

# Uncertain Linear Programs: Extended Affinely Adjustable Robust Counterparts

Xin Chen, Yuhan Zhang

Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana–Champaign, Urbana, Illinois 61801  
{xinchen@illinois.edu, yzhang30@illinois.edu}

In this paper, we introduce the extended affinely adjustable robust counterpart to modeling and solving multistage uncertain linear programs with fixed recourse. Our approach first reparameterizes the primitive uncertainties and then applies the affinely adjustable robust counterpart proposed in the literature, in which recourse decisions are restricted to be linear in terms of the primitive uncertainties. We propose a special case of the extended affinely adjustable robust counterpart—the splitting-based extended affinely adjustable robust counterpart—and illustrate both theoretically and computationally that the potential of the affinely adjustable robust counterpart method is well beyond the one presented in the literature. Similar to the affinely adjustable robust counterpart, our approach ends up with deterministic optimization formulations that are tractable and scalable to multistage problems.

*Subject classifications:* programming; stochastic.

*Area of review:* Optimization.

*History:* Received March 2007; revision received January 2008; accepted March 2008. Published online in *Articles in Advance* March 16, 2009.

## 1. Introduction

Decision making under uncertainty is the key ingredient in many operations research problems, for example, supply chain management, revenue management, and financial planning. One of the most important approaches for optimization under uncertainty is stochastic programming, in which objectives and constraints of optimization models are defined by averaging over possible outcomes or considering probabilities of events of interest. Over the past 50 years, a variety of stochastic programming theory and algorithms have been developed and some successful stochastic programming applications have also been reported (see, e.g., Ruszczyński and Shapiro 2003, Birge and Louveaux 1997).

However, despite its immense modeling potential, stochastic programming faces two significant challenges. First, stochastic programs, especially multistage problems, are notoriously difficult to solve to optimality, and, quite often, even finding a feasible solution is already a hard problem. Second, stochastic optimization problems require full distributional knowledge in each of the uncertain data. Unfortunately, such information may rarely be available in practice. The lack of tractable methodology and the full distributional requirement have restricted the applicability of stochastic programming in many practical settings.

To cope with some of the challenges faced by stochastic programming, robust optimization received considerable attention in recent years as an alternative approach to deal with optimization problems under uncertainty. The first step in this direction was taken by Soyster (1973), who proposed

a worst-case model for linear optimization such that constraints are satisfied under all possible perturbations of the uncertain data of the underlying model. Recent developments in robust optimization focused on more elaborate uncertainty sets of uncertain data to alleviate overconservatism in worst-case models, as well as to maintain computational tractability of the proposed approaches (see, for example, Ben-Tal and Nemirovski 1998, 1999, 2000; El-Ghaoui and Lebret 1997; El-Ghaoui et al. 1998; Goldfarb and Iyengar 2003; Bertsimas and Sim 2003, 2004a, b, 2006; and Atamtürk 2006).

Most of the research on robust optimization focuses on static settings, in which all decisions must be made before the actual realization of the uncertain data (referred to as the *primitive uncertainties*). To extend the robust optimization methodology to dynamic settings, Ben-Tal et al. (2004) proposed the *adjustable robust counterpart* (ARC), in which the primitive uncertainties are assumed to vary within an uncertainty set while some decisions (recourse variables) can be made after the realization of the primitive uncertainties and can be adjusted to its actual realization. A closely related approach was proposed by Bertsimas and Caramanis (2005). In this approach, they introduced the concept of finite adaptability, which is based on the selection of a finite number of (constant) contingency plans to incorporate the information revealed over time. Bertsimas et al. (2006) applied it to model air traffic control. On the other hand, under the adjustable robust counterpart framework, Atamtürk and Zhang (2007) analyzed network design problems under uncertainty.

Because the general adjustable robust counterpart is intractable, Ben-Tal et al. (2004) proposed a tractable approach for solving fixed-recourse instances using affine decision rules—restricting recourse variables as affine functions of the realization of the primitive uncertainties, referred to as the *affinely adjustable robust counterpart* (AARC). Even though the AARC has been successfully applied to inventory management (Ben-Tal et al. 2004) and supply contract problems (Ben-Tal et al. 2005), it is not surprising that the performance of the AARC may not be satisfactory under situations in which the recourse variables may exhibit high nonlinearity in terms of the primitive uncertainties.

The goal of this paper is to illustrate that the potential of the AARC method is well beyond the one presented by Ben-Tal et al. (2004). Indeed, by reparameterizing the primitive uncertainties and then applying the AARC method, we end up with a new model, which allows us to relax to a certain degree the linearity restriction imposed by the AARC. Specifically, in our approach, we reparameterize the primitive uncertainties by introducing auxiliary variables and represent the recourse variables as affine functions of the auxiliary variables. By using these auxiliary variables, the model can now capture certain nonlinear responses of the recourse variables to the primitive uncertainties. In the sequel, we refer to the AARC as the AARC method directly applied on the primitive uncertainties and the *extended affinely adjustable robust counterpart* (EAARC) as the AARC method applied to the reparameterized model.

Because the primitive uncertainties can be reparameterized in a variety of different ways, the EAARC is rather flexible and encompasses a broad class of decision rules. We analyze a specific EAARC—the splitting-based EAARC—in depth. In a simple setting, the splitting-based EAARC essentially introduces auxiliary variables to represent the positive and negative parts of the primitive uncertainties. We demonstrate both theoretically and computationally that the splitting-based EAARC may significantly improve upon the AARC.

The idea of reparameterizing the original problem before applying the robust counterpart has been used in several papers for different purposes. For example, to avoid the overconservatism incurred by working directly on the primitive uncertainties, Ben-Tal et al. (1999) reparameterized the original multiperiod portfolio selection problem and then applied the robust counterpart approach. In Ben-Tal et al. (2006), the authors used a reparameterization scheme in a linear control problem to avoid a nonconvex robust counterpart. It is also common in the robust optimization literature, including Ben-Tal et al. (2004) and this paper, to reparameterize the uncertainty data by a vector of perturbations, referred to as the *primitive uncertainties*, varying in a nonempty convex compact perturbation set. Indeed, the AARC in Ben-Tal et al. (2004) first applies the affine decision rule on the uncertain data, which are

then reparameterized in terms of the primitive uncertainties, whereas in this paper we first reparameterize the primitive uncertainties and then apply the AARC, which, interestingly, results in a more flexible AARC.

Our splitting-based EAARC approach bears some similarity to the approach suggested by Chen et al. (2007, 2008). Specifically, both approaches are built upon the splitting of the primitive uncertainties to their negative parts and positive parts. In addition, the segregated linear decision rule proposed by Chen et al. (2008) also represents the recourse response as affine functions of these negative parts and positive parts. Moreover, both approaches end up with second-order conic programming problems.

However, the models analyzed by Chen et al. (2007, 2008) are fundamentally different from the one proposed here. Indeed, Chen et al. (2007, 2008) started with a (chance-constrained) stochastic program and proposed tractable (convex) approximations to the stochastic program, whereas here we start with an ARC and use the EAARC to approximate the ARC. Therefore, in this paper, the primitive uncertainties are restricted to an uncertainty set and thus are nonstochastic, whereas in Chen et al. (2007, 2008), the primitive uncertainties are stochastic with possibly known mean, support, and some deviation measures, which require totally different techniques for the analysis. Finally, even though both approaches end up with second-order conic programming problems, the formulations are different.

The rest of this paper is organized as follows. In §2, we illustrate the limitation of the AARC and introduce the EAARC. In §3, we present equivalent formulations for constraints derived from the EAARC. In §4, we introduce and analyze the splitting-based EAARC and identify conditions under which the splitting-based EAARC improves upon the AARC. We then conduct numerical experiments to demonstrate the advantage of the splitting-based EAARC over the AARC in §5. Finally, we provide some concluding remarks in §6.

## 2. Extended Affinely Adjustable Robust Counterpart

Consider the following two-stage uncertain linear programming problem<sup>1</sup>:

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \leq \mathbf{b}, \end{aligned}$$

where  $\mathbf{c} \in \mathcal{R}^n$ ,  $\mathbf{A} \in \mathcal{R}^{r \times n}$ ,  $\mathbf{B} \in \mathcal{R}^{r \times m}$ , and  $\mathbf{b} \in \mathcal{R}^r$  are uncertain data and  $\mathbf{c}'$  denotes the transpose of vector  $\mathbf{c}$ . In this problem, decision variables are classified into two groups. The first group, denoted by  $\mathbf{x}$ , represents “here and now” decisions, i.e., decisions made before the realization of uncertain data  $(\mathbf{c}, \mathbf{A}, \mathbf{B}, \mathbf{b})$ . The second group, denoted by  $\mathbf{y}$ , represents “wait and see” decisions, i.e., decisions that can be adjusted to the realization of uncertainty.

A framework for modeling the two-stage uncertain linear programs is two-stage stochastic programming. In such a framework, some stochastic structure is imposed on the uncertain data  $(\mathbf{c}, \mathbf{A}, \mathbf{B}, \mathbf{b})$  and the objective is to minimize the expected cost such that the constraints are satisfied with high probability. Unfortunately, multistage stochastic programs are generally hard to solve to optimality. To make things worse, specifying the stochastic structure of the uncertain data may not be realistic in practice.

An alternative approach for modeling two-stage uncertain linear programs is the adjustable robust counterpart first introduced by Ben-Tal et al. (2004). In an ARC, the constraints are satisfied for all the uncertain data varying in a given uncertainty set, and the second-stage decisions can be tuned to the realization of the uncertain data. Specifically, the two-stage ARC for the uncertain linear program can be written as follows:

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \forall (\mathbf{A}, \mathbf{B}, \mathbf{b}) \in \mathcal{U} \exists \mathbf{y}, \quad \mathbf{Ax} + \mathbf{By} \leq \mathbf{b}, \end{aligned} \quad (1)$$

where  $\mathcal{U}$  is the uncertainty set. Here, without loss of generality, we assume that the cost coefficient vector  $\mathbf{c}$  is fixed.

If  $\mathbf{B}$  is fixed in (1), then the above formulation defines the ARC to an uncertain linear program with *fixed recourse*. From now on, we focus on uncertain linear programs with fixed recourse. In particular, we assume that  $\mathbf{B}$  is fixed and the uncertainty set can be parameterized affinely in terms of the primitive uncertainties  $\mathbf{z} \in \mathfrak{R}^N$ :

$$\mathcal{U} = \left\{ (\mathbf{A}, \mathbf{b}): \exists \mathbf{z} \in \Gamma, (\mathbf{A}, \mathbf{b}) = (\mathbf{A}^0, \mathbf{b}^0) + \sum_{j=1}^N (\Delta \mathbf{A}^j, \Delta \mathbf{b}^j) z_j \right\},$$

where  $(\mathbf{A}^j, \mathbf{b}^j) \in \mathfrak{R}^{r \times n} \times \mathfrak{R}^r$ ,  $j = 0, 1, \dots, N$ , are given, and  $\Gamma$  is a nonempty closed convex subset in  $\mathfrak{R}^N$ . Notice that, because  $\mathbf{B}$  is fixed, we remove  $\mathbf{B}$  from the representation of the uncertainty set.

Define  $\mathbf{m}^0(\mathbf{x}) = \mathbf{A}^0 \mathbf{x} - \mathbf{b}^0$ . For a given  $\mathbf{x}$ , let  $\mathbf{M}(\mathbf{x})$  be a matrix in  $\mathfrak{R}^{r \times N}$  with the  $j$ th column given by  $\Delta \mathbf{A}^j \mathbf{x} - \Delta \mathbf{b}^j$ . The feasible set of the first-stage decision in problem (1) can be written as

$$X_0 = \{ \mathbf{x}: \forall \mathbf{z} \in \Gamma, \exists \mathbf{y}, \mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z} + \mathbf{By} \leq \mathbf{0} \}.$$

In general, the ARC problem (1) is intractable (see Ben-Tal et al. 2004). To overcome this difficulty, Ben-Tal et al. (2004) proposed the AARC assuming that the “wait and see” (or recourse) variables are affinely dependent on the primitive uncertainties. That is,

$$\mathbf{y} = \mathbf{y}^0 + \sum_{j=1}^N \mathbf{y}^j z_j,$$

which will render the problem tractable. In this case, the feasible set  $X_0$  is approximated by

$$X_{\text{AARC}} = \left\{ \mathbf{x}: \exists \mathbf{y}^0, \mathbf{y}^j, \mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z} + \mathbf{By}^0 + \sum_{j=1}^N \mathbf{By}^j z_j \leq \mathbf{0} \forall \mathbf{z} \in \Gamma \right\}.$$

It is clear that  $X_{\text{AARC}} \subseteq X_0$ .

The AARC is motivated by the belief that the change in recourse variables is often linear to small changes in data uncertainty. However, the AARC may be too restrictive, particularly in cases where linear dependency fails to be a good approximation, as illustrated in the following example.

EXAMPLE 1. Consider the following ARC:

$$\begin{aligned} \min \quad & x \\ \forall \|\mathbf{z}\|_1 \leq 1 \exists \mathbf{y} \quad & \text{s.t.} \quad -y_i \leq z_i, \quad -y_i \leq -z_i, \quad i = 1, \dots, N, \\ & \sum_{i=1}^N y_i \leq x. \end{aligned}$$

The example implies that  $|z_i| \leq y_i$  and hence  $x \geq \sum_{i=1}^N y_i \geq \|\mathbf{z}\|_1$ . Therefore, the optimal objective value of the ARC is one.

If we employ the linear decision rule  $\mathbf{y} = \mathbf{y}^0 + \sum_{j=1}^N \mathbf{y}^j z_j$ , then the AARC is as follows:

$$\begin{aligned} \min \quad & x \\ \text{s.t.} \quad & -\left( y_i^0 + \sum_{j=1}^N y_i^j z_j \right) \leq z_i, \\ & -\left( y_i^0 + \sum_{j=1}^N y_i^j z_j \right) \leq -z_i, \quad i = 1, \dots, N \quad \forall \|\mathbf{z}\|_1 \leq 1, \\ & \sum_{i=1}^N \left( y_i^0 + \sum_{j=1}^N y_i^j z_j \right) \leq x. \end{aligned}$$

The first two constraints imply that  $|z_i| \leq y_i^0 + \sum_{j=1}^N y_i^j z_j$  for all  $\|\mathbf{z}\|_1 \leq 1$ . In particular, it is true for  $\mathbf{z} = \pm \mathbf{e}^i$ , where  $\mathbf{e}^i$  is the unit vector with one at its  $i$ th component. Therefore,

$$1 \leq y_i^0 + y_i^i, \quad 1 \leq y_i^0 - y_i^i,$$

which implies that  $y_i^0 \geq 1$ . In addition, if we let  $\mathbf{z} = \mathbf{0}$ , the last constraint then implies that  $x \geq N$ . Finally, if  $\mathbf{y}^j = \mathbf{0}$  for  $j = 1, \dots, N$  and  $\mathbf{y}^0 = \mathbf{e}$  where  $\mathbf{e}$  is the all-ones vector, then we know that the optimal objective value of the AARC is  $N$ .

The purpose of this paper is to relax the restriction of the AARC proposed in Ben-Tal et al. (2004), in which the recourse variables depend on the *primitive uncertainties* in an affine manner. Specifically, we introduce auxiliary variables  $\mathbf{u} \in \mathfrak{R}^K$  for some dimension  $K$ , such that the recourse variables can be represented as affine functions of the *auxiliary variables*  $\mathbf{u}$  in addition to the primitive uncertainties  $\mathbf{z}$ :

$$\mathbf{y} = \mathbf{y}^0 + \sum_{j=1}^N \mathbf{y}^j z_j + \sum_{j=N+1}^{N+K} \mathbf{y}^j u_j, \quad (2)$$

where  $(\mathbf{z}, \mathbf{u}) \in \Lambda$ ,  $u_j$  is the  $j$ th component of  $\mathbf{u}$ , and  $\Lambda$ , referred to as the *extended uncertainty set*, is a nonempty closed convex set in  $\mathfrak{R}^{N+K}$  to be specified later.

We now provide some motivation for using the above formulation (2). As we illustrated in Example 1, it may be too restrictive in certain settings to require the recourse variables to be affine functions of the realization of the primitive uncertainties. By introducing the new variables  $\mathbf{u}$ , we hope that they would capture certain nonlinearity of the response functions. Ideally, it would be nice to represent  $\mathbf{u}$  as some nonlinear functions of  $\mathbf{z}$ , say piecewise-linear functions of  $\mathbf{z}$ , which, however, usually leads to intractable formulations. Thus, instead of representing  $\mathbf{u}$  directly as some nonlinear functions of  $\mathbf{z}$ , we impose the constraint  $(\mathbf{z}, \mathbf{u}) \in \Lambda$ .

Letting  $\mathbf{y}$  take the form in (2), we can write down an approximation to problem (1) as follows:

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \exists \mathbf{y}^0, \mathbf{y}^j, \quad \mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z} + \mathbf{B}\mathbf{y}^0 + \sum_{j=1}^K \mathbf{B}\mathbf{y}^j z_j \\ & + \sum_{j=N+1}^{N+K} \mathbf{B}\mathbf{y}^j u_j \leq \mathbf{0} \quad \forall (\mathbf{z}, \mathbf{u}) \in \Lambda. \end{aligned} \quad (3)$$

We call the problem the *extended affinely adjustable robust counterpart*. In this problem, the feasible set of the first-stage decision is

$$\begin{aligned} X_{\text{EAARC}} := \left\{ \mathbf{x}: \exists \mathbf{y}^0, \mathbf{y}^j, \mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z} + \mathbf{B}\mathbf{y}^0 + \sum_{j=1}^K \mathbf{B}\mathbf{y}^j z_j \right. \\ \left. + \sum_{j=N+1}^{N+K} \mathbf{B}\mathbf{y}^j u_j \leq \mathbf{0} \quad \forall (\mathbf{z}, \mathbf{u}) \in \Lambda \right\}, \end{aligned} \quad (4)$$

which can be considered as an approximation of the feasible set  $X_0$ . When necessary, we will also use  $X_{\text{EAARC}}(\Lambda)$  to emphasize the extended uncertainty set  $\Lambda$ .

It is straightforward to show that if  $\Gamma \subseteq \text{Proj}_{\mathbf{z}}(\Lambda)$ , where  $\text{Proj}_{\mathbf{z}}(\Lambda)$  is the projection of  $\Lambda$  into the  $\mathbf{z}$  space, then  $X_{\text{EAARC}} \subseteq X_0$ .

Because there are many different ways of choosing the extended uncertainty set  $\Lambda$ , the EAARC is rather flexible. For example, the AARC is a special case of the EAARC. On the other hand, if the uncertainty set  $\Gamma$  itself is defined through some auxiliary variables, then there is a natural way of defining the extended uncertainty set  $\Lambda$ . Specifically, consider the uncertainty set analyzed in Ben-Tal et al. (2004):

$$\Gamma = \{ \mathbf{z}: \exists \mathbf{U}: \mathbf{Z}\mathbf{z} + \mathbf{U}\mathbf{u} \preceq_{\mathcal{H}} \mathbf{d} \},$$

where  $\mathcal{H}$  is a nonempty convex cone and  $\mathbf{x} \preceq \mathbf{y}$  if and only if  $\mathbf{y} - \mathbf{x} \in \mathcal{H}$ . In this case, it is natural to define

$$\Lambda = \{ (\mathbf{z}, \mathbf{u}): \mathbf{Z}\mathbf{z} + \mathbf{U}\mathbf{u} \preceq_{\mathcal{H}} \mathbf{d} \}.$$

Finally, if we choose  $\Lambda$  appropriately,  $X_{\text{EAARC}}$  may recover the feasible set  $X_0$ . Indeed, assume that the set  $\Gamma$  is a polytope with extreme points,  $\mathbf{z}^1, \dots, \mathbf{z}^M$ . That is,

$$\Gamma = \left\{ \mathbf{z}: \mathbf{z} = \sum_{j=1}^M \mathbf{z}^j u_j, \sum_{j=1}^M u_j = 1, u_j \geq 0 \quad \forall j = 1, \dots, M \right\}. \quad (5)$$

Choose  $K = M$  and let

$$\Lambda = \left\{ (\mathbf{z}, \mathbf{u}): \mathbf{z} = \sum_{j=1}^M \mathbf{z}^j u_j, \sum_{j=1}^M u_j = 1, u_j \geq 0 \quad \forall j = 1, \dots, M \right\}. \quad (6)$$

In this case, we have

$$\begin{aligned} X_{\text{EAARC}} = \left\{ \mathbf{x}: \exists \mathbf{y}^0, \mathbf{y}^j, \mathbf{m}^0(\mathbf{x}) + \mathbf{B}\mathbf{y}^0 \right. \\ \left. + \sum_{j=1}^M (\mathbf{M}(\mathbf{x})\mathbf{z}^j + \mathbf{B}\mathbf{y}^j) u_j \leq \mathbf{0} \quad \forall \mathbf{e}'\mathbf{u} = 1, \mathbf{u} \geq \mathbf{0} \right\}. \end{aligned}$$

In this following, we show that  $X_{\text{EAARC}} = X_0$ .

**THEOREM 1.** *If  $\Gamma$  and  $\Lambda$  are given by (5) and (6), respectively, then  $X_{\text{EAARC}} = X_0$ .*

**PROOF.** From the definition of  $\Gamma$  and  $\Lambda$ , we have  $\Gamma = \text{Proj}_{\mathbf{z}}(\Lambda)$ . Thus,  $X_{\text{EAARC}} \subseteq X_0$ . It remains to show that  $X_0 \subseteq X_{\text{EAARC}}$ .

Recall the definition of  $X_0 = \{ \mathbf{x}: \forall \mathbf{z} \in \Gamma, \exists \mathbf{y}, \mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z} + \mathbf{B}\mathbf{y} \leq \mathbf{0} \}$ . Hence, for a given  $\mathbf{x} \in X_0$ , there exists  $\mathbf{y}^j$  such that

$$\mathbf{m}^0(\mathbf{x}) + \mathbf{M}(\mathbf{x})\mathbf{z}^j + \mathbf{B}\mathbf{y}^j \leq \mathbf{0} \quad \forall j = 1, 2, \dots, M,$$

which implies that

$$\mathbf{m}^0(\mathbf{x}) + \sum_{j=1}^M (\mathbf{M}(\mathbf{x})\mathbf{z}^j + \mathbf{B}\mathbf{y}^j) u_j \leq \mathbf{0} \quad \forall \mathbf{e}'\mathbf{u} = 1, \mathbf{u} \geq \mathbf{0}.$$

Thus,  $x \in X_{\text{EAARC}}$  and  $X_0 \subseteq X_{\text{EAARC}}$ .  $\square$

### 3. Constraint Reformulation

In the previous section, we showed that the EAARC is rather flexible. In fact, if we define the extended uncertainty set  $\Lambda$  using the extreme points of the original uncertainty set  $\Gamma$ , we can recover the feasible set of the ARC. Unfortunately, in general, for a polyhedral set defined by linear equalities and inequalities, the number of extreme points is exponential in terms of the number of constraints of the polyhedral set, and we may end up with an intractable formulation.

Because the EAARC is essentially the application of the AARC on a reparameterized model, all of the theoretical results for the AARC method carry over to the EAARC verbatim as long as the extended uncertainty set is chosen appropriately. In the following, we present equivalent formulations for the robust constraints when the extended uncertainty set  $\Lambda$  is defined as follows:

$$\Lambda = \{ (\mathbf{z}, \mathbf{u}): \mathbf{Z}\mathbf{z} + \mathbf{U}\mathbf{u} \leq \mathbf{d} \}.$$

Under this assumption and using the constraint reformulation result of the AARC method (see Ben-Tal et al. 2004 for more details), we have that  $x \in X_{\text{EAARC}}$  if and only if

there exist  $\mathbf{W}, \mathbf{y}^0, \mathbf{Y}^Z, \mathbf{Y}^U$  such that the following linear inequalities hold:

$$\begin{aligned} \mathbf{m}^0(\mathbf{x}) + \mathbf{B}\mathbf{y}^0 + \mathbf{W}\mathbf{d} &\leq \mathbf{0}, \\ \mathbf{W}\mathbf{Z} &= \mathbf{M}(\mathbf{x}) + \mathbf{B}\mathbf{Y}^Z, \\ \mathbf{W}\mathbf{U} &= \mathbf{B}\mathbf{Y}^U, \\ \mathbf{W} &\geq \mathbf{0}. \end{aligned} \tag{7}$$

Thus, problem (3) can be equivalently reformulated as the following linear program:

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & (7) \text{ holds.} \end{aligned}$$

If the set  $\Lambda$  has a polynomial size representation in terms of the input data, then the above linear program and hence its associated EAARC are tractable.

In the remainder of this section, we derive the dual of the feasibility problem (7). This dual is useful when we compare the feasibility set based on the EAARC and the one based on the AARC for the extended uncertainty sets proposed in the next section.

LEMMA 2. *The dual of problem (7) is given as follows:*

$$\begin{aligned} \min \quad & -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle \\ \text{s.t.} \quad & \mathbf{Z}\boldsymbol{\beta}' + \mathbf{U}\boldsymbol{\gamma}' \leq \mathbf{d}\boldsymbol{\alpha}', \\ & \mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \\ & \mathbf{B}'\boldsymbol{\beta} = \mathbf{0}, \\ & \mathbf{B}'\boldsymbol{\gamma} = \mathbf{0}, \\ & \boldsymbol{\alpha} \geq \mathbf{0}, \end{aligned} \tag{8}$$

where  $\boldsymbol{\alpha} \in \mathbb{R}^r$ ,  $\boldsymbol{\beta} \in \mathbb{R}^{r \times N}$  and  $\langle \cdot, \cdot \rangle$  denotes the inner product (specifically,  $\langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle = \text{trace}(\mathbf{M}(\mathbf{x})'\boldsymbol{\beta})$  denotes the inner product of the two matrices). In addition,  $\mathbf{x} \in X_{\text{EAARC}}$  if and only if the optimal value of (8) is zero.

PROOF. Define the Lagrangian function of the feasibility problem (7):

$$\begin{aligned} L(\mathbf{y}^0, \mathbf{Y}^Z, \mathbf{W}, \mathbf{Y}^U, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) &= \langle -\mathbf{m}^0(\mathbf{x}) - \mathbf{B}\mathbf{y}^0 - \mathbf{W}\mathbf{d}, \boldsymbol{\alpha} \rangle + \langle \mathbf{W}\mathbf{Z} - \mathbf{M}(\mathbf{x}) - \mathbf{B}\mathbf{Y}^Z, \boldsymbol{\beta} \rangle \\ &\quad + \langle \mathbf{W}\mathbf{U} - \mathbf{B}\mathbf{Y}^U, \boldsymbol{\gamma} \rangle \\ &= -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle + \langle \mathbf{W}, \mathbf{Z}\boldsymbol{\beta}' + \mathbf{U}\boldsymbol{\gamma}' - \mathbf{d}\boldsymbol{\alpha}' \rangle \\ &\quad - \langle \mathbf{y}^0, \mathbf{B}'\boldsymbol{\alpha} \rangle - \langle \mathbf{Y}^Z, \mathbf{B}'\boldsymbol{\beta} \rangle - \langle \mathbf{Y}^U, \mathbf{B}'\boldsymbol{\gamma} \rangle. \end{aligned}$$

Consider the dual function defined by

$$Q(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \max_{\mathbf{W} \geq \mathbf{0}, \mathbf{y}^0, \mathbf{Y}^Z, \mathbf{Y}^U} L(\mathbf{y}^0, \mathbf{Y}^Z, \mathbf{W}, \mathbf{Y}^U, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}).$$

The Lagrangian dual of the feasibility problem (7) is given as

$$\min_{\boldsymbol{\alpha} \geq \mathbf{0}, \boldsymbol{\beta}, \boldsymbol{\gamma}} Q(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}),$$

which is equivalent to (8). It is clear that the feasibility problem (7) is feasible if and only if its dual has an optimal objective value zero.  $\square$

The sets  $\Gamma$  and  $\Lambda$  can be extended to incorporate conic constraints. We have a result parallel to Lemma 2. Because its proof is similar to the one for Lemma 2 and follows directly from the conic programming duality theory, we omit its proof.

LEMMA 3. *Assume that*

$$\Lambda = \{(\mathbf{z}, \mathbf{u}): \mathbf{Z}\mathbf{z} + \mathbf{U}\mathbf{u} \preceq_{\mathcal{K}} \mathbf{d}\},$$

and there exists  $(\mathbf{z}, \mathbf{u})$  such that  $\mathbf{d} - \mathbf{Z}\mathbf{z} - \mathbf{U}\mathbf{u}$  lies in the interior of  $\mathcal{K}$ . Then,  $\mathbf{x} \in X_{\text{EAARC}}$  if and only if zero is the optimal value of the following problem:

$$\begin{aligned} \min \quad & -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle \\ \text{s.t.} \quad & \mathbf{Z}\boldsymbol{\beta}' + \mathbf{U}\boldsymbol{\gamma}' \preceq_{\mathcal{K}} \mathbf{d}\boldsymbol{\alpha}', \\ & \mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \\ & \mathbf{B}'\boldsymbol{\beta} = \mathbf{0}, \\ & \mathbf{B}'\boldsymbol{\gamma} = \mathbf{0}, \\ & \boldsymbol{\alpha} \geq \mathbf{0}. \end{aligned}$$

## 4. The Splitting-Based EAARC

In this section, we propose one way of choosing the extended uncertainty set  $\Lambda$ . To illustrate the basic idea, we consider a simple setting in which the uncertainty set  $\Gamma$  is the intersection of a polyhedral set and a norm constrained set, that is,

$$\Gamma = \{\mathbf{z}: \mathbf{L}\mathbf{z} \leq \mathbf{1}\} \cap \{\mathbf{z}: \|\mathbf{z}\| \leq \Omega\} \tag{9}$$

for some norm  $\|\cdot\|$ . The idea is essentially to split  $\mathbf{z}$  into two parts, which can be thought of as the positive part and the negative part of  $\mathbf{z}$ . Specifically, we let  $\mathbf{u} = (\mathbf{v}', \mathbf{w}')'$  and  $\mathbf{z} = \mathbf{v} - \mathbf{w}$ . That is,  $\mathbf{z}$  is defined as the difference of two auxiliary variables  $\mathbf{v}$  and  $\mathbf{w}$ , which represents the positive part and the negative part of  $\mathbf{z}$ , respectively. The extended uncertainty set can be naturally defined as

$$\Lambda = \{(\mathbf{z}, \mathbf{u}): \mathbf{L}\mathbf{z} \leq \mathbf{1}\} \cap \{(\mathbf{z}, \mathbf{u}): \mathbf{u} = (\mathbf{v}', \mathbf{w}')', \mathbf{z} = \mathbf{v} - \mathbf{w}, \|\mathbf{v} + \mathbf{w}\| \leq \Omega, \mathbf{v} \geq \mathbf{0}, \mathbf{w} \geq \mathbf{0}\}.$$

Thus, instead of using affine decision rules in terms of  $\mathbf{z}$ , we consider decision rules that are affine in  $\mathbf{v}$  and  $\mathbf{w}$  (obviously,

the affine part in  $\mathbf{z}$  is automatically subsumed in this case), namely,

$$\mathbf{y} = \mathbf{y}^0 + \sum_{j=1}^N (\mathbf{r}^j v_j + \mathbf{s}^j w_j),$$

where  $\mathbf{y}^0$ ,  $\mathbf{r}^j$ , and  $\mathbf{s}^j$  are vectors to be determined. The resulting EAARC is referred to as the *splitting-based EAARC*.

We now apply the splitting-based EAARC to Example 1 to illustrate that the EAARC may significantly improve upon the AARC.

**EXAMPLE 2.** Consider the adjustable robust counterpart presented in Example 1. In the EAARC decision rule, we have  $\mathbf{z} = \mathbf{v} - \mathbf{w}$ ,  $\mathbf{v} \geq 0$ ,  $\mathbf{w} \geq 0$  and  $\mathbf{y} = \mathbf{y}^0 + \sum_{j=1}^N (\mathbf{r}^j v_j + \mathbf{s}^j w_j)$ . Then, the EAARC is defined as follows:

$$\begin{aligned} \min \quad & x \\ \text{s.t.} \quad & -\left(y_i^0 + \sum_{j=1}^N (r_i^j v_j + s_i^j w_j)\right) \leq v_i - w_i, \\ & -\left(y_i^0 + \sum_{j=1}^N (r_i^j v_j + s_i^j w_j)\right) \leq -(v_i - w_i), \\ & \quad i = 1, \dots, N, \forall \|\mathbf{v} + \mathbf{w}\|_1 \leq 1, \mathbf{v} \geq 0, \mathbf{w} \geq 0, \\ & \sum_{i=1}^N \left(y_i^0 + \sum_{j=1}^N (r_i^j v_j + s_i^j w_j)\right) \leq x. \end{aligned}$$

The first two constraints imply that

$$|v_i - w_i| \leq y_i^0 + \sum_{j=1}^N (r_i^j v_j + s_i^j w_j).$$

It is clear that  $\mathbf{y} = \sum_{j=1}^N \mathbf{e}^j (v_j + w_j)$  satisfies the first two constraints. Furthermore, in this case,

$$x \geq \sum_{i=1}^N (v_i + w_i) = \|\mathbf{v} + \mathbf{w}\|_1.$$

Hence, the optimal objective value of the EAARC is one. This is exactly the optimal objective value of the ARC, whereas the optimal objective value of the AARC is  $N$ .

Observe that in the above example the optimal response function is  $y_j(\mathbf{z}) = |z_j|$ , which is nonlinear in  $\mathbf{z}$ . By introducing the positive and negative parts of  $\mathbf{z}$ , we are able to capture the nonlinearity in this specific example.

We now extend the splitting idea to more general uncertainty sets. Specifically, we focus on the uncertainty set

$$\Gamma = \left\{ \mathbf{z}: \exists ((\mathbf{u}^1)', \dots, (\mathbf{u}^\tau)') \in \mathfrak{R}^{K_1} \times \dots \times \mathfrak{R}^{K_\tau} \quad \mathbf{Z}\mathbf{z} + \sum_{t=1}^{\tau} \mathbf{U}^t \mathbf{u}^t \leq \mathbf{d}, \right. \\ \left. \|\mathbf{z}\|_{(0)} \leq \Omega_0, \|\mathbf{u}^t\|_{(t)} \leq \Omega_t, t = 1, \dots, \tau \right\},$$

where  $\mathbf{d} \in \mathfrak{R}^\ell$ ,  $\mathbf{Z} \in \mathfrak{R}^{\ell \times N}$ , and  $\mathbf{U}_t \in \mathfrak{R}^{\ell \times K_t}$ . Here  $\|\cdot\|_{(t)}$ ,  $t = 0, 1, \dots, \tau$  are vector norms. In this paper, all of the vector norms  $\|\cdot\|_{(t)}$  in the uncertainty set satisfy the following condition:

$$\|\mathbf{u}^t\|_{(0)} = \|\mathbf{u}^t\|_{(t)}, \tag{10}$$

where  $\mathbf{u}^t$  is the vector with the  $j$ th component equal to  $|u_j| \forall j \in \{1, \dots, N\}$ . For technical reasons, we assume that the Slater condition holds. That is, there exists  $\mathbf{u}^t$ ,  $t = 0, 1, \dots, \tau$ , such that  $\mathbf{Z}\mathbf{u}^0 + \sum_{t=1}^{\tau} \mathbf{U}^t \mathbf{u}^t \leq \mathbf{d}$  with  $\|\mathbf{u}^t\|_{(t)} < \Omega_t$  if  $\|\cdot\|_{(t)}$  is not a polyhedral norm. This assumption would allow us to employ Lemma 3 in the following analysis.

The representation of our uncertainty set is broad enough to include many uncertainty sets commonly used in the robust optimization literature. Obviously, the uncertainty set (9) is a special case. More importantly, it also includes the intersection of several general ellipsoids as a special case.

We now propose a specific extended uncertainty set  $\Lambda$  by splitting  $(\mathbf{z}, \mathbf{u})$  into its positive and negative parts. Specifically, let  $\mathbf{v} = ((\mathbf{v}^0)', \dots, (\mathbf{v}^\tau)') \in \mathfrak{R}^{K_0} \times \dots \times \mathfrak{R}^{K_\tau}$ ,  $\mathbf{w} = ((\mathbf{w}^0)', \dots, (\mathbf{w}^\tau)') \in \mathfrak{R}^{K_0} \times \dots \times \mathfrak{R}^{K_\tau}$  with  $K_0 = N$ , and define the extended uncertainty set as follows:

$$\Lambda = \left\{ (\mathbf{z}, \mathbf{v}, \mathbf{w}): \mathbf{Z}\mathbf{z} + \sum_{t=1}^{\tau} \mathbf{U}^t (\mathbf{v}^t - \mathbf{w}^t) \leq \mathbf{d}, \mathbf{z} = \mathbf{v}^0 - \mathbf{w}^0, \mathbf{v}^t \geq \mathbf{0}, \right. \\ \left. \mathbf{w}^t \geq \mathbf{0}, \|\mathbf{v}^t + \mathbf{w}^t\|_{(t)} \leq \Omega_t, t = 0, 1, \dots, \tau \right\}. \tag{11}$$

Now instead of using affine decision rules in terms of the primitive uncertainties  $\mathbf{z}$ , we represent the recourse decision  $\mathbf{y}$  affinely in  $\mathbf{v}$  and  $\mathbf{w}$  (again the affine part in  $\mathbf{z}$  is automatically subsumed in this case), i.e.,

$$\mathbf{y} = \mathbf{y}^0 + \sum_{t=0}^{\tau} \sum_{i=1}^{K_t} (\mathbf{r}^{t,i} v_i^t + \mathbf{s}^{t,i} w_i^t). \tag{12}$$

We now formulate the EAARC as an equivalent conic programming problem, whose proof follows from Theorem 3.2 in Ben-Tal et al. (2004) and thus is omitted.

**THEOREM 4.** *The splitting-based EAARC with the extended uncertainty set (11) and the affine decision rule (12) is equivalent to the following conic programming problem:*

$$\begin{aligned} \min \quad & \mathbf{c}'\mathbf{x} \\ \text{s.t.} \quad & \mathbf{m}^0(\mathbf{x}) + B\mathbf{y}^0 - \Phi\mathbf{d} \leq \mathbf{0}, \\ & (\boldsymbol{\mu}^t)' \geq \Omega_t \|\mathbf{h}^t\|_{(t)}^*, \quad t = 0, 1, \dots, \tau, \\ & \mathbf{H}^0 \geq \mathbf{B}\mathbf{r}^0 + \Phi\mathbf{Z} + \mathbf{M}(\mathbf{x}), \\ & \mathbf{H}^0 \geq \mathbf{B}\mathbf{s}^0 - \Phi\mathbf{Z} - \mathbf{M}(\mathbf{x}), \\ & \mathbf{H}^t \geq \mathbf{B}\mathbf{r}^t + \Phi\mathbf{U}^t, \quad t = 1, 2, \dots, \tau, \\ & \mathbf{H}^t \geq \mathbf{B}\mathbf{s}^t - \Phi\mathbf{U}^t, \quad t = 1, 2, \dots, \tau, \\ & \mathbf{H}^t \geq \mathbf{0}, \quad t = 0, 1, \dots, \tau, \\ & \Phi \geq \mathbf{0}, \end{aligned} \tag{13}$$

where  $\Phi \in \mathbb{R}^{r \times \ell}$ ,  $\mathbf{H}^t \in \mathbb{R}^{r \times K_t}$ , and  $\boldsymbol{\mu}^t \in \mathbb{R}^r$ . In addition,  $\|(\mathbf{H}^t)'\|_{(t)}^*$  is an  $r$ -dimensional row vector with its  $j$ th entry equal to the conjugate norm of  $\|\cdot\|_{(t)}$  taken over the  $j$ th row of  $\mathbf{H}^t$ .

Note that when all the vector norms  $\|\cdot\|_{(t)}$  are 2-norms, the above problem (13) becomes a second-order conic program.

We are interested in identifying conditions under which  $X_{\text{EAARC}}(\Lambda)$  improves upon  $X_{\text{AARC}}$  when  $\Lambda$  is given in (11). However, rather than directly comparing  $X_{\text{EAARC}}(\Lambda)$  and  $X_{\text{AARC}}$ , we will compare  $X_{\text{EAARC}}(\Lambda)$  and another extended uncertainty set  $X_{\text{EAARC}}(\Lambda^0)$ , where

$$\Lambda^0 = \left\{ (\mathbf{z}, \mathbf{u}): \mathbf{Z}\mathbf{z} + \sum_{t=1}^{\tau} \mathbf{U}^t \mathbf{u}^t \leq \mathbf{d}, \|\mathbf{z}\|_{(0)} \leq \Omega_0, \right. \\ \left. \|\mathbf{u}^t\|_{(t)} \leq \Omega_t, t = 1, 2, \dots, \tau \right\}$$

is the natural extension of the original uncertainty set.

Notice that Lemma 3 implies that  $\mathbf{x} \in X_{\text{EAARC}}(\Lambda^0)$  if and only if zero is the optimal value of the following problem:

$$\begin{aligned} \min \quad & -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle \\ \text{s.t.} \quad & \mathbf{Z}\boldsymbol{\beta}' + \sum_{t=1}^{\tau} \mathbf{U}^t (\boldsymbol{\gamma}^t)' \leq \mathbf{d}\boldsymbol{\alpha}', \\ & \mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \\ & \boldsymbol{\beta} = \boldsymbol{\gamma}^0, \\ & \mathbf{B}'\boldsymbol{\gamma}^t = \mathbf{0}, \quad t = 0, 1, \dots, \tau, \\ & \|(\boldsymbol{\gamma}^t)'\|_{(t)} \leq \Omega_t \boldsymbol{\alpha}', \quad t = 0, 1, \dots, \tau, \\ & \boldsymbol{\alpha} \geq \mathbf{0}. \end{aligned} \tag{14}$$

Here  $\boldsymbol{\alpha} \in \mathbb{R}^r$ ,  $\boldsymbol{\beta} \in \mathbb{R}^{r \times N}$ ,  $\boldsymbol{\gamma}^t \in \mathbb{R}^{r \times K_t}$ , and  $\|(\boldsymbol{\gamma}^t)'\|_{(t)}$  is an  $r$ -dimensional row vector with each entry equal to the norm of the corresponding column in  $(\boldsymbol{\gamma}^t)'$ .

Similarly,  $\mathbf{x} \in X_{\text{EAARC}}$  if and only if zero is the optimal value of the following problem:

$$\begin{aligned} \min \quad & -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle \\ \text{s.t.} \quad & \mathbf{Z}\boldsymbol{\beta}' + \sum_{t=1}^{\tau} \mathbf{U}^t (\boldsymbol{\eta}^t - \boldsymbol{\delta}^t)' \leq \mathbf{d}\boldsymbol{\alpha}', \\ & \mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \\ & \boldsymbol{\beta} = \boldsymbol{\eta}^0 - \boldsymbol{\delta}^0, \\ & \mathbf{B}'\boldsymbol{\eta}^t = \mathbf{0}, \quad t = 0, 1, \dots, \tau, \\ & \mathbf{B}'\boldsymbol{\delta}^t = \mathbf{0}, \quad t = 0, 1, \dots, \tau, \\ & \|(\boldsymbol{\eta}^t)' + (\boldsymbol{\delta}^t)'\|_{(t)} \leq \Omega_t \boldsymbol{\alpha}', \quad t = 0, 1, \dots, \tau, \\ & \boldsymbol{\alpha}, \boldsymbol{\eta}^t, \boldsymbol{\delta}^t \geq \mathbf{0}, \quad t = 0, 1, \dots, \tau. \end{aligned} \tag{15}$$

Here  $\boldsymbol{\eta}^t, \boldsymbol{\delta}^t \in \mathbb{R}^{r \times K_t}$ .

It is straightforward to see that for any given feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  of problem (15),  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$  with  $\boldsymbol{\gamma}' = \boldsymbol{\eta}' - \boldsymbol{\delta}'$  is feasible for problem (14). If, in addition, for any given feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$  of problem (14), we can find  $(\boldsymbol{\eta}, \boldsymbol{\delta})$  such that  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  is feasible for problem (15), then  $X_{\text{EAARC}}(\Lambda) = X_{\text{EAARC}}(\Lambda^0)$ . However, if the projection of the feasible set of problem (15) onto the  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  space is a true subset of the projection of the feasible set of problem (14) onto the  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  space, then it is possible that  $X_{\text{EAARC}}(\Lambda^0)$  and thus  $X_{\text{AARC}}$  are true subsets of  $X_{\text{EAARC}}(\Lambda)$ . In the following, we compare the two sets under several different cases. First, we assume that  $\Omega_t$  is finite. The following theorem illustrates that, when we use the infinity norm in the uncertainty set, the EAARC with the extended uncertainty set  $\Lambda$  does not improve upon the EAARC with the extended uncertainty set  $\Lambda^0$ .

**THEOREM 5.** *If  $\|\cdot\|_{(t)} = \|\cdot\|_{\infty}$  and  $\Omega_t < \infty$  for all  $t = 0, 1, \dots, \tau$ , then the projection of the feasible set of problem (15) onto the  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  space coincides with the projection of the feasible set of problem (14) onto the  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  space. Thus, in this case,  $X_{\text{EAARC}}(\Lambda) = X_{\text{EAARC}}(\Lambda^0)$ .*

**PROOF.** For any feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$  of problem (14), we have that  $\|(\boldsymbol{\gamma}^t)'\|_{\infty} \leq \Omega_t \boldsymbol{\alpha}'$ . Define for  $t = 0, 1, \dots, \tau$ ,

$$\boldsymbol{\eta}^t = \frac{1}{2}(\Omega_t \mathbf{e}\boldsymbol{\alpha}' + \boldsymbol{\gamma}^t)$$

and

$$\boldsymbol{\delta}^t = \frac{1}{2}(\Omega_t \mathbf{e}\boldsymbol{\alpha}' - \boldsymbol{\gamma}^t).$$

It is straightforward to check that  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  is feasible for problem (15) and gives the same objective value. Thus,  $X_{\text{EAARC}}(\Lambda) = X_{\text{EAARC}}(\Lambda^0)$ .  $\square$

The above proof can be easily extended to the case in which  $\Omega_t = \infty$  for any  $t = 0, 1, \dots, \tau$ .

**THEOREM 6.** *Assume that  $\Omega_t = \infty$  for all  $t = 0, 1, \dots, \tau$ . If  $\Lambda^0$  is bounded, then  $X_{\text{EAARC}}(\Lambda) = X_{\text{EAARC}}(\Lambda^0)$ .*

We now present an example to show that Theorem 6 may fail if  $\Lambda^0$  is not bounded.

**EXAMPLE 3.** Let  $r = N = 2$ ,  $\Lambda^0 = \{(z_1, z_2, u_1, u_2): z_1 - z_2 \leq 1, \mathbf{z} = \mathbf{u}\}$ , and

$$\mathbf{B} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

In this case, it is clear that  $\mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \boldsymbol{\alpha} \geq \mathbf{0}$  implies that  $\boldsymbol{\alpha} = \mathbf{0}$ . Thus, problem (15) has a unique solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta}) = \mathbf{0}$ . However, in addition to the feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \mathbf{0}$ , problem (14) has a nonzero feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ , in which  $\boldsymbol{\alpha} = \mathbf{0}, \boldsymbol{\beta} = \boldsymbol{\gamma}$  and

$$\boldsymbol{\beta} = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}.$$

Therefore, the feasible set of problem (14) is a strict subset of the feasible set of problem (15), and thus Theorem 6 does not hold.

We now show that Theorem 5 fails if the norm is different from the infinity norm. For this purpose, we need the following result.

LEMMA 7. *Given a norm  $\|\cdot\|$  with the property that  $\|\mathbf{u}\| = \|\|\mathbf{u}\|\|$  for any  $\mathbf{u}$ ,*

$$\|\mathbf{u}\| \leq \|\mathbf{v}\| \quad \forall \mathbf{0} \leq \mathbf{u} \leq \mathbf{v}.$$

PROOF. Let  $\mathbf{e}^i$  be the unit vector with its  $i$ th component being one. It suffices to show that

$$\|\mathbf{u}\| \leq \|\mathbf{v}\|$$

for any  $\mathbf{u} \geq \mathbf{0}$  and  $\mathbf{v} = \mathbf{u} + \gamma \mathbf{e}^i$  for any  $i$  and  $\gamma \geq 0$ .

Define a new vector  $\hat{\mathbf{v}}$  such that

$$\hat{\mathbf{v}} = \mathbf{u} - (2u_i + \gamma)\mathbf{e}^i.$$

It is clear that  $\|\hat{\mathbf{v}}\| = \|\mathbf{v}\|$ . In addition,  $\mathbf{u}$  lies within the line segment between  $\hat{\mathbf{v}}$  and  $\mathbf{v}$ . Thus,

$$\|\mathbf{u}\| \leq \max\{\|\hat{\mathbf{v}}\|, \|\mathbf{v}\|\} = \|\mathbf{v}\|. \quad \square$$

In the following, we further assume that the vector norm  $\|\cdot\|_{(t)}$  satisfies the following conditions:

$$\|\mathbf{e}^i\|_{(t)} = 1 \quad \forall i.$$

THEOREM 8. *If  $\|\cdot\|_{(t)} \neq \|\cdot\|_\infty$  and  $\Omega_t$  is finite for some  $t$ , then  $X_{\text{EAARC}}(\Lambda)$  may be a true subset of  $X_{\text{EAARC}}(\Lambda^0)$ . In this case, the EAARC based on the extended uncertainty set  $\Lambda$  may provide a strict improvement upon the AARC.*

PROOF. We prove this result by constructing an example. Specifically, we construct an example in which, for a feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$  of problem (14), we cannot find  $(\boldsymbol{\eta}, \boldsymbol{\delta})$  such that  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  is feasible for problem (15). We consider the basic setting in which

$$\Lambda^0 = \{(\mathbf{z}, \mathbf{u}): L\mathbf{z} \leq \mathbf{1}, \mathbf{z} = \mathbf{u}, \|\mathbf{u}\| \leq \Omega\}.$$

In this case, we have for a feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$  of problem (14),  $\boldsymbol{\gamma} = \boldsymbol{\beta}$ . Thus, it suffices to talk about the feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  of problem (14).

Let  $\Omega = \|[1 \ 1]\|$ . Choose  $\mathbf{B}$  such that the null space of  $\mathbf{B}'$  is spanned by  $\boldsymbol{\alpha} = [1 \ 1 \ 1 \ 1]'$ ,  $\boldsymbol{\beta}^1$  and  $\boldsymbol{\beta}^2$ , where

$$\boldsymbol{\beta} = [\boldsymbol{\beta}^1, \boldsymbol{\beta}^2] = \begin{bmatrix} 1 & 1 \\ \Omega & 0 \\ 0 & \Omega \\ -\Omega & 0 \\ 0 & -\Omega \end{bmatrix}.$$

It is easy to verify that  $(\boldsymbol{\alpha}, \boldsymbol{\beta})$  is feasible for problem (14). Now assume that there exist  $\boldsymbol{\eta}$  and  $\boldsymbol{\delta}$  such that

$(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  is feasible for problem (15). Because  $\mathbf{B}'\boldsymbol{\delta} = \mathbf{0}$ , we have that

$$\boldsymbol{\delta}^j = \omega_j \boldsymbol{\alpha} + \mu_j \boldsymbol{\beta}^1 + \nu_j \boldsymbol{\beta}^2, \quad j = 1, 2,$$

for some scalars  $\omega_j, \mu_j$ , and  $\nu_j$ . Because  $\boldsymbol{\beta}' = \boldsymbol{\eta}' - \boldsymbol{\delta}'$ , we have that  $\boldsymbol{\eta} = [\boldsymbol{\eta}^1 \ \boldsymbol{\eta}^2]'$  with

$$\boldsymbol{\eta}^1 = \omega_1 \boldsymbol{\alpha} + (1 + \mu_1) \boldsymbol{\beta}^1 + \nu_1 \boldsymbol{\beta}^2$$

and

$$\boldsymbol{\eta}^2 = \omega_2 \boldsymbol{\alpha} + \mu_2 \boldsymbol{\beta}^1 + (1 + \nu_2) \boldsymbol{\beta}^2.$$

Thus, the  $j$ th column of  $\boldsymbol{\eta}' + \boldsymbol{\delta}'$  is given by

$$\begin{bmatrix} 2\omega_1 + 2\mu_1\beta_j^1 + 2\nu_1\beta_j^2 + \beta_j^1 \\ 2\omega_2 + 2\mu_2\beta_j^1 + 2\nu_2\beta_j^2 + \beta_j^1 \end{bmatrix}.$$

Because  $\|\boldsymbol{\eta}' + \boldsymbol{\delta}'\| \leq \Omega \boldsymbol{\alpha}_j$  and  $\boldsymbol{\delta} \geq \mathbf{0}$ , letting  $j = 1$  implies that

$$\omega_j + \mu_j + \nu_j = 0 \quad \text{for } j = 1, 2.$$

Similarly, letting  $j = 2, 3$  implies that

$$\omega_1 + \Omega \mu_1 = 0, \quad \omega_2 + \Omega \nu_2 = 0.$$

The above equalities imply that

$$\omega_1 = -\Omega \mu_1, \quad \nu_1 = (\Omega - 1) \mu_1,$$

and

$$\omega_2 = -\Omega \nu_2, \quad \mu_2 = (\Omega - 1) \nu_2.$$

Letting  $j = 4, 5$ ,  $\boldsymbol{\eta} \geq \mathbf{0}$  implies that  $-\mu_1 \geq 1/2$  and  $-\nu_2 \geq 1/2$ . In addition, we have that

$$\left\| \begin{bmatrix} -4\mu_1\Omega - \Omega \\ -2\Omega^2\nu_2 \end{bmatrix} \right\| \leq \Omega, \quad \left\| \begin{bmatrix} -2\Omega^2\mu_1 \\ -4\nu_2\Omega - \Omega \end{bmatrix} \right\| \leq \Omega. \quad (16)$$

However,  $-4\mu_1\Omega - \Omega \geq \Omega$  and  $-2\Omega^2\nu_2 \geq \Omega^2$ . The above inequalities together with Lemma 7 imply that  $\|[1 \ \Omega]\| \leq 1$ . Hence,  $\Omega = 1$  and  $\|[1 \ 1]\| = 1$ . Again, this together with Lemma 7 implies that  $\|(u_1, u_2)'\| = 1$  if and only if  $\|(u_1, u_2)'\| = \|(u_1, u_2)'\|_\infty$  for any  $u_1$  and  $u_2$ .  $\square$

In the construction of the uncertainty set of the EAARC, we essentially split the primitive uncertainty  $\mathbf{z}$  and the auxiliary variable  $\mathbf{u}$  to the positive parts and negative parts. We may generalize the idea by further splitting  $\mathbf{u}$  to more parts as follows:

$$\Lambda(K) = \left\{ (\mathbf{z}, \mathbf{v}, \mathbf{w}): \mathbf{Z}\mathbf{z} + \sum_{t=1}^{\tau} \sum_{k=1}^K \mathbf{U}^t(\mathbf{v}^{t,k} - \mathbf{w}^{t,k}) \leq \mathbf{d}, \mathbf{z} = \sum_{k=1}^K (\mathbf{v}^{0,k} - \mathbf{w}^{0,k}) \right\} \cap \bigcap_{t=0}^{\tau} \mathcal{G}_t(K),$$



where  $\mathbf{v}^t = [(\mathbf{v}^{t,1})' \dots (\mathbf{v}^{t,K})']'$ ,  $\mathbf{w}^t = [(\mathbf{w}^{t,1})' \dots (\mathbf{w}^{t,K})']'$ , and for  $t = 0, 1, \dots, \tau$ ,

$$\mathcal{G}_t(K) = \left\{ (\mathbf{z}, \mathbf{v}, \mathbf{w}): \left\| \sum_{k=1}^K (\mathbf{v}^{t,k} + \mathbf{w}^{t,k}) \right\|_{(t)} \leq \Omega_t, \right. \\ \left. \mathbf{0} \leq \mathbf{v}^{t,k} \leq \mathbf{a}^{t,k}, \mathbf{0} \leq \mathbf{w}^{t,k} \leq \mathbf{b}^{t,k}, k = 1, \dots, K \right\}.$$

One may conjecture that by introducing more flexibility into the uncertainty set, we can make further improvement. For a fair comparison, we require that for  $t = 0, 1, \dots, \tau$ ,

$$\left\{ \xi = \sum_{k=1}^K (\mathbf{v}^{t,k} - \mathbf{w}^{t,k}): (\mathbf{z}, \mathbf{v}, \mathbf{w}) \in \mathcal{G}_t(K) \right\} = \{ \xi: \|\xi\|_{(t)} \leq \Omega_t \}. \quad (17)$$

Unfortunately, under these assumptions,  $\Lambda(K)$  may not provide any improvement over  $\Lambda(1)$ . To see this, we consider the dual associated with the uncertainty set  $\Lambda(K)$ , which can be written as follows:

$$P(K): \min -\langle \mathbf{m}^0(\mathbf{x}), \boldsymbol{\alpha} \rangle - \langle \mathbf{M}(\mathbf{x}), \boldsymbol{\beta} \rangle \\ \text{s.t. } \mathbf{Z}\boldsymbol{\beta}' + \sum_{t=1}^{\tau} \sum_{k=1}^K \mathbf{U}^t(\boldsymbol{\eta}^{t,k} - \boldsymbol{\delta}^{t,k})' \leq \mathbf{d}\boldsymbol{\alpha}', \\ \mathbf{B}'\boldsymbol{\alpha} = \mathbf{0}, \\ \boldsymbol{\beta} = \sum_{k=1}^K (\boldsymbol{\eta}^{0,k} - \boldsymbol{\delta}^{0,k}), \\ \mathbf{B}'\boldsymbol{\eta}^{t,k} = \mathbf{0}, \\ t = 0, 1, \dots, \tau, k = 1, 2, \dots, K, \\ \mathbf{B}'\boldsymbol{\delta}^{t,k} = \mathbf{0}, \\ t = 0, 1, \dots, \tau, k = 1, 2, \dots, K, \\ \left\| \sum_{k=1}^K ((\boldsymbol{\eta}^{t,k})' + (\boldsymbol{\delta}^{t,k})') \right\|_{(t)} \leq \Omega_t \boldsymbol{\alpha}', \\ t = 0, 1, \dots, \tau, \\ \mathbf{0} \leq \boldsymbol{\eta}^{t,k} \leq \mathbf{a}^{t,k} \boldsymbol{\alpha}', \\ t = 0, 1, \dots, \tau, k = 1, 2, \dots, K, \\ \mathbf{0} \leq \boldsymbol{\delta}^{t,k} \leq \mathbf{b}^{t,k} \boldsymbol{\alpha}', \\ t = 0, 1, \dots, \tau, k = 1, 2, \dots, K, \\ \boldsymbol{\alpha} \geq \mathbf{0}.$$

It is obvious that for any feasible solution of problem (18) for general  $K$ , we can construct a feasible solution for problem (15) with the same objective value.

On the other hand, because (17) holds, we claim that

$$\sum_{k=1}^K \mathbf{a}^{t,k} \geq \Omega_t \mathbf{e}, \quad \sum_{k=1}^K \mathbf{b}^{t,k} \geq \Omega_t \mathbf{e}.$$

Indeed, because  $\|\Omega_t \mathbf{e}^i\|_{(t)} = \Omega_t$ , there exist  $\mathbf{v}^{t,k}$  and  $\mathbf{w}^{t,k}$  such that

$$\mathbf{0} \leq \mathbf{v}^{t,k} \leq \mathbf{a}^{t,k}, \mathbf{0} \leq \mathbf{w}^{t,k} \leq \mathbf{b}^{t,k}, \left\| \sum_{k=1}^K (\mathbf{v}^{t,k} + \mathbf{w}^{t,k}) \right\|_{(t)} \leq \Omega_t,$$

and  $\sum_{t=1}^K (\mathbf{v}^{t,k} - \mathbf{w}^{t,k}) = \Omega_t \mathbf{e}^i$ . Because  $\mathbf{v}^{t,k}, \mathbf{w}^{t,k} \geq \mathbf{0}$ ,

$$\sum_{k=1}^K a_i^{t,k} \geq \sum_{k=1}^K v_i^{t,k} \geq \Omega_t.$$

Thus,  $\sum_{k=1}^K \mathbf{a}^{t,k} \geq \Omega_t \mathbf{e}$ . Similarly, we can show that  $\sum_{k=1}^K \mathbf{b}^{t,k} \geq \Omega_t \mathbf{e}$ . For any feasible solution  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  of problem (15), Lemma 7 implies that

$$\mathbf{0} \leq (\boldsymbol{\eta}^j)' \leq \Omega \boldsymbol{\alpha}', \quad \mathbf{0} \leq (\boldsymbol{\delta}^j)' \leq \Omega \boldsymbol{\alpha}'.$$

Therefore, there exists  $\rho_k^i \geq 0$  and  $\phi_k^i \geq 0$  such that

$$0 \leq (\boldsymbol{\eta}_i^{t,k})' = \psi_i^{t,k} (\boldsymbol{\eta}_i^t)' \leq \psi_i^{t,k} \Omega_t \boldsymbol{\alpha}' \leq a_i^{t,k} \boldsymbol{\alpha}', \quad \sum_{k=1}^K \psi_i^{t,k} = 1,$$

and

$$0 \leq (\boldsymbol{\delta}_i^{t,k})' = \phi_i^{t,k} (\boldsymbol{\delta}_i^t)' \leq \phi_i^{t,k} \Omega_t \boldsymbol{\alpha}' \leq b_i^{t,k} \boldsymbol{\alpha}', \quad \sum_{k=1}^K \phi_i^{t,k} = 1.$$

Hence,  $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\delta})$  is feasible for problem (18) for general  $K$ , which implies that further splitting  $(\mathbf{z}, \mathbf{u})$  does not provide an improvement.

## 5. Numerical Experiment

In the previous section, we proposed one way of choosing the uncertainty set in the EAARC and identified conditions under which the EAARC improves upon the AARC. In this section, we conduct numerical experiments to illustrate the improvement on a project management problem.

A project management problem can be represented by a directed graph with  $m$  arcs and  $n$  nodes. Each node on the graph represents an event marking the completion of a particular subset of activities. We denote the set of directed arcs on the graph as  $E$ . Hence, an arc  $(i, j) \in E$  is an activity that connects event  $i$  to event  $j$ . By convention, we use node 1 as the start event and the last node  $n$  as the end event.

We consider a project with several activities. The completion of activities must satisfy precedent constraints. For example, activity  $e_1$  precedes activity  $e_2$  if activity  $e_1$  must be completed before starting activity  $e_2$ .

Each activity  $(i, j) \in E$  has an uncertain duration  $t_{ij} + z_{ij}\epsilon_{ij}$  in which  $t_{ij}$  and  $\epsilon_{ij}$  are constants and  $z_{ij} \in [-1, 1]$  is the primitive uncertainty. The value of  $z_{ij}$  is realized after event  $i$  is completed. But before this realization, certain resources can be allocated to the activity to shorten its duration. Specifically, we assume that if  $y_{ij}$  units of resource are allocated to activity  $(i, j) \in E$ , then the duration of activity  $(i, j)$  would become  $t_{ij} + z_{ij}\epsilon_{ij} - y_{ij}$ . Let  $b_{ij}$  be the cost of using each unit of resource for the activity on the arc  $(i, j)$ . Our goal is to find a trade-off of the completion time of the project and the total cost of resource allocations.

Mathematically, our project management problem can be formulated as a multistage uncertain linear program:

$$\left\{ \begin{array}{l} \min \sum_{ij} b_{ij}y_{ij} + Cx_n \\ \text{s.t. } x_j - x_i + y_{ij} - z_{ij}\epsilon_{ij} \geq t_{ij} \quad \forall (i, j) \in E, \\ y_{ij} \geq 0 \quad \forall (i, j) \in E, \\ t_{ij} + z_{ij}\epsilon_{ij} - y_{ij} \geq M_{ij} \quad \forall (i, j) \in E, \\ x_1 = 0, \\ x_n \leq D, \end{array} \right\} \quad \forall z_{ij}.$$

In this model,  $C$  is the per-unit cost on the completion time, and  $x_i$  denotes the completion time of event  $i$ . The first constraint implies that the completion time of event  $j$  is no less than the completion time of event  $i$  plus the completion time of activity  $(i, j)$ . The third constraint requires that the reduction of the completion time of an activity cannot be arbitrarily large. In particular, in our experiment, we assume that the minimum duration of project  $(i, j) \in E$  must be at least  $M_{ij}$ . We also require that the completion time of the entire project meets a strict deadline  $D$ .

If the distributional information of the uncertain data is available, one would formulate the problem as a multistage stochastic programming problem. Unfortunately, analysis of the project management problem within the stochastic programming framework, such as determining the expected completion time and quantile of completion time, is notoriously difficult (Hagstrom 1988). A tractable approximation is proposed by Chen et al. (2008) to a two-stage project management problem with uncertainty, which requires mild distributional knowledge of the uncertain completion time  $t_{ij}$ .

In our experiment, instead of imposing distributional assumptions on the uncertain data, we assume that the uncertain data are restricted within some uncertainty set and formulate the project management problem within the adjustable robust counterpart framework. Specifically, the uncertainty set  $\Gamma$  is defined as follows:

$$\Gamma = \{ \mathbf{z} = (z_{ij})_{(i,j) \in E} : -\bar{\mathbf{w}} \leq \mathbf{z} \leq \bar{\mathbf{v}}, \|\mathbf{z}\|_2 \leq \Omega \}.$$

We will compare the performance of the AARC and the splitting-based EAARC on the multistage project management problem. In the splitting-based EAARC, we define the uncertainty set  $\Lambda$  as follows:

$$\Lambda = \{ (\mathbf{z}, \mathbf{v}, \mathbf{w}) = (z_{ij}, v_{ij}, w_{ij})_{(i,j) \in E} : \mathbf{z} = \mathbf{v} - \mathbf{w}, -\bar{\mathbf{w}} \leq \mathbf{v} - \mathbf{w} \leq \bar{\mathbf{v}}, \|\mathbf{v} + \mathbf{w}\|_2 \leq \Omega, (\mathbf{v}, \mathbf{w}) \geq \mathbf{0} \}.$$

In addition, the decision variables  $x_i$  and  $y_{ij}$  are represented as

$$\begin{aligned} x_i &= x_i^0 + \sum_{(k,l) \in I_i} x_i^{kl,v} v_{kl} + \sum_{(k,l) \in I_i} x_i^{kl,w} w_{kl}, \\ y_{ij} &= y_{ij}^0 + \sum_{(k,l) \in I_i} y_{ij}^{kl,v} v_{kl} + \sum_{(k,l) \in I_i} y_{ij}^{kl,w} w_{kl}, \end{aligned} \tag{19}$$

where  $v_{kl}$  and  $w_{kl}$  can be regarded as the positive part and negative part of the primitive uncertainty  $z_{kl}$ , respectively. It is clear that, when we impose the constraints  $x_i^{kl,v} = -x_i^{kl,w}$  and  $y_{ij}^{kl,v} = -y_{ij}^{kl,w}$ , the EAARC reduces to the AARC.

Note that in the above formulations, instead of summing  $(k, l)$  across all arcs, we take the summation only in a selected set  $I_i$ . The set  $I_i$  is called the *information set* for decision variable  $x_i$ . By choosing the information set properly, we ensure that, at any stage of decision, the system takes into account only information from previous realized uncertainties. Furthermore, based on our assumption that resource on an arc is allocated before the primitive uncertainty on that arc is realized, arc  $(i, j)$  should always have the same information set as node  $i$ .

Employing Theorem 4, the splitting-based EAARC can be reformulated as a second-order conic program. However, it seems more convenient to carry out the reformation using the following result, which is shown in Chen et al. (2008).

LEMMA 9. For a given scalar  $\alpha$  and a vector  $\mathbf{a}$ , the robust constraint

$$\alpha + \mathbf{a}'\mathbf{v} + \mathbf{b}'\mathbf{w} \leq 0 \quad \forall (\mathbf{v}, \mathbf{w}) \in \{ (\mathbf{v}, \mathbf{w}) : -\bar{\mathbf{w}} \leq \mathbf{v} - \mathbf{w} \leq \bar{\mathbf{v}}, \|\mathbf{v} + \mathbf{w}\|_2 \leq \Omega, (\mathbf{v}, \mathbf{w}) \geq \mathbf{0} \}$$

can be equivalently written as

$$\begin{aligned} \alpha + \Omega \|\mathbf{u}\|_2 + \mathbf{r}'\bar{\mathbf{v}} + \mathbf{s}'\bar{\mathbf{w}} &\leq 0, \\ u_j &\geq a_j - r_j + s_j \quad \forall j, \\ u_j &\geq b_j + r_j - s_j \quad \forall j, \\ \mathbf{u}, \mathbf{r}, \mathbf{s} &\geq \mathbf{0}. \end{aligned}$$

Because all of the constraints in the EAARC have the same form as the robust constraint in Lemma 9, they are referred to as robust constraints in the sequel, and we will use Lemma 9 to reformulate all of the robust constraints into their equivalent second-order conic constraints. But before we do this, note that any primitive uncertainty not in the information set should not have an influence on the corresponding decision variable; therefore, we have the following constraints:

$$\begin{aligned} x_i^{kl,v} &= x_i^{kl,w} = 0 \quad \forall (k, l) \notin I_i, \\ y_{ij}^{kl,v} &= y_{ij}^{kl,w} = 0 \quad \forall (k, l) \notin I_{ij}. \end{aligned}$$

Similarly the constraint  $x_1 = 0$  means  $x_0^{kl,v} = x_0^{kl,w} = 0 \quad \forall (k, l)$ .

The objective of minimizing  $\sum_{ij} b_{ij}y_{ij} + Cx_n$  can be written as minimizing a new variable  $\tau$ , subject to the robust constraint  $\tau \geq \sum_{ij} b_{ij}y_{ij} + Cx_n$ . Given the representation of  $x_i$  and  $y_{ij}$  in (19) and the extended uncertainty set  $\Lambda$ , this robust constraint, together with the four other sets of robust constraints, is turned into their equivalent second-order conic constraints. The reformulations are presented below for convenience.

**First set:**

$$\tau \geq \sum_{ij} b_{ij} y_{ij} + Cx_n$$

becomes

$$\tau \geq Cx_n^0 + \sum_{(i,j) \in E} b_{ij} y_{ij}^0 + \Omega \|t\|_2 + \sum_{(k,l) \in E} (t_v^{kl} \bar{v}_{kl} + t_w^{kl} \bar{w}_{kl}),$$

$$t^{kl} \geq \sum_{(i,j) \in E} b_{ij} y_{ij}^{kl,v} + Cx_n^{kl,v} - t_v^{kl} + t_w^{kl} \quad \forall (k,l) \in E,$$

$$t^{kl} \geq \sum_{(i,j) \in E} b_{ij} y_{ij}^{kl,w} + Cx_n^{kl,w} + t_v^{kl} - t_w^{kl} \quad \forall (k,l) \in E,$$

$$t, t_v, t_w \geq 0,$$

where  $t = (t^{kl})_{(k,l) \in E}$ ,  $t_v = (t_v^{kl})_{(k,l) \in E}$ , and  $t_w = (t_w^{kl})_{(k,l) \in E}$ .

**Second set:**

$$x_j - x_i + y_{ij} - z_{ij} \epsilon_{ij} \geq t_{ij} \quad \forall (i,j) \in E$$

becomes

$$x_i^0 - x_j^0 - y_{ij}^0 + t_{ij} + \Omega \|\gamma_{ij}\|_2 + \sum_{(k,l) \in E} (\gamma_{v,ij}^{kl} \bar{v}_{kl} + \gamma_{w,ij}^{kl} \bar{w}_{kl}) \leq 0 \quad \forall (i,j) \in E,$$

$$\gamma_{ij}^{kl} \geq x_i^{kl,v} - x_j^{kl,v} - y_{ij}^{kl,v} + \epsilon_{ij} \delta_{ij}^{kl} - \gamma_{v,ij}^{kl} + \gamma_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\gamma_{ij}^{kl} \geq x_i^{kl,w} - x_j^{kl,w} - y_{ij}^{kl,w} - \epsilon_{ij} \delta_{ij}^{kl} + \gamma_{v,ij}^{kl} - \gamma_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\gamma_{ij}, \gamma_{v,ij}, \gamma_{w,ij} \geq 0 \quad \forall (i,j) \in E,$$

where  $\gamma_{ij} = (\gamma_{ij}^{kl})_{(k,l) \in E}$ ,  $\gamma_{v,ij} = (\gamma_{v,ij}^{kl})_{(k,l) \in E}$ ,  $\gamma_{w,ij} = (\gamma_{w,ij}^{kl})_{(k,l) \in E}$ , and  $\delta_{ij}^{kl} = 1$  if  $(i,j) = (k,l)$  and 0 otherwise.

**Third set:**

$$y_{ij} \geq 0 \quad \forall (i,j) \in E$$

becomes

$$-y_{ij}^0 + \Omega \|\alpha_{ij}\|_2 + \sum_{(k,l) \in E} (\alpha_{v,ij}^{kl} \bar{v}_{kl} + \alpha_{w,ij}^{kl} \bar{w}_{kl}) \leq 0 \quad \forall (i,j) \in E,$$

$$\alpha_{ij}^{kl} \geq -y_{ij}^{kl,v} - \alpha_{v,ij}^{kl} + \alpha_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\alpha_{ij}^{kl} \geq -y_{ij}^{kl,w} + \alpha_{v,ij}^{kl} - \alpha_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\alpha_{ij}, \alpha_{v,ij}, \alpha_{w,ij} \geq 0 \quad \forall (i,j) \in E,$$

where  $\alpha_{ij} = (\alpha_{ij}^{kl})_{(k,l) \in E}$ ,  $\alpha_{v,ij} = (\alpha_{v,ij}^{kl})_{(k,l) \in E}$ , and  $\alpha_{w,ij} = (\alpha_{w,ij}^{kl})_{(k,l) \in E}$ .

**Fourth set:**

$$t_{ij} + z_{ij} \epsilon_{ij} - y_{ij} \geq M_{ij} \quad \forall (i,j) \in E$$

becomes

$$y_{ij}^0 + \Omega \|\beta_{ij}\|_2 + \sum_{(k,l) \in E} (\beta_{v,ij}^{kl} \bar{v}_{kl} + \beta_{w,ij}^{kl} \bar{w}_{kl}) \leq t_{ij} - M_{ij} \quad \forall (i,j) \in E,$$

$$\beta_{ij}^{kl} \geq y_{ij}^{kl,v} - \epsilon_{ij} \delta_{ij}^{kl} - \beta_{v,ij}^{kl} + \beta_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\beta_{ij}^{kl} \geq y_{ij}^{kl,w} + \epsilon_{ij} \delta_{ij}^{kl} + \beta_{v,ij}^{kl} - \beta_{w,ij}^{kl} \quad \forall (i,j) \in E, (k,l) \in E,$$

$$\beta_{ij}, \beta_{v,ij}, \beta_{w,ij} \geq 0 \quad \forall (i,j) \in E,$$

where  $\beta_{ij} = (\beta_{ij}^{kl})_{(k,l) \in E}$ ,  $\beta_{v,ij} = (\beta_{v,ij}^{kl})_{(k,l) \in E}$ , and  $\beta_{w,ij} = (\beta_{w,ij}^{kl})_{(k,l) \in E}$ .

**Fifth set:**

$$x_n \leq D$$

becomes

$$x_n^0 + \Omega \|\psi\|_2 + \sum_{(k,l) \in E} (\psi_v^{kl} \bar{v}_{kl} + \psi_w^{kl} \bar{w}_{kl}) \leq D,$$

$$\psi^{kl} \geq x_n^{kl,v} - \psi_v^{kl} + \psi_w^{kl} \quad \forall (k,l) \in E,$$

$$\psi^{kl} \geq -x_n^{kl,w} + \psi_v^{kl} - \psi_w^{kl} \quad \forall (k,l) \in E,$$

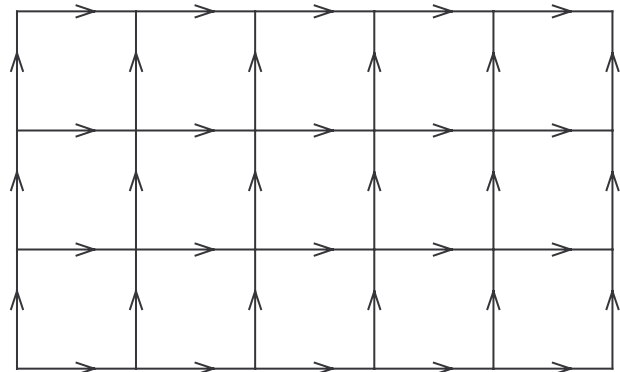
$$\psi, \psi_v, \psi_w \geq 0,$$

where  $\psi = (\psi^{kl})_{(k,l) \in E}$ ,  $\psi_v = (\psi_v^{kl})_{(k,l) \in E}$ , and  $\psi_w = (\psi_w^{kl})_{(k,l) \in E}$ .

Putting all of the above together, we end up with a second-order conic program, for which we use CPLEX version 10 to solve. For our computational experiment, we create a fictitious project with the activity network in the form of a  $H$  by  $W$  grid (see Figure 1). There are a total of  $(H + 1) \times (W + 1)$  nodes, with the first node at the bottom left corner and the last node at the upper right corner. Each arc on the graph points either upward or to the right.

In an instance of the uncertain project management problem, the related parameters  $t_{ij}$ ,  $M_{ij}$ ,  $b_{ij}$  are generated randomly. Specifically, on horizontal arcs,  $t_{ij}$  is generated

**Figure 1.** Project management grid with height  $H = 3$  and width  $W = 5$ .



from  $U[6, 10]$  (the uniform distribution in  $[6, 10]$ ), the minimum duration time  $M_{ij}$  is generated from  $U[1, 5]$ , and the resource unit cost  $b_{ij}$  is generated from  $U[1, 10]$ . On the vertical arcs,  $t_{ij}$  is generated from  $U[4, 6]$ ,  $M_{ij}$  is generated from  $U[1, 3]$ , and  $b_{ij}$  is generated from  $U[1, 5]$ . Let  $\epsilon_{ij} = t_{ij} - M_{ij}$ . We also fix  $C = 0.3$  and let  $\bar{v}_{ij} = \rho_+$  and  $\bar{w}_{ij} = \rho_-$  for some positive constants  $\rho_+$  and  $\rho_-$  (the exact values of  $\rho_+$  and  $\rho_-$  will be specified later). It is clear that  $\rho_+$  and  $\rho_-$  measure the asymmetry of the primitive uncertainties.

We use two difference criteria to compare the performance of the EAARC and the AARC. In the first criterion, we measure the improvement of the optimal objective value of the EAARC relative to the AARC. In the second criterion, we compare the simulated average costs incurred using decision rules (19) derived from the EAARC and the AARC. Here is a precise description of how this is done:

- For an instance of the uncertain project management problem, we solve the EAARC and AARC, respectively, to derive decision rules (19).
- Generate 100 samples of  $(z_{ij})_{(i,j) \in E}$  from  $U[-\rho_-, \rho_+]$  for the instance of the uncertain project management problem.
- For each sample, compute the cost of the project management problem when the decisions  $x_i$  and  $y_{ij}$  are determined by the decision rules (19) derived from solving the EAARC and the AARC. The average costs of the EAARC and the AARC are then defined as the average of the corresponding costs of all samples. Because we assume that only the primitive uncertainty  $\mathbf{z}$  is observable, in the implementation of the EAARC decision rule, we let  $v_{kl} = \max(z_{kl}, 0)$  and  $w_{kl} = \max(-z_{kl}, 0)$ .
- Compute the percentage of improvement:

$$\text{percentage of improvement} = \frac{(\text{average cost of EAARC}) - (\text{average cost of AARC})}{(\text{average cost of AARC})} \times 100\%.$$

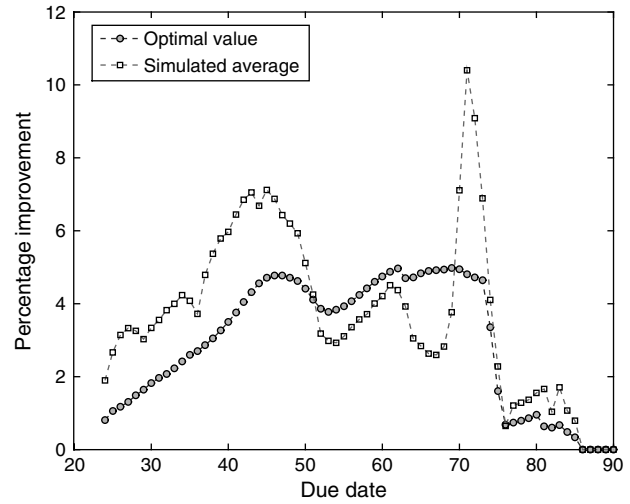
We now illustrate the impacts of the due date, the asymmetry property of the primitive uncertainties, problem size, information set, and level of robustness on the performance of the EAARC and the AARC.

**Experiment 1: Algorithm Improvement vs. Due-Date Constraint**

We use the  $3 \times 4$  grid network. Let  $\Omega = 3.0$ ,  $\rho_+ = 1$ , and  $\rho_- = 0.7$ . We also use the complete information set, i.e., for each event  $i$ , the information set  $I_i$  consists of the realization of all past primitive uncertainties. We pick a range of due dates  $D$  between 24 and 90. The percentage of improvement versus due-date relation is shown in Figure 2.

Our experiment indicates that there is a lower bound  $l$  on the due date ( $l = 24$  in our example), below which the due-date constraint would become so tight that both the EAARC and the AARC become infeasible. On the other hand, when the due date goes above an upper bound  $u$

**Figure 2.** Algorithm improvement vs. due date.



( $u = 85$  in our example), the time constraint becomes so loose that no project needs to be shortened, and, therefore, the EAARC and the AARC yield the same cost.

From Figure 2, we observe that the EAARC outperforms the AARC under both criteria (simulated average or optimal value). We also observe that the largest percentage of improvement always appears somewhere in the middle of  $l$  and  $u$ . For due dates near  $l$  or  $u$ , the costs derived from the EAARC and the AARC are close. The explanation is as follows: when the due date is too loose or too tight, the problem becomes somewhat simplified; i.e., all projects must be shortened (in the tight case) or no project needs to be shortened (in the loose case). In either case, the EAARC does not have a big advantage over the AARC.

Around the midpoint of  $l$  and  $u$ , however, it is not immediately clear which project to shorten and how much to shorten. This adds more variability to the problem, which would in turn demand additional flexibility in the response function. The seemingly erratic shape in the middle portion suggests that the percentage of improvement is sensitive to the due date. Especially, we observe that there is a big jump at a due date around 71 in the graph. One possible interpretation is that, when the due date becomes rather loose, the optimal costs of both EAARC and AARC decrease rapidly and thus their ratio and the percentage of improvement become unstable. This is in fact a common observation throughout our experiments. We also observe that not surprisingly the percentage of improvement under the two criteria demonstrates a certain degree of correlation. The percentage of improvement under the simulated average cost criterion, however, appears to be more volatile than the other.

**Experiment 2: Algorithm Improvement vs. Asymmetric Uncertainty Set**

As described before, our primitive uncertainty set can model asymmetric uncertainties by adjusting  $\rho_+$  and  $\rho_-$ .

**Table 1.** Algorithm improvement vs. asymmetric uncertainty.

| $\rho_-$      | 1   | 0.9 | 0.8 | 0.7 | 0.6 | 0.5 | 0.4 | 0.3 | 0.2 | 0.1 | 0 |
|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| Optimal (%)   | 0.0 | 2.3 | 3.7 | 4.5 | 4.8 | 4.9 | 4.6 | 3.9 | 2.9 | 0.5 | 0 |
| Simulated (%) | 0.0 | 2.1 | 2.9 | 3.4 | 3.5 | 3.5 | 3.3 | 2.9 | 2.3 | 1.4 | 0 |

We are interested in comparing the performances of the EAARC and the AARC under different levels of asymmetry. Without loss of generality, we fix  $\rho_+$  to be 1 and let  $\rho_-$  change. We set  $D = 60$  and still use the  $3 \times 4$  grid network with  $\Omega$  set to 3.0. The computational results are shown in Table 1.

Surprisingly, when  $\rho_- = 1$  (completely symmetric uncertainty set), the EAARC and the AARC always give the same cost in our experiment. On the other end, when  $\rho_- = 0$ , the uncertainty set lies completely in the positive orthant, and therefore EAARC reduces to AARC (and gives 0% improvement). When  $\rho_-$  takes value in the middle range, under both criteria, the percentage of improvement gets higher and peaks at around  $\rho_- = 0.5$ .

**Experiment 3: Algorithm Improvement vs. Problem Size**

We now evaluate the algorithm improvement with different problem size. To do this, three grid networks are selected with size  $2 \times 3$ ,  $3 \times 3$ , and  $3 \times 4$ , respectively. The due dates are set to be 25, 30, and 35, respectively. Again, let  $\Omega = 3.0$ ,  $\rho_+ = 1$ , and  $\rho_- = 0.7$ .

The percentages of improvements are listed in Table 2. From this table, it is clear that the improvements in both the optimal objective values and the simulation averages of using the EAARC grow when the problem size grows. Interestingly, the improvement of the simulated average cost outperforms that of the optimal objective value.

**Experiment 4: Algorithm Improvement vs. Information Set**

This experiment is carried out on a  $3 \times 4$  grid network with  $\Omega = 3.0$ ,  $D = 60$ ,  $\rho_+ = 1$ , and  $\rho_- = 0.7$ . In the experiment, we compare the performance of the EAARC and the AARC using the complete information set, in which all of the past information is available, and the partial information set, in which information too distant away in the past is “lost.”

To be more precise, we define the degree of information availability  $L$  as follows: for arc  $(i, j)$  to be in the information set for node  $k$ , activity  $(i, j)$  must complete before event  $k$ , and there is a path from event  $i$  to event  $k$  using

**Table 2.** Algorithm improvement vs. problem size.

| Size          | $3 \times 4$ | $4 \times 4$ | $4 \times 5$ |
|---------------|--------------|--------------|--------------|
| Optimal (%)   | 1.3          | 2.4          | 3.8          |
| Simulated (%) | 2.6          | 3.5          | 5.1          |

**Table 3.** Algorithm improvement vs. information set.

| $D$           | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   |
|---------------|-----|-----|-----|-----|-----|-----|-----|-----|
| Optimal (%)   | INF | 3.1 | 3.2 | 3.2 | 3.2 | 3.2 | 3.2 | 3.2 |
| Simulated (%) | INF | 4.4 | 4.6 | 4.6 | 4.6 | 4.6 | 4.6 | 4.7 |

no more than  $L$  arcs. In our experiment, we vary  $L$  from zero (information becomes lost immediately, e.g., no information is available) to 7 (for the  $3 \times 4$  grid, this means no information is lost).

Results are listed in Table 3 (“INF” stands for infeasible).

As we can easily observe, the percentage of improvement does not change much when information set shrinks. The explanation is as follows: the decision on a node depends heavily on the most “recent” information. Even though we are shrinking the information set, the most recent information is still kept. Therefore, the performance does not change much. When  $L = 0$ , there is essentially no information available, and both the EAARC and the AARC become infeasible easily.

To further justify our explanation, we have also tried an information set that includes all past information except the most recent. Both algorithms become infeasible frequently under this information set. This further confirms the intuition that, for our project management problem, decision in each stage relies mostly on recent information.

**Experiment 5: Algorithm Improvement vs. Level of Robustness ( $\Omega$ )**

This experiment is conducted on the  $3 \times 4$  network with  $D = 60$ ,  $\rho_+ = 1$ , and  $\rho_- = 0.7$ . We vary the values for  $\Omega$  to adjust the level of robustness, and we report the results in Table 4.

Clearly, the EAARC outperforms the AARC by larger percentages when  $\Omega$  is small. When  $\Omega$  grows large, we essentially put more value on robustness: both the EAARC and the AARC need to attain feasibility for a larger portion of primitive uncertainties of the problem. In this situation, the flexibility of the EAARC is confined; therefore, the percentage of improvement decreases as  $\Omega$  grows.

In summary, the EAARC improves upon the AARC, and its improvement depends on the tightness of the due date, the asymmetric property of the uncertainty set, the information set, the size of the problem, and the level of robustness. Specifically, our experiment demonstrates that the EAARC brings significant advantage over the AARC for due dates that are not too tight or too loose, information sets that

**Table 4.** Algorithm improvement vs. level of robustness.

| $\Omega$      | 2.0  | 2.5 | 3.0 | 3.5 | 4.0 |
|---------------|------|-----|-----|-----|-----|
| Optimal (%)   | 13.6 | 8.1 | 5.0 | 3.2 | 1.7 |
| Simulated (%) | 10.4 | 6.3 | 4.0 | 4.5 | 3.3 |

include the most recent history, larger problem size, and less stringent robustness. However, it is less clear how this improvement depends on the level of asymmetry of the uncertainty sets.

## 6. Conclusion

In this paper, we propose the extended affinely adjustable robust counterpart to modeling and solving a class of multistage uncertain linear programs with fixed recourse. Our approach ends up with well structured conic programming formulations, which are tractable and scalable to multistage problems and allow for large-scale implementation. We demonstrate both theoretically and computationally that the splitting-based extended affinely adjustable robust counterpart may significantly improve upon the affinely adjustable robust counterpart.

Our extended affinely adjustable robust counterpart is rather flexible. However, a significant challenge is how to choose an appropriate extended affinely decision rule. Specifically, the following question is of great interest: For a given constant  $\rho \geq 1$ , can we construct a tractable EAARC such that

$$X_{\text{EAARC}} \subseteq X_0 \subseteq \rho X_{\text{EAARC}}?$$

## Endnote

1. The approach presented here can be straightforwardly extended to multistage uncertain linear programming problems.

## Acknowledgments

The authors thank Melvyn Sim for stimulating discussions on robust optimization, especially the splitting-based EAARC. The authors also thank the associate editor and two reviewers for insightful comments and constructive suggestions, which greatly improved the presentation of this paper.

## References

- Atamtürk, A. 2006. Strong formulations of robust mixed 0-1 programming. *Math. Programming* **108**(2) 235–250.
- Atamtürk, A., M. Zhang. 2007. Two-stage robust network flow and design under demand uncertainty. *Oper. Res.* **55**(4) 662–673.
- Ben-Tal, A., A. Nemirovski. 1998. Robust convex optimization. *Math. Oper. Res.* **23** 769–805.
- Ben-Tal, A., A. Nemirovski. 1999. Robust solutions to uncertain programs. *Oper. Res. Lett.* **25** 1–13.
- Ben-Tal, A., A. Nemirovski. 2000. Robust solutions of linear programming problems contaminated with uncertain data. *Math. Programming* **88** 411–424.
- Ben-Tal, A., S. Boyd, A. Nemirovski. 2006. Extending the scope of robust optimization: Comprehensive robust counterparts of uncertain problems. *Math. Programming* **107** 63–89.
- Ben-Tal, A., T. Margalit, A. Nemirovski. 1999. Robust modeling of multi-stage portfolio problems. H. Frenk, K. Ross, T. Terlaky, S. Zhang, eds. *High Performance Optimization*, Chapter 12. Kluwer Academic Press, Rotterdam, The Netherlands, 303–328.
- Ben-Tal, A., B. Golany, A. Nemirovski, J. P. Vial. 2005. Retailer-supplier flexible commitments contracts: A robust optimization approach. *Manufacturing Service Oper. Management* **7**(3) 248–271.
- Ben-Tal, A., A. Goryashko, E. Guslitzer, A. Nemirovski. 2004. Adjusting robust solutions of uncertain linear programs. *Math. Programming* **99** 351–376.
- Bertsimas, D., C. Caramanis. 2005. Finite adaptability in linear optimization. Technical report, Massachusetts Institute of Technology, Cambridge, MA.
- Bertsimas, D., M. Sim. 2003. Robust discrete optimization and network flows. *Math. Programming* **98** 49–71.
- Bertsimas, D., M. Sim. 2004a. Price of robustness. *Oper. Res.* **52**(1) 35–53.
- Bertsimas, D., M. Sim. 2004b. Robust discrete optimization and downside risk measures. Working paper, National University of Singapore, Singapore.
- Bertsimas, D., M. Sim. 2006. Tractable approximations to robust conic optimization problems. *Math. Programming* **107**(1) 5–36.
- Bertsimas, D., C. Caramanis, W. Moser. 2006. Multistage finite adaptability: Application to air traffic control. Working paper, Massachusetts Institute of Technology, Cambridge, MA.
- Birge, J. R., F. Louveaux. 1997. *Introduction to Stochastic Programming*. Springer, New York.
- Chen, X., M. Sim, P. Sun. 2007. A robust optimization perspective on stochastic programming. *Oper. Res.* **55**(6) 1058–1071.
- Chen, X., M. Sim, P. Sun, J. Zhang. 2008. A linear decision-based approximation approach to stochastic programming. *Oper. Res.* **56**(2) 344–357.
- El-Ghaoui, L., H. Lebret. 1997. Robust solutions to least-square problems to uncertain data matrices. *SIAM J. Matrix Anal. Appl.* **18** 1035–1064.
- El-Ghaoui, L., F. Oustry, H. Lebret. 1998. Robust solutions to uncertain semidefinite programs. *SIAM J. Optim.* **9** 33–52.
- Goldfarb, D., G. Iyengar. 2003. Robust convex quadratically constrained programs. *Math. Programming Ser. B* **97**(3) 495–515.
- Hagstrom, J. N. 1988. Computational complexity of PERT problems. *Networks* **18** 139–147.
- Ruszczynski, A., A. Shapiro, eds. 2003. *Stochastic Programming, Handbook in Operations Research and Management Science*. Elsevier Science, Amsterdam.
- Soyster, A. L. 1973. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* **21**(5) 1154–1157.