{v} \in \mathbb{R}^q, \bar{v} \in \mathbb{R}^q.$$

Constraints of problem (59) can be simplified to respectively:

$$\begin{cases} d^\top u + f(V^\top d + b) \leq 0 \\ = d^\top u + f\left(\begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + 1/2 - \tau/2 \\ \bar{v}^\top d - 1/2 - \tau/2 \end{pmatrix}\right) \leq 0 \\ = d^\top u + \bar{v}^\top d - \frac{1}{2} - \frac{\tau}{2} + \left\| \begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + \frac{1}{2} - \frac{\tau}{2} \end{pmatrix} \right\|_2 \leq 0 \end{cases}$$

$$\begin{cases} -D_i^\top u + f(A_i - V^\top D_i) \leq 0 \\ = -D_i^\top u + f\left(\begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - \hat{v}^\top D_i \\ \ell_i/2 - \bar{v}^\top D_i \end{pmatrix}\right) \leq 0 \\ = -D_i^\top u + \frac{\ell_i}{2} - \bar{v}^\top D_i + \left\| \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - \hat{v}^\top D_i \end{pmatrix} \right\|_2 \leq 0 \end{cases}$$

$$\begin{cases} -u_i + f(-V_{(i)}^\top) \leq 0 \\ = -u_i + f\left(\begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \\ -\bar{v}_i \end{pmatrix}\right) \leq 0 \\ = -u_i - \bar{v}_i + \left\| \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \end{pmatrix} \right\|_2 \leq 0. \end{cases}$$

Thus the upper bound approximation problem can be represented as the following second-order conic program:

$$\begin{aligned}
& \inf \tau \\
& \text{s. t. } d^\top u + \bar{v}^\top d - (1 + \tau)/2 + \left\| \begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + (1 - \tau)/2 \end{pmatrix} \right\|_2 \leq 0 \\
& \quad -D_i^\top u + \frac{\ell_i}{2} - \bar{v}^\top D_i + \left\| \begin{pmatrix} L_i - \tilde{V}^\top D_i \\ \ell_i/2 - \hat{v}^\top D_i \end{pmatrix} \right\|_2 \leq 0, \quad i = 1, \dots, n \\
& \quad -u_i - \bar{v}_i + \left\| \begin{pmatrix} -\tilde{V}_{(i)}^\top \\ -\hat{v}_i \end{pmatrix} \right\|_2 \leq 0, \quad i = 1, \dots, q.
\end{aligned} \tag{60}$$

## Appendix F: Lower Bound Scenarios for Quadratic Maximization

In the light of problems (26), (27), (28), the worst-case scenarios are collected by:

$$(26): \arg \sup_{(w, \tilde{w}) \in \text{dom } f^*} \left\{ -(1 + d^\top r) f^* \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} + (d^\top V + b^\top) \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} + d^\top u \right\}, \tag{61}$$

$$(27): \arg \sup_{(w, \tilde{w}) \in \text{dom } f^*} \left\{ (D_i^\top r) f^* \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} + (A_i^\top - D_i^\top V) \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} - D_i^\top u \right\}, \quad i = 1, \dots, n \tag{62}$$

$$(28): \arg \sup_{(w, \tilde{w}) \in \text{dom } f^*} \left\{ -u_i - V_{(i)} \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} + r_i f^* \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} \right\}, \quad i = 1, \dots, q. \tag{63}$$

We already showed the convex conjugate of  $f$  takes value 0 in its domain. Recalling

$$A = \begin{bmatrix} L \\ \ell^\top/2 \\ \ell^\top/2 \end{bmatrix}, \quad b = \begin{bmatrix} \mathbf{0} \\ (1 - \tau)/2 \\ (-1 - \tau)/2 \end{bmatrix}, \quad V = [\tilde{V} \ \hat{v} \ \bar{v}],$$

we can rewrite problem (61) as:

$$\begin{aligned}
& \sup_{w \in \mathbb{R}^{m+1}, \tilde{w} \in \mathbb{R}} \left( d^\top [\tilde{V} \ \hat{v} \ \bar{v}] + [\mathbf{0} \ (1 - \tau)/2 \ (-1 - \tau)/2] \right) \begin{pmatrix} w \\ \tilde{w} \end{pmatrix} + d^\top u \\
& \text{s. t. } \tilde{w} = 1 \\
& \quad \|w\|_2 \leq 1.
\end{aligned}$$

By using  $\tilde{w} = 1$ , we can eliminate  $\tilde{w}$  from the problem. Moreover,  $w$  only appears in a linear term, so we can change  $\|w\|_2 \leq 1$  constraint to be  $\|w\|_2 = 1$  instead, i.e.,  $w$  is a unit vector. So the problem becomes finding the value of:

$$\sup_{w: \|w\|_2=1} \left\{ \left[ \begin{pmatrix} \tilde{V}^\top d \\ \hat{v}^\top d + (1 - \tau)/2 \end{pmatrix}^\top w \right] + d^\top u + d^\top \bar{v} - (1 + \tau)/2. \right.$$

Hence we need to maximize a linear function over the unit ball, which can be solved analytically. This yields the objective value:

$$\left\| \begin{pmatrix} \tilde{V}^\top d \\ d^\top \hat{v} + (1 - \tau)/2 \end{pmatrix} \right\|_2 + d^\top u + d^\top \bar{v} - (1 + \tau)/2, \tag{64}$$

and the maximizer is:

$$\bar{W}_1 = \begin{bmatrix} h \left( \begin{pmatrix} \tilde{V}^\top d \\ d^\top \hat{v} + (1 - \tau)/2 \end{pmatrix} \right) \\ 1 \end{bmatrix},$$

where  $h(a) = a/\|a\|$  normalizes its input. Notice that the last element 1 comes since  $\tilde{w} = 1$  is in the domain of convex conjugate. The worst-case of constraints (62) and (63) are obtained via similar calculations.

## Appendix G: Data of Problems in Numerical Experiments

Here we explain the construction of the test data. The exact problem data, solutions, and the codes are available for download on GitHub at <https://github.com/selvi-aras/convex-max>.

### G.1. Experiments of Section 4.1

**Problem 1** The problem data is:

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} -1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \quad a = \begin{bmatrix} 0 \\ 2 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \quad \rho = 3.$$

For the next problems we generate larger problems by uniform random sampling (denoted simply as  $\sim$ ).

Remember that  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$ ,  $a \in \mathbb{R}^n$ . We let  $A_{ij}$  denote the elements of  $A$ .

**Problem 2:**  $m = 15$ ,  $n = 20$ ,  $A_{ij} \sim \{0, 1\}$ ,  $b_i \sim [-5, 5]$ ,  $a_j \sim [0, 3]$ ,  $\rho = 8$

**Problem 3:**  $m = 120$ ,  $n = 100$ ,  $A_{ij} \sim \{0, 1\}$ ,  $b_i \sim [-5, 5]$ ,  $a_j \sim [0, 4]$ ,  $\rho = 14$

**Problem 4:**  $m = 40$ ,  $n = 20$ ,  $A_{ij} \sim \{-4, -3, \dots, 3, 4\}$ ,  $b_i \sim [-5, 5]$ ,  $a_j \sim [0, 4]$ ,  $\rho = 10$

**Problem 5:**  $m = 100$ ,  $n = 50$ ,  $A_{ij} \sim [-5, 5]$ ,  $b_i \sim [-2, 2]$ ,  $a_j \sim [-4, 4]$ ,  $\rho = 12$

**Problem 6:**  $m = 100$ ,  $n = 100$ ,  $A_{ij} \sim [-4, 4]$ ,  $b_i \sim [-3, 3]$ ,  $a_j \sim [-4, 4]$ ,  $\rho = 15$

**Problem 7:**  $m = 30$ ,  $n = 200$ ,  $A_{ij} \sim [-4, 2]$ ,  $b_i \sim [-1, 1]$ ,  $a_j \sim [-3, 3]$ ,  $\rho = 16$

**Problem 8:**  $m = 80$ ,  $n = 400$ ,  $A_{ij} \sim [-2, 1]$ ,  $b_i \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $a_j \sim [-1, 1]$ ,  $\rho = 12$

**Problem 9:**  $m = 20$ ,  $n = 50$ ,  $A_{ij} \sim [0, 8]$ ,  $b_i \sim [-1, 1]$ ,  $a_j \sim [0, 4]$ ,  $\rho = 14$

**Problem 10:**  $m = 100$ ,  $n = 10,000$ ,  $A_{ij} \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $b_i \sim [-\frac{1}{4}, \frac{1}{4}]$ ,  $a_j \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $\rho = 15$

**Problem 11:**  $m = 1,000$ ,  $n = 1,000$ ,  $A_{ij} \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $b_i \sim [-\frac{1}{4}, \frac{1}{4}]$ ,  $a_j \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $\rho = 18$

**Problem 12:**  $m = 700$ ,  $n = 2,000$ ,  $A_{ij} \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $b_i \sim [-\frac{1}{4}, \frac{1}{4}]$ ,  $a_j \sim [-\frac{1}{2}, \frac{1}{2}]$ ,  $\rho = 24$ .

### G.2. Experiments of Section 4.2

**Problem 1:**  $A = \begin{bmatrix} -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}$ ,  $b = \mathbf{0}_{5 \times 1}$ ,  $a = \mathbf{0}_{5 \times 1}$ ,  $\rho = 3$

**Problem 2:**  $m = 5$ ,  $n = 20$ ,  $A_{ij} \sim [-1, 1]$ ,  $b_i \sim [-2, 2]$ ,  $a_j \sim [0, 1]$ ,  $\rho = 5$

**Problem 3:**  $m = 20$ ,  $n = 50$ ,  $A_{ij} \sim [-10, 10]$ ,  $b_i \sim [-3, 3]$ ,  $a_j \sim [-2, 2]$ ,  $\rho = 6$

**Problem 4:**  $m = 20$ ,  $n = 180$ ,  $A_{ij} \sim [-1, 0.5]$ ,  $b_i = 0$ ,  $a_j \sim [0, 1]$ ,  $\rho = 1$

**Problem 5:**  $m = 300$ ,  $n = 30$ ,  $A_{ij} \sim [-1, 1]$ ,  $b_i = 0$ ,  $a_j \sim [0, 1]$ ,  $\rho = 2$

### G.3. Experiments of Section 4.3

**Problem 1 (Enkhbat et al. 2006)** In this example, we solve:

$$\begin{aligned} \max_{x \in \mathbb{R}_+^{20}} \quad & \frac{1}{2} \sum_{i=1}^{20} (x_i - 2)^2 \\ \text{s. t.} \quad & Dx \leq d, \end{aligned}$$



where

$$D^\top = \begin{bmatrix} -3 & 7 & 0 & -5 & 1 & 1 & 0 & 2 & -1 & 1 \\ 7 & 0 & -5 & 1 & 1 & 0 & 2 & -1 & -1 & 1 \\ 0 & -5 & 1 & 1 & 0 & 2 & -1 & -1 & -9 & 1 \\ -5 & 1 & 1 & 0 & 2 & -1 & -1 & -9 & 3 & 1 \\ 1 & 1 & 0 & 2 & -1 & -1 & -9 & 3 & 5 & 1 \\ 1 & 0 & 2 & -1 & -1 & -9 & 3 & 5 & 0 & 1 \\ 0 & 2 & -1 & -1 & -9 & 3 & 5 & 0 & 0 & 1 \\ 2 & -1 & -1 & -9 & 3 & 5 & 0 & 0 & 1 & 1 \\ -1 & -1 & -9 & 3 & 5 & 0 & 0 & 1 & 7 & 1 \\ -1 & -9 & 3 & 5 & 0 & 0 & 1 & 7 & -7 & 1 \\ -9 & 3 & 5 & 0 & 0 & 1 & 7 & -7 & -4 & 1 \\ 3 & 5 & 0 & 0 & 1 & 7 & -7 & -4 & -6 & 1 \\ 5 & 0 & 0 & 1 & 7 & -7 & -4 & -6 & -3 & 1 \\ 0 & 0 & 1 & 7 & -7 & -4 & -6 & -3 & 7 & 1 \\ 0 & 1 & 7 & -7 & -4 & -6 & -3 & 7 & 0 & 1 \\ 1 & 7 & -7 & -4 & -6 & -3 & 7 & 0 & -5 & 1 \\ 7 & -7 & -4 & -6 & -3 & 7 & 0 & -5 & 1 & 1 \\ -7 & -4 & -6 & -3 & 7 & 0 & -5 & 1 & 1 & 1 \\ -4 & -6 & -3 & 7 & 0 & -5 & 1 & 1 & 0 & 1 \\ -6 & -3 & 7 & 0 & -5 & 1 & 1 & 0 & 2 & 1 \end{bmatrix}, d = \begin{bmatrix} -5 \\ 2 \\ -1 \\ -3 \\ 5 \\ 4 \\ -1 \\ 0 \\ 9 \\ 40 \end{bmatrix}. \quad (65)$$

**Problem 2** Same as problem 1, but the objective function is  $\frac{1}{2} \sum_{i=1}^{20} (x_i + 5)^2$ .

Problems 3-7 use the following problem:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} x^\top L^\top L x \\ & \text{s. t. } Dx \leq d \\ & \quad x \leq x_u, \end{aligned}$$

where  $L \in \mathbb{R}^{m \times n}$  is a matrix generated randomly where all the entries are sampled uniformly from 0 – 1. Moreover, let  $D$  be a similar matrix with all 0 – 1 random coefficients (except the last problem uses 0 – 2 random coefficients) and  $d$  to have integer entries uniformly distributed in range  $[d_l, d_u]$ . Denote  $q$  to be the number of total constraints.

**Problem 3**  $x_u = 5, d_l = 20, d_u = 60, n = 10, q = 15$

**Problem 4**  $x_u = 3, d_l = 30, d_u = 60, n = 50, q = 62$

**Problem 5**  $x_u = 2, d_l = 80, d_u = 120, n = 100, q = 130$

**Problem 6**  $x_u = 2, d_l = 160, d_u = 240, n = 200, q = 240$

**Problem 7**  $x_u = 1, d_l = 150, d_u = 300, n = 240, q = 280$ .

#### G.4. Experiments of Section 4.4

**Problem 1** considers the following problem:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} \log \left( \sum_{i=1}^m \exp(A_{(i)} x) \right) \\ & \text{s. t. } -\frac{i}{n} \leq x_i \leq \frac{n}{i}, \quad i = 1, \dots, n, \end{aligned}$$

where  $A_{ij} \sim [-3, 3]$ . In the numerical experiments  $n$  will vary.

Problems 2-4 consider the following problem:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} \log \left( \sum_{i=1}^m \exp(A_{(i)} x + b_i) \right) \\ & \text{s. t. } Dx \leq d. \end{aligned}$$

**Problem 2**  $n = q = m = 20, A_{ij} \sim [-3, 3], b_i \sim [-2, 2], D_{i,j} \sim [0, 1], d_i \sim [10, 30]$

**Problem 3**  $n = q = m = 50, A_{ij} \sim [-3, 3], b_i \sim [-1, 1], D_{i,j} \sim [0, 1], d_i \sim [20, 60]$

**Problem 4**  $n = q = m = 100, A_{ij} \sim [-3, 3], b_i \sim [-1, 1], D_{i,j} \sim [0, 1], d_i \sim [25, 75]$

Problems 5-6 consider:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} \log \left( \sum_{i=1}^m \exp(A_{(i)}x + b_i) \right) \\ \text{s. t. } & x_i \leq c, \quad i = 1, \dots, n \\ & x_i + x_j \leq u_{ij}, \quad i, j = 1, \dots, n, \quad i \neq j, \end{aligned}$$

**Problem 5**  $n = m = 10, A_{ij} \sim [-3, 3], b_i \sim [-1, 1], u_{ij} \sim [5, 15], c = 8$

**Problem 6**  $n = m = 30, A_{ij} \sim [-3, 3], b_i \sim [-1, 1], u_{ij} \sim [4, 12], c = 6.$

### G.5. Experiments of Section 4.5

For the easiness of bookkeeping, we generate problem with every max-term having the same number of elements, i.e.,  $|\mathcal{I}_k| = |\mathcal{I}_{k'}| \quad \forall k, k' \in \{1, \dots, K\}$ .

Problems 1-6 are defined by:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{(j)}x\} \\ \text{s. t. } & -\frac{i}{n} \leq x_i \leq \frac{n}{i}, \quad i = 1, \dots, n, \end{aligned}$$

where:

**Problem 1**  $n = 5, A_{ij} \sim [-5, 5], |\mathcal{I}_k| = 5, K = 1$

**Problem 2**  $n = 5, A_{ij} \sim [-5, 5], |\mathcal{I}_k| = 5, K = 10$

**Problem 3**  $n = 20, A_{ij} \sim [-5, 5], |\mathcal{I}_k| = 10, K = 10$

**Problem 4**  $n = 30, A_{ij} \sim [-5, 5], |\mathcal{I}_k| = 20, K = 20$

**Problem 5**  $n = 100, A_{ij} \sim [-5, 5], |\mathcal{I}_k| = 40, K = 30$

**Problem 6**  $n = 200, A_{ij} \sim [-4, 4], |\mathcal{I}_k| = 50, K = 50.$

Problems 7-10 are defined by:

$$\begin{aligned} & \max_{x \in \mathbb{R}_+^n} \sum_{k=1}^K \max_{j \in \mathcal{I}_k} \{A_{(j)}x + b_j\} \\ \text{s. t. } & Dx \leq d, \end{aligned}$$

**Problem 7**  $n = 10, A_{ij} \sim [-5, 5], b_j \sim [-10, 10], D_{ij} \sim [0, 1], d_i \sim [5, 15], |\mathcal{I}_k| = 5, K = 2$

**Problem 8**  $n = 10, A_{ij} \sim [-5, 5], b_j \sim [-10, 10], D_{ij} \sim [0, 1], d_i \sim [5, 15], |\mathcal{I}_k| = 50, K = 50$

**Problem 9**  $n = 30, A_{ij} \sim [-5, 5], b_j \sim [-10, 10], D_{ij} \sim [0, 1], d_i \sim [5, 15], |\mathcal{I}_k| = 50, K = 50$

**Problem 10**  $n = 50, A_{ij} \sim [-5, 5], b_j \sim [-10, 10], D_{ij} \sim [0, 1], d_i \sim [5, 15], |\mathcal{I}_k| = 60, K = 60 .$

Problems 10-13 consider the same problem as above, but  $D$  and  $d$  are as given in (65) with  $n = 20$ . The objective function varies as:

**Problem 11**  $A_{ij} \sim [-5, 10], b_j \sim [-10, 10], |\mathcal{I}_k| = 10, K = 10$

**Problem 12**  $A_{ij} \sim [-5, 10], b_j \sim [-10, 10], |\mathcal{I}_k| = 50, K = 10$

**Problem 13**  $A_{ij} \sim [-5, 10], b_j \sim [-10, 10], |\mathcal{I}_k| = 100, K = 50 .$