Finite adaptability in multistage linear optimization

Dimitris Bertsimas, Constantine Caramanis, *Member, IEEE,*

Abstract—In multistage problems, decisions are implemented sequentially, and thus may depend on past realizations of the uncertainty. Examples of such problems abound in applications of stochastic control and operations research; yet, where robust optimization has made great progress in providing a tractable formulation for a broad class of single-stage optimization problems with uncertainty, multistage problems present significant tractability challenges. In this paper we consider an adaptability model designed with discrete second stage variables in mind. We propose a hierarchy of increasing adaptability that bridges the gap between the static robust formulation, and the fully adaptable formulation. We study the geometry, complexity, formulations, algorithms, examples and computational results for finite adaptability. In contrast to the model of affine adaptability proposed in [2], our proposed framework can accommodate discrete variables. In terms of performance for continuous linear optimization, the two frameworks are complementary, in the sense that we provide examples that the proposed framework provides stronger solutions and vice versa. We prove a positive tractability result in the regime where we expect finite adaptability to perform well, and illustrate this claim with an application to Air Traffic Control.

I. INTRODUCTION

Optimization under uncertainty has long been at the frontier of both theoretical and computational research. Multistage problems, closely related to stochastic control, model decision-making over time, where the uncertainty is revealed sequentially, and future stage decisions may depend on past realizations of uncertainty.[1]

Stochastic optimization (see [10], [22], [24], [25], and references therein) explicitly incorporates a probabilistic description of the uncertainty, often relaxing hard constraints by penalizing infeasibility ([23]), or by using so-called chance constraints ([21]). In the last decade, much work has been done in the single-stage robust optimization framework. Here, the decision-maker makes no probabilistic assumptions, but rather seeks deterministic protection to some bounded level of uncertainty. Recent work has considered the case of linear, semidefinite, and general conic optimization, as well as discrete robust optimization; see, e.g., [3],[4],[8],[9], [18].

[1]Problems from stochastic control differ primarily in the focus on feasibility. While we do not discuss here the applicability of techniques from dynamic programming versus stochastic programming, we refer the reader to [5], [6] for work in dynamic and approximate dynamic programming, and then [13], [22], [25] and references therein for further discussion of this in the Stochastic Optimization formulation of uncertainty).

The focus of this paper is on two-stage optimization models, where the uncertainty follows the robust paradigm, i.e., it is set-based and deterministic:

$$\begin{aligned} \min: \quad & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\omega}) \\ \text{s.t.:} \quad & \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}(\boldsymbol{\omega}) \le \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega. \end{aligned} \tag{I.1}$$

We investigate the class of piecewise constant adaptability functions for $\boldsymbol{y}(\boldsymbol{\omega})$. We are particularly interested in formulations of adaptability that are able to address the case of discrete second stage variables.

Remark 1: While our central motivation is the two-stage optimization model (and extensions to multi-stage problems), it is also interesting to consider the second stage problem as a single stage problem:

$$\begin{aligned} \min: \quad & \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\omega}) \\ \text{s.t.:} \quad & \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}(\boldsymbol{\omega}) \le \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega. \end{aligned} \tag{I.2}$$

In this context, piecewise constant adaptability to the uncertainty, $\boldsymbol{\omega}$, is equivalent to a formulation where the decision-maker receives some advance partial information about the realization of the uncertainty, namely, the uncertainty realization will lie in some given region of a partition of the uncertainty set Ω.

For deterministic uncertainty models, the landscape of solution concepts has two extreme cases. On the one side, we have the static robust formulation where the decision-maker has no adaptability to, or information about, the realization of the uncertainty. On the other extreme is the formulation with complete adaptability, where the decision-maker has arbitrary adaptability to the exact realization of the uncertainty and then selects an optimal solution accordingly.[2] This latter set-up is overly optimistic for several reasons. Exact observations of the uncertainty are rarely possible. Moreover, even if in principle feasible, computing the optimal arbitrarily adaptable second stage function is typically an intractable problem. Furthermore, even implementing such complete adaptability in practice may be too expensive, since effectively it requires complete flexibility in the second stage, and hence in itself may be undesirable[3]. This motivates us to consider the middle ground.

Contributions and Paper Outline

In a departure from the static robust optimization paradigm, we consider a set-up where the decision-maker (perhaps at

[2]In the context of a single-stage problem, this corresponds to having complete knowledge of the exact realization of the uncertainty, as opposed to some coarse model for the advance information. As we comment throughout the paper, while we focus on the two-stage model, the interpretation of the adaptability we introduce, in the one-stage model, is exactly one corresponding to a finite amount of information revealed to the decision-maker.

[3]For an example from circuit design where such second stage limited adaptability constraints are physcially motivated by design considerations, see [28].

some cost) may be able to select some finite number, k, of contingency plans for the second stage solution, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k)$, as opposed to a single robust solution, $\boldsymbol{y}_R$. The central topic of this paper is to understand the structure, properties and value of this finite adaptability.

Our goals in this paper are as follows:

(1) To provide a model of adaptability that addresses the conservativeness of the static robust formulation in the case of a two-stage optimization problem.

(2) To develop a hierarchy of adaptability that bridges the gap between the static robust and completely adaptable formulations, as the level, k, of adaptability increases.

(3) To structure this adaptability specifically to be able to accommodate discrete second-stage variables.

(4) To investigate how to optimally structure the adaptability (i.e., how to choose the contingency plans) for small k. Furthermore, we want to understand the complexity of solving the problem optimally.

(5) In addition to structural properties and theoretical characterizations of the optimal adaptability structure, we would like practical algorithms that perform well in computational examples.

Point by point, we believe the above goals are important for the following reasons. (1) While there exist proposals for adaptability, to the best of our knowledge none are structured specifically to address the fact that the static robust formulation cannot model non-convexity in the uncertainty set, or non-constraintwise uncertainty ([4]). (2) Also, as far as we know, there exist no adaptability proposals that allow a variable degree of adaptability, specifically with the ability to cover the middle ground between the static robust and completely adaptable formulations. (3) While there has been some effort in Stochastic Optimization to address the case of discrete second-stage variables (see, e.g., [19], and references therein) there has been no work addressing the case of integer second-stage variables within the framework of deterministic set-based uncertainty. (4) The completely adaptable formulation is known to be NP-hard to solve in general ([2]) as are other adaptability proposals ([2], [29],[1]), as well as various approaches to Stochastic Programming and chance constraints ([22]). It is important, then, to try to understand how much is possible, and the complexity of achieving it. (4) Given the inherent difficulty of these problems, efficient practical algorithms are of high importance.

In Section II, we provide the basic setup of our adaptability proposal, and we define the problem of selecting k contingency plans. Because of its inherent discrete nature, this proposal can accommodate discrete variables. To the best of our knowledge, this is the first proposal for adaptability that can reasonably deal with discrete variables. In Section III, we give a geometric interpretation of the conservativeness of the static robust formulation. We provide a geometric characterization of when finite adaptability can improve the static robust solution by η, for any (possibly large) chosen $\eta \geq 0$. We obtain necessary conditions that any finite adaptability scheme must satisfy in order to improve the static robust solution by at least η. The full collection of these conditions also constitutes a sufficient

condition for η improvement, when restricted to the second-stage model (I.2).

In Section IV, we consider an exact formulation of the k-adaptability problem as a bilinear optimization problem. For the special case of right hand side uncertainty, we have shown in [11] that the bilinear optimization becomes a discrete optimization problem, and there we provide an integer optimization formulation for the $k = 2$ contingency plan problem. In Section V, we consider the complexity of optimally computing k-adaptability, and we show that structuring the $k = 2$ adaptability optimally, is NP-hard in the minimum of the dimension of the uncertainty, the dimension of the problem, and the number of constraints affected. In particular, we show that if the minimum of these three quantities is small, then optimally structuring 2-adaptability is theoretically tractable.

In Section VI, we consider an example in detail, illustrating several of the subtleties of the geometric characterizations of Section III. Here, we also compare k-adaptability to the affine adaptability proposal of [2]. Following that work, there has been renewed interest in adaptability (e.g., [1],[14],[12],[29]). Our work differs from continuous adaptability proposals in several important ways. First, our model offers a natural hierarchy of increasing adaptability. Second, the intrinsic discrete aspect of the adaptability proposal makes this suitable for any situation where it may not make sense to require information about infinitesimal changes in the data. Indeed, only coarse observations may be available. In addition, especially from a control viewpoint, infinite (and thus infinitesimal) adjustability as required by the affine adaptability framework, may not be feasible, or even desirable. We provide an example where affine adaptability is no better than the static robust solution, while finite adaptability with 3 contingency plans significantly improves the solution.

In Section VII, we provide a heuristic algorithm based on the qualitative prescriptions of Section III. This algorithm is also suitable for solving problems with discrete variables, where if the original discrete static robust problem is computationally tractable, so is our algorithm. Section VIII provides several computational examples, continuous and discrete, illustrating the efficient algorithm of Section VII. We consider a large collection of randomly generated scheduling problems in an effort to obtain some appreciation in the generic case, for the benefit of the first few levels of the adaptability hierarchy. Then, we discuss an application to Air Traffic Control (this application is further considered in [7]). This example serves as an opportunity to discuss when we expect finite adaptability to be appropriate for large scale applications.

II. DEFINITIONS

We consider linear optimization problems with deterministic uncertainty in the coefficients, where the uncertainty set is polyhedral. Uncertainty in the right hand side or in the objective function can be modeled by uncertainty in the matrix (see, e.g., [8]). In Section II-A, we define the static robust formulation, the completely adaptable formulation, and our finite adaptability formulation.

A. *Static Robustness, Complete and Finite Adaptability*

The general two-stage problem we consider, and wish to approximate, is the one with complete adaptability, that can be formulated as:

$$\text{CompAdapt}(\Omega) \triangleq \qquad\qquad\qquad (\text{II.3})$$

$$\left[\begin{array}{ll} \min: & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\omega}) \\ \text{s.t.:} & \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}(\boldsymbol{\omega}) \le \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega \end{array} \right] = (\text{II.4})$$

$$\min_{\boldsymbol{x}} \max_{\boldsymbol{\omega} \in \Omega} \min_{\boldsymbol{y}} \left[\begin{array}{ll} & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\ \text{s.t.:} & \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y} \le \boldsymbol{b} \end{array} \right].$$

Without loss of generality, we assume that only the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ have an explicit dependence on the uncertain parameter, $\boldsymbol{\omega}$ We assume throughout this paper that the parameters of the problem (that is, the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$) depend affinely on the uncertain parameter $\boldsymbol{\omega}$.

On the other end of the spectrum from the completely adaptable formulation, is the static robust formulation, where the second stage variables have no dependence on $\boldsymbol{\omega}$:

$$\text{Static}(\Omega) \triangleq \left[\begin{array}{ll} \min: & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\ \text{s.t.:} & \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y} \le \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega \end{array} \right].$$
$$(\text{II.5})$$

We assume throughout that (II.5) is feasible.

In the k-adaptability problem, the decision-maker chooses k second-stage solutions, $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k\}$, and then commits to one of them only after seeing the realization of the uncertainty. At least one of the k solutions must be feasible regardless of the realization of the uncertainty. We define $\text{Adapt}_k(\Omega)$ as:

$$\left[\begin{array}{ll} \min: & \boldsymbol{c}^\top \boldsymbol{x} + \max\{\boldsymbol{d}^\top \boldsymbol{y}_1, \ldots, \boldsymbol{d}^\top \boldsymbol{y}_k\} \\ & \left\{ \begin{array}{c} \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}_1 \ge \boldsymbol{b} \\ \text{or} \\ \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}_2 \ge \boldsymbol{b} \\ \text{or} \\ \vdots \\ \text{or} \\ \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}_k \ge \boldsymbol{b} \end{array} \right\} \ \forall \boldsymbol{\omega} \in \Omega. \end{array} \right].$$
$$(\text{II.6})$$

This is a disjunctive optimization problem with infinitely many constraints. In Section IV, we formulate this as a (finite) bilinear optimization problem.

If we think of the collection of k second stage vectors, $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_k)$ as contingency plans, where each is implemented depending on the realization of the uncertainty, then the k-adaptability problem becomes a k-partition problem. The decision-maker selects a partition of the uncertainty set Ω into k (possibly non-disjoint) regions: $\Omega = \Omega_1 \cup \cdots \cup \Omega_k$. Thus we can rewrite $\text{Adapt}_k(\Omega)$ as

$$\min_{\Omega = \Omega_1 \cup \cdots \cup \Omega_k} \left[\begin{array}{l} \min: \boldsymbol{c}^\top \boldsymbol{x} + \max\{\boldsymbol{d}^\top \boldsymbol{y}_1, \ldots, \boldsymbol{d}^\top \boldsymbol{y}_k\} \\ \text{s.t.:} \ \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}_1 \ge \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega_1 \\ \qquad\qquad \vdots \\ \qquad \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}_k \ge \boldsymbol{b}, \quad \forall \boldsymbol{\omega} \in \Omega_k \end{array} \right].$$
$$(\text{II.7})$$

The equivalence of formulations (II.6) and (II.7) is immediate.

Throughout this paper we refer equivalently to either k contingency plans, or k-partitions, for the k-adaptability problem.

The inequalities $\text{Static}(\mathcal{P}) \ge \text{Adapt}_k(\mathcal{P}) \ge \text{CompAdapt}(\mathcal{P})$ hold in general.

In the area of multi-stage optimization, there has been significant effort to model the sequential nature of the uncertainty, specifically modeling the fact that some variables may be chosen with (partial) knowledge of the uncertainty. This is often known as recourse ([13],[22]). In [2], the authors consider a multi-stage problem with deterministic uncertainty, where the variables in stage t are affine functions of the uncertainty revealed up to time t. We henceforth refer to this model as *affine adaptability*. The affine adaptability approximation to (II.3) is

$$\text{Affine}(\mathcal{P}) \triangleq \left[\begin{array}{l} \min: \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y}(\boldsymbol{\omega}) \\ \text{s.t.:} \ \boldsymbol{A}(\boldsymbol{\omega})\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{\omega})\boldsymbol{y}(\boldsymbol{\omega}) \ge \boldsymbol{b}, \ \forall \boldsymbol{\omega} \in \Omega \end{array} \right],$$
$$(\text{II.8})$$

where $\boldsymbol{y}(\boldsymbol{\omega})$ is an affine function of the uncertain parameter, $\boldsymbol{\omega}$:

$$\boldsymbol{y}(\boldsymbol{\omega}) = \boldsymbol{Q}\boldsymbol{\omega} + \boldsymbol{q}.$$

The authors show that computing affine adaptability is in general NP-hard, although in some cases it can be well-approximated tractably.

Our finite adaptability proposal is not comparable to affine adaptability: in some cases affine adaptability fails where finite adaptability succeeds, and vice versa.

III. A GEOMETRIC PERSPECTIVE

It is convenient for some of our geometric results to reparameterize the uncertainty set Ω in terms of the actual matrices, $(\boldsymbol{A}(\boldsymbol{\omega}), \boldsymbol{B}(\boldsymbol{\omega}))$, rather than the space of the uncertain parameter, Ω. Then we define:

$$\mathcal{P} \triangleq \{(\boldsymbol{A}, \boldsymbol{B}) = (\boldsymbol{A}(\boldsymbol{\omega}), \boldsymbol{B}(\boldsymbol{\omega})) \ : \ \boldsymbol{\omega} \in \Omega\}.$$

Thus, for example, the static problem now becomes:

$$\text{Static}(\mathcal{P}) \triangleq \left[\begin{array}{ll} \min: & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\ \text{s.t.:} & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} \ge \boldsymbol{b}, \quad \forall (\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P} \end{array} \right].$$
$$(\text{III.9})$$

We assume throughout, that the uncertainty set $\mathcal{P}$ is a polytope, that there are m uncertain constraints, and $\boldsymbol{y} \in \mathbb{R}^n$. We consider both the case where $\mathcal{P}$ is given as a convex hull of its extreme points, and where it is given as the intersection of half-spaces. Some results are more convenient to present in the case of the convex hull representation.

In this section, we provide a geometric view of the gap between the completely adaptable and static robust formulations, and also of the way in which finite adaptability bridges this gap. The key intuition is that the static robust formulation is inherently unable to model non-constraintwise uncertainty and, as is explained below, effectively replaces any given uncertainty set $\mathcal{P}$, with a potentially much larger uncertainty set.

We use this geometric interpretation to obtain necessary conditions that any k-partition must satisfy in order to improve the static robust solution value by at least η, for any chosen value η.

A. The Geometric Gap

Since we consider matrix uncertainty, the elements of $\mathcal{P}$ are two $m \times n$ matrices, $(\boldsymbol{A}, \boldsymbol{B}) = (a_{ij}, b_{ij})$, $1 \leq i \leq m$ and $1 \leq j \leq n$. Given any uncertainty region $\mathcal{P}$, let $\pi_l(\mathcal{P})$ denote the projection of $\mathcal{P}$ onto the components corresponding to the lth constraint of (III.9), i.e., this is the projection onto the lth row of the matrices:

$$\begin{aligned} \pi_l(\mathcal{P}) = \ & \{((a_{l1}, b_{l1}), \ldots, (a_{ln}, b_{ln})) : (\hat{a}_{ij}, \hat{b}_{ij}) \in \mathcal{P}, \\ & (\hat{a}_{lj}, \hat{b}_{lj}) = (a_{lj}, b_{lj}), 1 \leq j \leq n\}. \end{aligned}$$

Then, we define:

$$(\mathcal{P})_R \overset{\triangle}{=} \operatorname{conv}(\pi_1 \mathcal{P}) \times \operatorname{conv}(\pi_2 \mathcal{P}) \times \cdots \times \operatorname{conv}(\pi_m \mathcal{P}). \tag{III.10}$$

The set $(\mathcal{P})_R$ is the smallest hypercube (in the above sense) that contains the set $\mathcal{P}$ (see Figure 1).

Lemma 1: For $\mathcal{P}$ and $(\mathcal{P})_R$ defined as above, we have

(a) $\operatorname{Static}(\mathcal{P}) = \operatorname{Static}((\mathcal{P})_R)$ as well as $\operatorname{Static}(\mathcal{P}) = \operatorname{CompAdapt}((\mathcal{P})_R)$.

(b) For $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ the optimal k-partition of the uncertainty set, we have

$$\operatorname{Adapt}_k(\mathcal{P}) = \operatorname{CompAdapt}((\mathcal{P}_1)_R \cup \cdots \cup (\mathcal{P}_k)_R).$$

(c) There is a sequence of partitions $\{\mathcal{P}_{k1} \cup \mathcal{P}_{k2} \cup \cdots \cup \mathcal{P}_{kk}\}$ so that

$$(\mathcal{P}_{k1})_R \cup \cdots \cup (\mathcal{P}_{kk})_R \longrightarrow \mathcal{P}, \quad k \to \infty.$$

The first part of the lemma says that the static robust formulation cannot model correlation across different constraints, nor can it capture non-convexity in the uncertainty set. Furthermore, it says that this is exactly the reason for the gap between the static robust formulation, and the completely adaptable formulation. The second part of the lemma explains from a geometric perspective why, and how, the adaptive solution improves the static robust cost. The third part gives a geometric interpretation of how finite adaptability bridges the gap between the static robust and completely adaptable formulations.

PROOF. **(a)** To prove this part, we use a simple sandwiching technique that we employ throughout this section. Consider the formulation where not only $\boldsymbol{y}$, but also $\boldsymbol{x}$ may depend on the realization of the uncertainty. This is not implementable, hence we call it the utopic solution, and we define $\operatorname{Utopic}(\mathcal{P})$ as:

$$\begin{bmatrix} \min: & \boldsymbol{c}^\top \boldsymbol{x}(\omega) + \boldsymbol{d}^\top \boldsymbol{y}(\omega) \\ \text{s.t.}: & \boldsymbol{A}(\omega)\boldsymbol{x}(\omega) + \boldsymbol{B}(\omega)\boldsymbol{y}(\omega) \leq \boldsymbol{b}, \quad \forall \omega \in \Omega. \end{bmatrix},$$

and $\operatorname{Utopic}_k(\Omega)$ as:

$$\min_{\Omega = \Omega_1 \cup \cdots \cup \Omega_k} \begin{bmatrix} \min: \max\{\boldsymbol{c}^\top \boldsymbol{x}_1 + \boldsymbol{d}^\top \boldsymbol{y}_1, \ldots, \boldsymbol{c}^\top \boldsymbol{x}_k + \boldsymbol{d}^\top \boldsymbol{y}_k\} \\ \text{s.t.}: \boldsymbol{A}(\omega)\boldsymbol{x}_1 + \boldsymbol{B}(\omega)\boldsymbol{y}_1 \geq \boldsymbol{b}, \quad \forall \omega \in \Omega_1 \\ \vdots \\ \boldsymbol{A}(\omega)\boldsymbol{x}_k + \boldsymbol{B}(\omega)\boldsymbol{y}_k \geq \boldsymbol{b}, \quad \forall \omega \in \Omega_k \end{bmatrix}.$$

We always have: $\operatorname{Utopic}(\mathcal{P}) \leq \operatorname{CompAdapt}(\mathcal{P})$, and $\operatorname{Utopic}_k(\mathcal{P}) \leq \operatorname{Adapt}_k(\mathcal{P})$. By a simple duality argument (see [4]) it follows that we have

$$\operatorname{Static}(\mathcal{P}) = \operatorname{Static}((\mathcal{P})_R) = \operatorname{Utopic}((\mathcal{P})_R).$$

But since $\operatorname{Static}(\mathcal{P}) \leq \operatorname{CompAdapt}((\mathcal{P})_R) \leq \operatorname{Utopic}((\mathcal{P})_R)$, the result follows.

(b) Fix a partition $\mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$. Let $(\boldsymbol{x}^*, \boldsymbol{y}_1^*(\boldsymbol{A}, \boldsymbol{B}), \ldots, \boldsymbol{y}_k^*(\boldsymbol{A}, \boldsymbol{B}))$ be an optimal completely adaptable solution for the uncertainty set $(\mathcal{P}_1)_R \times (\mathcal{P}_2)_R$. Now fix $\boldsymbol{x} = \boldsymbol{x}^*$. Consider the problem of computing an optimal $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$, for this fixed $\boldsymbol{x}$. Since $\boldsymbol{x}$ is fixed, the problem decouples into two problems, with uncertainty sets $(\mathcal{P}_1)_R$ and $(\mathcal{P}_2)_R$, respectively. The optimal completely adaptable solution for this single-stage problem is $\boldsymbol{y}_1^*(\cdot)$, and $\boldsymbol{y}_2^*(\cdot)$. But by part **(a)**, we know that adaptability cannot help. Therefore there exist vectors $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$, that have no dependence on $(\boldsymbol{A}, \boldsymbol{B})$, yet have the same performance. This is what we wanted to show.

(c) It suffices to consider any sequence of partitions where the maximum diameter of any region goes to zero as $k \to \infty$. As the diameter of any region goes to zero, the smallest hypercube (in the sense of (III.10)) also shrinks to a point. $\square$

Example: To illustrate this geometric concept, consider the constraints $\{b_{11}y_1 \leq 1, b_{22}y_2 \leq 1\}$, where the uncertainty set is $\mathcal{P} = \{(b_{11}, b_{22}) : 0 \leq b_1, b_2 \leq 1, b_1 + b_2 \leq 1\}$ (and $b_{12} = b_{21} = 0$). The set $\mathcal{P}$ can be identified with the simplex in $\mathbb{R}^2$. The set $(\mathcal{P})_R$, then, is the unit square. The sets $\mathcal{P}, (\mathcal{P})_R$, and various partitions, are illustrated in Figure 1. ▲

We would like to conclude from Lemma 1 that k-adaptability bridges the gap between the static robust and completely adaptable values, i.e., $\operatorname{Adapt}_k(\mathcal{P}) \to \operatorname{CompAdapt}(\mathcal{P})$ as k increases. With an additional continuity assumption, the proposition below asserts that this is in fact the case.

Continuity Assumption: *For any $\varepsilon > 0$, for any $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}$, there exists $\delta > 0$ and a point $(\boldsymbol{x}, \boldsymbol{y})$, feasible for $(\boldsymbol{A}, \boldsymbol{B})$ and within ε of optimality, such that $\forall (\boldsymbol{A}', \boldsymbol{B}') \in \mathcal{P}$ with $d((\boldsymbol{A}, \boldsymbol{B}), (\boldsymbol{A}', \boldsymbol{B}')) \leq \delta$, $(\boldsymbol{x}, \boldsymbol{y})$ is also feasible for $(\boldsymbol{A}', \boldsymbol{B}')$.*

The Continuity Assumption is relatively mild. It asks that if two matrices are infinitesimally close (here $d(\cdot, \cdot)$ is the usual notion of distance) then there should be a point that is almost optimal for both. Therefore, any problem that has an almost-optimal solution in the strict interior of the feasibility set, satisfies the Continuity Assumption. If the Continuity Assumption does not hold, then note that any optimization model requires exact (completely noiseless) observation of $(\boldsymbol{A}, \boldsymbol{B})$ in order to approach optimality.

Proposition 1: If the Continuity Assumption holds, then for any sequence of partitions of the uncertainty set, $\{\mathcal{P} = ((\mathcal{P}_{k1})_R \cup \cdots \cup (\mathcal{P}_{kk})_R)\}_{k=1}^{\infty}$, with the diameter of the largest set going to zero, the value of the adaptable solution approaches the completely adaptable value. In particular,

$$\lim_{k \to \infty} \operatorname{Adapt}_k(\mathcal{P}) = \operatorname{CompAdapt}(\mathcal{P}).$$

PROOF. Using Lemma 1 parts **(b)** and **(c)**, the proposition says that as long as the Continuity Assumption holds, then

$$[(\mathcal{P}_{k1})_R \cup \cdots \cup (\mathcal{P}_{kk})_R \to \mathcal{P}] \Longrightarrow$$
$$[\operatorname{CompAdapt}((\mathcal{P}_{k1})_R \cup \cdots \cup (\mathcal{P}_{kk})_R) \to \operatorname{CompAdapt}(\mathcal{P})].$$

Indeed, given any $\varepsilon > 0$, for every $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}$, consider the $\delta((\boldsymbol{A}, \boldsymbol{B}))$-neighborhood around $(\boldsymbol{A}, \boldsymbol{B})$ as given by the Continuity Assumption. These neighborhoods form an open cover
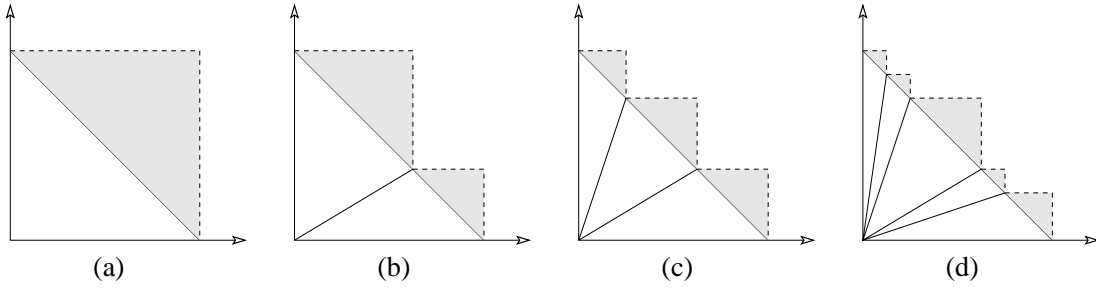
Fig. 1. This figure illustrates the definition in (III.10), and Lemma 1. Let $\mathcal{P} = \{(b_{11}, b_{12}, b_{21}, b_{22}) : 0 \leq b_{11}, b_{22} \leq 1, b_{11} + b_{22} \leq 1, b_{12} = b_{21} = 0\}$. We identify $\mathcal{P}$ with a subset of the plane. The unshaded triangle in Figure (a) illustrates the set $\mathcal{P} = \{(b_{11}, b_{22}) : 0 \leq b_{11}, b_{22} \leq 1, b_{11} + b_{22} \leq 1\}$. The set $(\mathcal{P})_R$ is the entire square, and the shaded part is the difference, $(\mathcal{P})_R \setminus \mathcal{P}$. Figures (b),(c), and (d) show three successively finer partitions, illustrating how $(\mathcal{P}_{k1})_R \cup \cdots \cup (\mathcal{P}_{kk})_R \longrightarrow \mathcal{P}$.

of $\mathcal{P}$. Since $\mathcal{P}$ is compact, we can select a finite subcover. Let the partition $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ be (the closure of) such a subcover. Then, by the Continuity Assumption, $\mathrm{Static}(\mathcal{P}_i) \leq \mathrm{CompAdapt}(\mathcal{P}_i) + \varepsilon$. By definition, $\mathrm{CompAdapt}(\mathcal{P}_i) \leq \max_j \mathrm{CompAdapt}(\mathcal{P}_j) = \mathrm{CompAdapt}(\mathcal{P})$. We have shown that there exists a single sequence of partitions for which the corresponding adaptable solution value approaches the value of complete adaptability. This implies that $\mathrm{Adapt}_k(\mathcal{P}) \to \mathrm{CompAdapt}(\mathcal{P})$. Then recalling that the value of a linear optimization problem is continuous in the parameters, the proof is complete, as any sequence of partitions with diameter going to zero, eventually is a refinement of (a perturbation of) any given finite partition. We give an example in Section VI that shows that the Continuity Assumption cannot be removed. □

B. Necessary Conditions for η-Improvement

In Section III-A, we use duality to show that the static robust problem and the k-adaptability problem are each equivalent to a completely adaptable problem with a larger uncertainty set. This uncertainty set is smaller in the case of the k-adaptability problem, than in the static robust problem. In this section, we characterize how much smaller this effective uncertainty set must be, in order to achieve a given level of improvement from the static robust value. We show that the points of the larger uncertainty set that must be eliminated to obtain a given improvement level, each correspond to necessary conditions that a partition must satisfy in order to guarantee improvement. Furthermore, we show that for the problem where the first-stage decision $\boldsymbol{x}$ is fixed, and we are only considering conditions for finding an improved second-stage solution, i.e., the problem introduced in I.2, collectively these necessary conditions turn out to be sufficient.

Thus in this section we use the geometric characterization of the previous section to essentially characterize the set of partitions that achieve a particular level of improvement over the static robust solution.

Lemma 1 says that $\mathrm{Static}(\mathcal{P}) = \mathrm{CompAdapt}((\mathcal{P})_R)$. Therefore, there must exist some $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}) \in (\mathcal{P})_R$ for which the nominal problem $\min : \{\boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} : \hat{\boldsymbol{A}}\boldsymbol{x} + \hat{\boldsymbol{B}}\boldsymbol{y} \geq \boldsymbol{b}\}$ has value equal to the static robust optimal value of (III.9). Let $\mathcal{M}$ denote all such matrix pairs. In fact, we show that for any $\eta > 0$, there exists a set $\mathcal{M}_\eta \subseteq (\mathcal{P})_R$ such that if

$\boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} < \mathrm{Static}(\mathcal{P}) - \eta$, then $(\boldsymbol{x}, \boldsymbol{y})$ does not satisfy $\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} \geq \boldsymbol{b}$, for any $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{M}_\eta$. We show below that the sets $\mathcal{M}$ and $\mathcal{M}_\eta$ are the images under a computable map, of a polytope associated with the dual of the static robust problem. In Proposition 2 we show that these sets are related to whether a given partition can achieve η-improvement over the static robust value. In Proposition 3 we then show that each point of these sets maps to a necessary condition which any η-improving partition must satisfy.

Proposition 2: **(a)** The sets $\mathcal{M}$ and $\mathcal{M}_\eta$ are the images under a computable map, of a polytope associated with the dual of the static robust problem.
(b) Adaptability with k contingency plans corresponding to the partition $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ improves the cost by more than η only if

$$((\mathcal{P}_1)_R \cup \cdots \cup (\mathcal{P}_k)_R) \cap \bar{\mathcal{M}}_\eta = \emptyset.$$

Here, $\bar{\mathcal{M}}_\eta$ denotes the closure of the set $\mathcal{M}_\eta$.
(c) There is some $k < \infty$ for which k optimally chosen contingency plans can improve the cost by at least η only if $\mathcal{P} \cap \bar{\mathcal{M}}_\eta = \emptyset$.
(d) In the case of the second-stage setup (I.2), the necessary conditions given in parts **(b)** and **(c)** above are also sufficient. For the proof, we first describe a polytope associated to the dual of the robust problem, and we give the map that yields the sets $\mathcal{M}$ and $\mathcal{M}_\eta$, proving **(a)**. Then we prove parts **(b)**, **(c)** and **(d)** of the proposition using the results of Lemma 2 below.

We consider the case where the uncertainty is given as the convex hull of a given set of extreme points: $\mathcal{P} = \mathrm{conv}\{(\boldsymbol{A}^1, \boldsymbol{B}^1), \ldots, (\boldsymbol{A}^K, \boldsymbol{B}^K)\}$. The robust optimization problem has the particularly convenient form,

$$
\begin{aligned}
\min : \quad & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\
\text{s.t.} : \quad & \boldsymbol{A}^i \boldsymbol{x} + \boldsymbol{B}^i \boldsymbol{y} \geq \boldsymbol{b}, \quad 1 \leq i \leq K.
\end{aligned}
\tag{III.11}
$$

For any $\eta > 0$, we consider the infeasible problem

$$
\begin{aligned}
\min : \quad & 0 \\
\text{s.t.} : \quad & \boldsymbol{A}^i \boldsymbol{x} + \boldsymbol{B}^i \boldsymbol{y} \geq \boldsymbol{b}, \quad 1 \leq i \leq K \\
& \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \leq \mathrm{Static}(\mathcal{P}) - \eta.
\end{aligned}
\tag{III.12}
$$

The dual of (III.12) is feasible, and hence unbounded. Let $\mathcal{C}_\eta(\mathcal{P})$ be the closure of the set of directions of dual unbound-

edness of (III.12):

$$\mathcal{C}_\eta = \mathcal{C}_\eta(\mathcal{P}) \triangleq$$
$$\left\{ (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_K) : \begin{array}{l} (\boldsymbol{p}_1 + \cdots + \boldsymbol{p}_K)^\top \boldsymbol{b} \geq \mathrm{Static}(\mathcal{P}) - \eta \\ \boldsymbol{p}_1^\top \boldsymbol{A}^1 + \cdots + \boldsymbol{p}_K^\top \boldsymbol{A}^K = \boldsymbol{c} \\ \boldsymbol{p}_1^\top \boldsymbol{B}^1 + \cdots + \boldsymbol{p}_K^\top \boldsymbol{B}^K = \boldsymbol{d} \\ \boldsymbol{p}_1, \ldots, \boldsymbol{p}_K \geq 0 \end{array} \right\}.$$

Note the dependence on the uncertainty set $\mathcal{P}$. We suppress this when the uncertainty set is clear from the context. $\mathcal{C}_0 = \mathcal{C}_0(\mathcal{P})$ is the set of dual optimal solutions to (III.11). For $(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_K) \in \mathcal{C}_\eta$, let p_{ij} denote the jth component of $\boldsymbol{p}_i$. Let $(\boldsymbol{A}^i)_j$ denote the jth row of the matrix $\boldsymbol{A}^i$, and similarly for $(\boldsymbol{B}^i)_j$. Construct matrices $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ whose j^{th} rows are given by

$$(\tilde{\boldsymbol{A}})_j = \begin{cases} \boldsymbol{0}, & \text{if } \sum_i p_{ij} = 0. \\ \dfrac{p_{1j}(\boldsymbol{A}^1)_j + \cdots + p_{Kj}(\boldsymbol{A}^K)_j}{\sum_i p_{ij}}, & \text{otherwise.} \end{cases}$$

$$(\tilde{\boldsymbol{B}})_j = \begin{cases} \boldsymbol{0}, & \text{if } \sum_i p_{ij} = 0. \\ \dfrac{p_{1j}(\boldsymbol{B}^1)_j + \cdots + p_{Kj}(\boldsymbol{B}^K)_j}{\sum_i p_{ij}}, & \text{otherwise.} \end{cases}$$
(III.13)

Therefore, each nonzero row of $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ is a convex combination of the corresponding rows of the $(\boldsymbol{A}^i, \boldsymbol{B}^i)$ matrices. Let $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$ be any matrix pair in $(\mathcal{P})_R$ that coincides with $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ on all its non-zero rows.

Lemma 2: For $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$ defined as above,

$$\begin{bmatrix} \min : & \boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} \\ \text{s.t.} : & \hat{\boldsymbol{A}}\boldsymbol{x} + \hat{\boldsymbol{B}}\boldsymbol{y} \geq \boldsymbol{b} \end{bmatrix} \geq \mathrm{Static}(\mathcal{P}) - \eta. \quad \text{(III.14)}$$

If $\eta = 0$, and if $(\boldsymbol{x}, \boldsymbol{y})_R$ is an optimal solution for the static robust problem (III.11), then $(\boldsymbol{x}, \boldsymbol{y})_R$ is also an optimal solution for the nominal problem with the matrix pair $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$.
PROOF. The proof follows by duality. We first consider the case $\eta = 0$. The dual to the nominal problem $\min : \{\boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y} : \hat{\boldsymbol{A}}\boldsymbol{x} + \hat{\boldsymbol{B}}\boldsymbol{y} \geq \boldsymbol{b}\}$ is given by $\max : \{\boldsymbol{q}^\top \boldsymbol{b} : \boldsymbol{q}^\top \hat{\boldsymbol{A}} = \boldsymbol{c}, \boldsymbol{q}^\top \hat{\boldsymbol{B}} = \boldsymbol{d}, \boldsymbol{q} \geq \boldsymbol{0}\}$. We construct a solution $\boldsymbol{q}$ to this dual, and show that its objective value is equal to $\boldsymbol{c}^\top \boldsymbol{x}_R + \boldsymbol{d}^\top \boldsymbol{y}_R$, thus implying $\boldsymbol{q}$ is optimal. For $(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_K) \in \mathcal{C}_0$, define the vector $\boldsymbol{q}$ by $q_j \triangleq p_{1j} + p_{2j} + \cdots + p_{Kj}$. The vector $\boldsymbol{q}$ is nonnegative, and in addition, for any $1 \leq r \leq n$, we also have:

$$\begin{aligned} (\boldsymbol{q}^\top \hat{\boldsymbol{A}})_r &= \sum_{j=1}^m q_j \hat{A}_{jr} \\ &= \sum_{j=1}^m \left(\sum_i p_{ij} \right) \left(\frac{1}{\sum_i p_{ij}} \sum_i p_{ij}(\boldsymbol{A}^i)_{jr} \right) \\ &= \sum_{j=1}^m \sum_{i=1}^K p_{ij}(\boldsymbol{A}^i)_{jr} \\ &= \left(\boldsymbol{p}_1 \boldsymbol{A}^1 + \cdots + \boldsymbol{p}_K \boldsymbol{A}^K \right)_r \\ &= c_r. \end{aligned}$$

Similarly, $(\boldsymbol{q}^\top \hat{\boldsymbol{B}})_r = d_r$, and

$$\boldsymbol{q}^\top \boldsymbol{b} = (\boldsymbol{p}_1 + \cdots + \boldsymbol{p}_K)^\top \boldsymbol{b} = \boldsymbol{c}^\top \boldsymbol{x}_R + \boldsymbol{d}^\top \boldsymbol{y}_R.$$

Therefore, $\boldsymbol{q}$ as constructed is an optimal (and feasible) solution to the dual of (III.14), with objective value the same as the dual to the original robust problem (III.11). Since $(\boldsymbol{x}_R, \boldsymbol{y}_R)$ is certainly feasible for problem (III.14), it must then also be optimal. A similar argument holds for $\eta > 0$. $\square$
We can now prove Proposition 2.
PROOF. **(a)** The collection of such $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$ obtained as images of points in $\mathcal{C}_0$ and $\mathcal{C}_\eta$ respectively, under the map given in (III.13) make up the sets $\mathcal{M}$ and $\mathcal{M}_\eta$. Lemma 2 shows that these sets indeed have the required properties.
(b) The value of the k-adaptable solution corresponding to the partition $\mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ is lower-bounded by:

$$\max_{1 \leq d \leq k} \{\mathrm{Static}(\mathcal{P}_d)\}. \quad \text{(III.15)}$$

For the optimal partition choice, this corresponds to $\mathrm{Utopic}_k(\mathcal{P})$, and this may be strictly better than $\mathrm{Adapt}_k(\mathcal{P})$.

By Lemma 1, $\mathrm{Static}(\mathcal{P}_d) = \mathrm{Static}((\mathcal{P}_d)_R)$. If $((\mathcal{P}_1)_R \cup \cdots \cup (\mathcal{P}_k)_R) \cap \bar{\mathcal{M}}_\eta \neq \emptyset$, then we can find some $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}) \in (\mathcal{P}_d)_R \cap \bar{\mathcal{M}}_\eta$, for some $1 \leq d \leq k$, and also we can find matrix pairs $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})_l \in \mathcal{M}_\eta$ with $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})_l \to (\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$. By Lemma 2, the nominal problem with matrices $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})_l$ must have value at least $\mathrm{Static}(\mathcal{P}) - \eta$, for every l. The optimal value of a linear optimization problem is continuous in its parameters. Therefore, the value of the nominal problem with matrices $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$ must also be at least $\mathrm{Static}(\mathcal{P}) - \eta$. The value of $\mathrm{Static}(\mathcal{P}_d)$ can be no more than the value of the nominal problem with matrices $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}})$, and hence $\mathrm{Static}(\mathcal{P}_d) \geq \mathrm{Static}(\mathcal{P}) - \eta$, which means that the improvement cannot be greater than η.

(c) If $\bar{\mathcal{M}}_\eta \cap \mathcal{P} \neq \emptyset$, then the point of intersection will always belong to some element of any partition, and hence no partition can satisfy the condition of part **(b)**.
(d) We can prove the converse for both **(b)** and **(c)** for the case of second stage adaptability. Note that in this case, there is no distinction between what we call the utopic, and the adaptable formulation. Now, for the converse for **(b)**, if the partition does not improve the value by more than η, then there must exist some $1 \leq d \leq k$ such that $\mathrm{Static}(\mathcal{P}_d) \geq \mathrm{Static}(\mathcal{P}) - \eta$. This implies that $\mathcal{C}_\eta(\mathcal{P}_d)$ is non-empty. Any point of $\mathcal{C}_\eta(\mathcal{P}_d)$ then maps via (III.13) to some $\hat{\boldsymbol{B}} \in \mathcal{M}_\eta \cap (\mathcal{P}_d)_R$, and the intersection is non-empty, as required.

For the converse for **(c)**, if the intersection is empty, then since both $\mathcal{P}$ and $\bar{\mathcal{M}}_\eta$ are closed, and $\mathcal{P}$ is compact, the minimum distance

$$\inf_{\boldsymbol{B} \in \mathcal{P}, \hat{\boldsymbol{B}} \in \bar{\mathcal{M}}_\eta} d(\boldsymbol{B}, \hat{\boldsymbol{B}}),$$

is attained, and therefore is strictly positive. Then by Lemma 1 part **(c)**, there must exist some partition of $\mathcal{P}$ that satisfies the empty intersection property of condition **(b)** above. $\square$

We now use the characterization of Proposition 2 to obtain necessary conditions that any η-improving partition must satisfy. To this end, let $\boldsymbol{\alpha}_j$ denote the convex-combination

coefficients used to construct the jth row of $(\tilde{A}, \tilde{B})$ above for all non-zero rows, so that

$$\boldsymbol{\alpha}_j = \frac{1}{\sum_i p_{ij}} \left(p_{1j}, p_{2j}, \ldots, p_{Kj} \right).$$

Using these coefficients, we define matrices $\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_m \in \mathcal{P}$ by

$$\boldsymbol{Q}_j = \sum_{i=1}^{K} (\boldsymbol{\alpha}_j)_i (\boldsymbol{A}, \boldsymbol{B})^i.$$

Consider now any partition of the uncertainty set, $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$. If for some $1 \leq d \leq k$, we have $\{\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_m\} \subset \mathcal{P}_d$, then $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}) \in (\mathcal{P}_d)_R$. Therefore, $(\mathcal{P}_d)_R \cap \mathcal{M}_\eta \neq \emptyset$, and thus by Proposition 2, the proposed partition cannot improve the static robust cost by more than η. Therefore, the set of matrices $\{\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_m\}$ of $\mathcal{P}$ constitutes a *necessary condition* that any η-improving partition of $\mathcal{P}$ must satisfy: a partition of $\mathcal{P}$ can improve the solution more than η only if it splits the set $\{\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_m\}$. Indeed, something more general is true.

Proposition 3: **(a)** Consider any element $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ obtained from a point of $\mathcal{C}_\eta$, according to (III.13). Let us assume that the first r rows of the matrix pair $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$ are nonzero. Let $\mathcal{Q}_i = \pi_i^{-1}((\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})_i)$ denote the set of matrices in $\mathcal{P}$ whose i^{th} row equals the ith row of $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$, $1 \leq i \leq r$. Then a partition $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ can achieve an improvement of more than η only if for any region $\mathcal{P}_d$, $1 \leq d \leq k$, there exists some $1 \leq i \leq r$, such that

$$\mathcal{P}_d \cap \mathcal{Q}_i = \emptyset.$$

(b) Collectively, these necessary conditions are also sufficient, for the second-stage problem (I.2).

PROOF. **(a)** Suppose that there exists a region $\mathcal{P}_d$ of the partition, for which no such index i exists, and we have $\mathcal{P}_d \cap \mathcal{Q}_i \neq \emptyset$ for $1 \leq i \leq r$. Then we can find matrices $\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_r$, such that $\boldsymbol{Q}_i \in \mathcal{P}_d \cap \mathcal{Q}_i$. By definition, the ith row of matrix $\boldsymbol{Q}_i$ coincides with the ith row of $(\tilde{\boldsymbol{A}}, \tilde{\boldsymbol{B}})$. Therefore, $(\hat{\boldsymbol{A}}, \hat{\boldsymbol{B}}) \in (\mathcal{P}_d)_R$. Now the proof of necessity follows from Proposition 2.

(b) Suppose that a partition $\mathcal{P} = \mathcal{P}_1 \cup \cdots \cup \mathcal{P}_k$ satisfies the full list of necessary conditions corresponding to all elements of $\bar{\mathcal{M}}_\eta$, yet the corresponding value of $\mathrm{Adapt}_k(\mathcal{P})$ does not achieve the guaranteed improvement, i.e., $\mathrm{Adapt}_k(\mathcal{P}) = \mathrm{Static}(\mathcal{P}) - \eta' \geq \mathrm{Static}(\mathcal{P}) - \eta$, for some $\eta' \leq \eta$. Then, by the structure of the finite adaptability problem there must be one region of the partition, say $\mathcal{P}_d$, such that $\mathrm{Adapt}_k(\mathcal{P}) = \mathrm{Static}(\mathcal{P}_d)$. Note that this is only always true for the case of single-stage adaptability – in the two-stage case, the equality is only for Utopic_k, which may be strictly less than Adapt_k, and thus this converse need not hold in general. Then $\mathcal{C}_\eta(\mathcal{P}_d)$ is non-empty. Given any point of $\mathcal{C}_\eta(\mathcal{P}_d)$, we can then construct $\hat{\boldsymbol{B}}$ and the corresponding unsatisfied necessary condition $\{\mathcal{Q}_1, \ldots, \mathcal{Q}_r\}$. Expressing the extreme points of $\mathcal{P}_d$ as a convex combination of extreme points of $\mathcal{P}$, this unsatisfied necessary condition corresponds to a point in $\mathcal{C}_\eta(\mathcal{P})$, a contradiction. $\square$

Therefore, we can map any point of $\mathcal{C}_\eta$ to a necessary condition that any partition improving the solution of the static robust problem by at least η, must satisfy. In Section V, we show that computing the optimal partition into two (equivalently, computing the best two contingency plans) is NP-hard. In Section VII, we provide an efficient, but possibly sub-optimal algorithm for the 2^k-partition problem. However, this algorithm does not offer any theoretical guarantee that more progress cannot be made with another choice of partition. Nevertheless, a small list of necessary conditions may provide a short certificate that there does not exist a partition with $k \leq k'$, that achieves η-improvement. In Section VI, we provide a simple example of this phenomenon. In this example, a finite (and small) set of necessary conditions reveals the limits, and structure of 2,3,4,5-adaptability.

IV. EXACT FORMULATIONS

In this section we give an exact and finite formulation of the optimal 2-adaptability problem. We show that the infinite-constraint disjunctive optimization problem (II.6) can be formulated as a bilinear problem.

Thus far we have considered a geometric point of view. Here we follow an algebraic development. In (II.6) we formulated the k-adaptability problem as an infinite-constraint disjunctive program:

$$\begin{aligned}
\min: \quad & \boldsymbol{c}^\top \boldsymbol{x} + \max\{\boldsymbol{d}^\top \boldsymbol{y}_1, \ldots, \boldsymbol{d}^\top \boldsymbol{y}_k\} \\
\text{s.t.}: \quad & [\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_1 \geq \boldsymbol{b} \ \text{ or } \ \cdots \ \text{ or } \ \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_k \geq \boldsymbol{b}] \\
& \forall (\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}.
\end{aligned}$$
(IV.16)

We reformulate this problem as a (finite) bilinear optimization problem. In general, bilinear problems are hard to solve but much work has been done algorithmically (see [15],[26],[27] and references therein) toward their solution. For notational convenience, we consider the case $k = 2$, but the extension to the general case is straightforward. Also, for this section as well, we focus on the case where the uncertainty set $\mathcal{P}$ is given as a convex hull of its extreme points: $\mathcal{P} = \mathrm{conv}\{(\boldsymbol{A}^1, \boldsymbol{B}^1), \ldots, (\boldsymbol{A}^K, \boldsymbol{B}^K)\}$.

Proposition 4: The optimal 2-adaptability value, and the optimal two contingency plans, are given by the solution to the following bilinear optimization:

$$\begin{aligned}
\min: \quad & \boldsymbol{c}^\top \boldsymbol{x} + \max\{\boldsymbol{d}^\top \boldsymbol{y}_1, \boldsymbol{d}^\top \boldsymbol{y}_2\} \\
\text{s.t.}: \quad & \mu_{ij} \left[(\boldsymbol{A}^l \boldsymbol{x} + \boldsymbol{B}^l \boldsymbol{y}_1)_i - b_i \right] + \\
& (1 - \mu_{ij}) \left[(\boldsymbol{A}^l \boldsymbol{x} + \boldsymbol{B}^l \boldsymbol{y}_2)_j - b_j \right] \geq 0, \\
& \forall 1 \leq i, j \leq m, \quad \forall 1 \leq l \leq K \\
& 0 \leq \mu_{ij} \leq 1, \quad \forall 1 \leq i, j \leq m.
\end{aligned}$$
(IV.17)

Recall that m is the number of rows of $\boldsymbol{A}$ and $\boldsymbol{B}$. We can interpret the variables μ_{ij} essentially as a mixing of the constraints. For any $\{\mu_{ij}\}$, the triple $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2) = (\boldsymbol{x}_R, \boldsymbol{y}_R, \boldsymbol{y}_R)$ is feasible. Indeed, fixing $\mu_{ij} = 1$ for all (i, j) leaves $\boldsymbol{y}_2$ unrestricted, and the resulting constraints on $\boldsymbol{y}_1$ recover the original static robust problem. Thus, the problem is to find the optimal mixing weights.

PROOF. We show that a triple $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2)$ is a feasible solution to problem (IV.16) if and only if there exist weights

$\mu_{ij} \in [0, 1]$, $1 \le i, j \le m$, such that

$$\mu_{ij}\left[(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i - b_i\right] +$$
$$(1-\mu_{ij})\left[(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j - b_j\right] \ge 0,$$
$$\forall\, 1 \le i, j \le m, \quad \forall\, 1 \le l \le K.$$

Suppose that the triple $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2)$ is not a feasible solution to problem (IV.16). Then there exists $1 \le i, j \le m$ and matrix pair $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}$ such that $(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_1)_i - b_i < 0$, and $(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_2)_j - b_j < 0$. Since $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}$, we must have $(\boldsymbol{A}, \boldsymbol{B}) = \sum_{l=1}^{K} \lambda_l(\boldsymbol{A}^l, \boldsymbol{B}^l)$, for a convex combination given by $\boldsymbol{\lambda}$. For any $\mu_{ij} \in [0, 1]$ we have:

$$\mu_{ij}\left[\sum_l \lambda_l(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i - b_i\right] +$$
$$(1-\mu_{ij})\left[\sum_l \lambda_l(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j - b_j\right] < 0$$
$$\Rightarrow \quad \sum_l \lambda_l\Big[\mu_{ij}\left\{(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i - b_i\right\} +$$
$$(1-\mu_{ij})\left\{(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j - b_j\right\}\Big] < 0.$$

This follows since $\sum_l \lambda_l = 1$. But then there must be some index l^* for which the corresponding term in the sum is negative, i.e.,

$$\mu_{ij}\left\{(\boldsymbol{A}^{l^*}\boldsymbol{x} + \boldsymbol{B}^{l^*}\boldsymbol{y}_1)_i - b_i\right\} +$$
$$(1-\mu_{ij})\left\{(\boldsymbol{A}^{l^*}\boldsymbol{x} + \boldsymbol{B}^{l^*}\boldsymbol{y}_2)_j - b_j\right\} < 0.$$

For the converse, let $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2)$ be a feasible solution to problem (IV.16). We show there exist weights $\mu_{ij} \in [0, 1]$ satisfying the required inequalities. By assumption, for any $(\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}$, either $[\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_1 \ge \boldsymbol{b}]$, or $[\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_2 \ge \boldsymbol{b}]$. In particular, for any $1 \le i, j \le m$, the value of the following optimization over $\mathcal{P}$ is finite and non-positive (recall that $\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2$ are fixed).

$$\begin{aligned}
\max: \quad & \varepsilon \\
\text{s.t.:} \quad & (\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_1)_i + \varepsilon \le b_i \\
& (\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}_2)_j + \varepsilon \le b_j \\
& (\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}.
\end{aligned}$$

Writing $\mathcal{P} = \{\sum_l \lambda_l(\boldsymbol{A}^l, \boldsymbol{B}^l) : \sum_l \lambda_l = 1, \lambda_l \ge 0\}$, and taking the dual using dual variables μ, ν for the two inequality constraints, and w for the normalization constraint in $\boldsymbol{\lambda}$, we have:

$$\begin{aligned}
\min: \quad & \mu b_i + \nu b_j - w \\
\text{s.t.:} \quad & \mu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i + \nu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j - w \ge 0, \\
& \qquad \forall\, 1 \le l \le K \\
& \mu + \nu = 1 \\
& \mu, \nu \ge 0.
\end{aligned}$$

By strong duality, this problem is feasible, and its optimal value is non-positive. In particular, the following system is feasible:

$$\left\{
\begin{aligned}
& \mu b_i + \nu b_j \le w \\
& \mu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i + \nu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j \ge w \\
& \mu + \nu = 1 \\
& \mu, \nu \ge 0
\end{aligned}
\right\}$$

and therefore there exists $\mu \in [0, 1]$, and w such that

$$\mu b_i + (1-\mu)b_j \le w \le \mu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i + \nu(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j,$$
$$\forall\, 1 \le l \le K.$$

Grouping the terms on one side of the inequality, we have that there exists a weight $\mu \in [0, 1]$ such that

$$\mu\left[(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_1)_i - b_i\right] + (1-\mu)\left[(\boldsymbol{A}^l\boldsymbol{x} + \boldsymbol{B}^l\boldsymbol{y}_2)_j - b_j\right] \ge 0,$$
$$\forall\, 1 \le l \le K.$$

$\square$

V. COMPLEXITY

In this section, we consider the complexity of k-adaptability. We show that even in the restricted case of right hand side uncertainty, in fact even in the special case where $\mathcal{P}$ has the form of a generalized simplex, computing the optimal partition of $\mathcal{P}$ into two sets, $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$, is NP-hard. We then go on to show that, despite this negative complexity result, there are cases of interest where the tractability of finding the optimal hyperplane partition is tractable. In particular, as we claim via some computational examples in Section VIII, finite adaptability is particularly well-suited for when the dimension of the uncertainty set is small; this is included in the classes of problems for which computing the optimal finite adaptability is tractable.

We show that if any of the three quantities: dimension of the uncertainty, dimension of the problem, number of uncertain constraints, is fixed, then computing the optimal 2-adaptability is theoretically tractable.

Proposition 5: Obtaining the optimal split $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$ is in general NP-hard.

In particular, computing 2-adaptability is NP-hard. We obtain our hardness result using a reduction from PARTITION, which is NP-complete ([17],[20]). We show that if we can find the optimal split of an uncertainty set, then we can solve any PARTITION problem.

PROOF. The data for the PARTITION problem are the positive numbers $v_1, \ldots, v_m$. The problem is to minimize $|\sum_{i \in S} v_i - \sum_{j \in S^c} v_j|$ over subsets S. Given any such collection of numbers, consider the polytope $\mathcal{P} = \text{conv}\{\boldsymbol{e}_1 v_1, \ldots, \boldsymbol{e}_m v_m\}$, where the $\boldsymbol{e}_i$ form the standard basis for $\mathbb{R}^m$. Thus, $\mathcal{P}$ is the simplex in $\mathbb{R}^m$, but with general intercepts v_i. Consider the static robust optimization problem:

$$\begin{aligned}
\min: \quad & x \\
\text{s.t.:} \quad & x \ge \sum_i y_i \\
& I\boldsymbol{y} \ge \boldsymbol{b}, \quad \forall \boldsymbol{b} \in \mathcal{P}.
\end{aligned} \tag{V.18}$$

Suppose the optimal partition is $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$. Then, letting $\boldsymbol{e} = (1, 1, \ldots, 1)$, the 2-adaptable problem can be written as:

$$\begin{aligned}
\min: \quad & x \\
\text{s.t.:} \quad & x \ge \boldsymbol{e}^\top \boldsymbol{y}_1 \\
& x \ge \boldsymbol{e}^\top \boldsymbol{y}_2 \\
& I\boldsymbol{y}_1 \ge \boldsymbol{b}_1 \\
& I\boldsymbol{y}_2 \ge \boldsymbol{b}_2,
\end{aligned}$$

where $\boldsymbol{b}_1, \boldsymbol{b}_2$ are the component-wise minimum over $\mathcal{P}_1$ and $\mathcal{P}_2$, respectively. Since $\mathcal{P}$ has the particularly easy form

given above, it is straightforward to see that, without loss of generality, we can write $\boldsymbol{b}_1$, and $\boldsymbol{b}_2$ as

$$\begin{aligned}\boldsymbol{b}_1 &= (v_1, v_2, \ldots, v_k, \lambda_{k+1}, \ldots, \lambda_m)\\ \boldsymbol{b}_2 &= (\mu_1, \mu_2, \ldots, \mu_k, v_{k+1}, \ldots, v_m),\end{aligned}$$

where $0 \le \lambda_j \le v_j$ for $k+1 \le j \le m$. In this case, we must have (see [11] for full details)

$$\begin{aligned}\mu_1 &\ge \max\left\{v_1 - \left(\frac{v_1}{v_{k+1}}\right)\lambda_{k+1}, v_1 - \left(\frac{v_1}{v_{k+2}}\right)\lambda_{k+2}, \ldots,\right.\\ &\qquad\qquad \left.\ldots, v_1 - \left(\frac{v_1}{v_m}\right)\lambda_m\right\}\\ &\vdots\\ \mu_k &\ge \max\left\{v_k - \left(\frac{v_k}{v_{k+1}}\right)\lambda_{k+1}, v_k - \left(\frac{v_k}{v_{k+2}}\right)\lambda_{k+2}, \ldots,\right.\\ &\qquad\qquad \left.\ldots, v_k - \left(\frac{v_k}{v_m}\right)\lambda_m\right\}.\end{aligned}$$

Since we claim that the pair $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ corresponds to the optimal partition of $\mathcal{P}$, we can take the inequalities above to be satisfied by equality, i.e., we take the μ_i to be as small as possible. Therefore, once the $\{\lambda_j\}$ are fixed, so are the $\{\mu_i\}$, and the pair $(\boldsymbol{b}_1, \boldsymbol{b}_2)$ is determined.

Now we compute the value of the free parameters $(\lambda_{k+1}, \ldots, \lambda_m)$ that determine the pair $(\boldsymbol{b}_1, \boldsymbol{b}_2)$. For the specific form of the optimization problem we consider, given a split $\mathcal{P}_1 \cup \mathcal{P}_2$ where $\mathcal{P}_1$ is covered by $\boldsymbol{b}_1$ and $\mathcal{P}_2$ by $\boldsymbol{b}_2$, the optimization takes the simple form:

$$\left\{\begin{array}{ll}\min: & \max\left\{\left(\sum_i y_i^{(1)}\right), \left(\sum_i y_i^{(2)}\right)\right\}\\ \text{s.t.}: & \boldsymbol{y}^{(1)} \ge \boldsymbol{b}_1\\ & \boldsymbol{y}^{(2)} \ge \boldsymbol{b}_2\end{array}\right\} =$$

$$\max\left\{\left(\sum_i (\boldsymbol{b}_1)_i\right), \left(\sum_i (\boldsymbol{b}_2)_i\right)\right\}.$$

Therefore, if the partition is optimal, we must have $(\sum_i (\boldsymbol{b}_1)_i) = (\sum_i (\boldsymbol{b}_2)_i)$. Thus, we have

$$v_1 + \cdots + v_k + \lambda_{k+1} + \cdots + \lambda_m = \mu_1 + \cdots + \mu_k + v_{k+1} + \cdots + v_m. \tag{V.19}$$

We have $(m-k)$ parameters that are not specified. The maximizations above that determine the μ_j give $(m-k-1)$ equations. Then Eq. (V.19) gives the final equation to determine our parameters uniquely. From the maximizations defining $\{\mu_j\}$, we have

$$v_j - \left(\frac{v_j}{v_{k+i}}\right)\lambda_{k+i} = v_j - \left(\frac{v_j}{v_{k+i'}}\right)\lambda_{k+i'},$$
$$1 \le j \le k, \quad 1 \le i, i' \le m-k.$$

Solving in terms of λ_m, the above equations yield $\lambda_{k+i} = \left(\frac{v_{k+i}}{v_m}\right)\lambda_m$, $1 \le i \le m-k-1$. Substituting this back into Eq. (V.19), we obtain an equation in the single variable λ_m:

$$v_1 + \cdots + v_k + \lambda_m \left(\frac{v_{k+1} + \cdots + v_m}{v_m}\right) =$$
$$\left(v_1 - v_1 \frac{\lambda_m}{v_m}\right) + \cdots + \left(v_k - v_k \frac{\lambda_m}{v_m}\right) + (v_{k+1} + \cdots + v_m),$$

which gives:

$$\lambda_m \left(\frac{v_1 + \cdots + v_m}{v_m}\right) = (v_{k+1} + \cdots + v_m) \Longrightarrow$$
$$\lambda_{k+i} = \frac{v_i(v_{k+1} + \cdots + v_m)}{v_1 + \cdots + v_m}, \quad 1 \le i \le m-k.$$

Using these values of $\{\lambda_{k+i}\}$, we find that the optimal value of the optimization is given by

$$\begin{aligned}&\sum_{i=1}^k v_i + \sum_{j=k+1}^m \lambda_j\\ &= \frac{\left(\sum_{i=1}^k v_i\right)\left(\sum_{j=1}^m v_j\right) + \left(\sum_{j=k+1}^m v_j\right)\left(\sum_{j=k+1}^m v_j\right)}{\sum_{i=1}^m v_i}\\ &= \frac{\left(\left(\sum_{i=1}^k v_i\right) + \left(\sum_{j=k+1}^m v_j\right)\right)^2 - \left(\sum_{i=1}^k v_i\right)\left(\sum_{j=k+1}^m v_j\right)}{\sum_{i=1}^m v_i}.\end{aligned}$$

The first term in the numerator, and also the denominator, are invariant under choice of partition. Thus, if this is indeed the optimal solution to the optimization (V.18), as we assume, then the second term in the numerator must be maximized. Thus, we see that minimizing (V.18) is equivalent to maximizing the product $\left(\sum_{i \in S} v_i\right)\left(\sum_{j \in S^c} v_j\right)$ over $S \subseteq \{1, \ldots, m\}$. This is equivalent to the PARTITION problem. $\square$

Note that in this example, the dimension of the uncertainty, the dimension of the problem, and the number of constraints affected by the uncertainty are all equal. Next we show that if any one of these three quantities is fixed, then computing the optimal 2-adaptability is theoretically tractable.

Proposition 6: We consider the static robust problem

$$\begin{aligned}\min: \quad &\boldsymbol{c}^\top \boldsymbol{x} + \boldsymbol{d}^\top \boldsymbol{y}\\ \text{s.t.}: \quad &\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y} \ge \boldsymbol{b}, \quad \forall (\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P}\\ &\boldsymbol{F}\boldsymbol{x} + \boldsymbol{G}\boldsymbol{y} \ge \boldsymbol{f}.\end{aligned}$$

Let $\mathcal{P} = \text{conv}\{(\boldsymbol{A}_1, \boldsymbol{B})_1, \ldots, (\boldsymbol{A}\boldsymbol{B})_N\}$ be an uncertainty set that allows for efficient solution of the robustified linear optimization problem (note that N need not necessarily be small). Let $d = \min\{N, \dim(\mathcal{P})\}$ be the real dimension of $\mathcal{P}$, let n denote the number of optimization variables, and let m be the number of rows of $(\boldsymbol{A}, \boldsymbol{B})$, i.e., the number of uncertain constraints. Define $\kappa = \min\{d, n, m\}$. Then, we can compute the ε-optimal 2-adaptability generated by a hyperplane partition, in time $O\left(\text{poly}(d, n, m, 1/\varepsilon)e^\kappa\right)$. In particular, if κ is constant, the hyperplane generated 2-adaptability can be computed efficiently.

PROOF. There are three possibilities: κ is defined by d, n, or m. In the case where d or n are fixed, then the result follows immediately, since we can find the best partition, or the best two solutions $\{\boldsymbol{y}_1, \boldsymbol{y}_2\}$ by brute force discretization of the uncertainty set, or the feasible set, respectively. The only interesting case is when d and n are possibly large, but m is

a constant. In this case, Proposition 4 says:

$$
\begin{aligned}
\min : \quad & \boldsymbol{c}^\top \boldsymbol{x} + \max\{\boldsymbol{d}^\top \boldsymbol{y}_1, \boldsymbol{d}^\top \boldsymbol{y}_2\} \\
\text{s.t.} : \quad & \mu_{ij}\left[(\boldsymbol{Ax} + \boldsymbol{By}_1)_i - b_i\right] + \\
& (1 - \mu_{ij})\left[(\boldsymbol{Ax} + \boldsymbol{By}_2)_j - b_j\right] \geq 0, \\
& \forall\, 1 \leq i, j \leq m, \quad \forall (\boldsymbol{A}, \boldsymbol{B}) \in \mathcal{P} \\
& 0 \leq \mu_{ij} \leq 1, \quad \forall\, 1 \leq i, j \leq m.
\end{aligned}
$$

For any fixed values of $\{\mu_{ij}\}$, the resulting problem is a static robust problem with uncertainty set $\mathcal{P}$, and hence by our assumption, it can be solved efficiently. Now if m is small, we discretize the possible set of $\{\mu_{ij}\}$, and search over this set by brute force. This completes the proof. □

While in principle this result says that for κ small the problem is tractable, in large scale applications we require more than theoretical tractability. We describe one such example in Section VIII-C. In Section VII, we seek to give tractable algorithms that will be practically implementable in applications.

VI. AN EXTENDED EXAMPLE

In this section we consider a detailed example. Through this example, we aim to illustrate several points and aspects of the theory developed in Section III above:

1) Propositions 2 and 3 tell us how to map $\mathcal{C}_\eta$ to $\mathcal{M}_\eta$ and then to obtain necessary conditions for η-improvement. Here we illustrate this process.
2) A small set of necessary conditions (obtained as in Propositions 2 and 3) may reveal the limits of k-adaptability for some k.
3) While in general not sufficient to guarantee η-improvement, a small set of necessary conditions may even suffice to reveal the optimal structure of k-adaptability for some k.
4) Finite adaptability may improve the solution considerably, even when affine adaptability fails, i.e., even when affine adaptability is no better than the static robust solution.
5) The Continuity Assumption may not be removed from Proposition 1. Without it, (uncountably) infinite adaptability may be required for even arbitrarily small improvement from the static robust solution.
6) The closure of the sets $\mathcal{M}$ and $\mathcal{M}_\eta$ in Proposition 2 cannot be relaxed.

We consider an example with one-dimensional uncertainty set. Note that here there is no uncertainty in the $\boldsymbol{A}$-matrix.

$$
\begin{aligned}
\min : \quad & x \\
\text{s.t.} : \quad & x \geq y_1 + y_2 + y_3 \\
& \boldsymbol{By} \geq \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \\
& \forall\, \boldsymbol{B} \in \operatorname{conv}\{\boldsymbol{B}^1, \boldsymbol{B}^2\} = \\
& \operatorname{conv}\left\{ \begin{pmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ \frac{1}{2} & \frac{1}{5} & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 1 \\ \frac{1}{5} & \frac{1}{2} & 0 \end{pmatrix} \right\} \\
& y_1, y_2, y_3 \geq 0.
\end{aligned}
\tag{VI.20}
$$

The unique optimal solution is $x_R = 27/7$, $\boldsymbol{y}_R = (10/7, 10/7, 1)$, so the corresponding value is $\operatorname{Static}(\mathcal{P}) = 27/7$. The completely adaptable value is $\operatorname{CompAdapt}(\mathcal{P}) = 3$. For notational convenience, can rewrite this problem more compactly by minimizing $y_1 + y_2 + y_3$ directly. Then the optimal value for x will be that minimum value.

Then, the dual to the robust problem (VI.20) is

$$
\begin{aligned}
\max : \quad & (\boldsymbol{p} + \boldsymbol{q})^\top \boldsymbol{b} \\
\text{s.t.} : \quad & \boldsymbol{p}^\top \boldsymbol{B}^1 + \boldsymbol{q}^\top \boldsymbol{B}^2 \leq \boldsymbol{d} \\
& \boldsymbol{p}, \boldsymbol{q} \geq \boldsymbol{0}.
\end{aligned}
$$

There are two extreme dual optimal solutions: $\boldsymbol{p}, \boldsymbol{q} = (0, 0, 10/7), (0, 1, 10/7)$, and $\boldsymbol{p}, \boldsymbol{q} = (1, 0, 10/7), (0, 0, 10/7)$. We illustrate point (1) above by mapping these two points to the corresponding necessary conditions. Each of these maps to a unique matrix $\tilde{\boldsymbol{B}}$. Recall that, considering the ith component of $\boldsymbol{p}$, and the ith component of $\boldsymbol{q}$, we obtain the ith row of the matrix:

$$
(p_i, q_i) \longmapsto (\tilde{\boldsymbol{B}})_i \triangleq \frac{1}{p_i + q_i}(p_i \cdot (\boldsymbol{B}^1)_i + q_i \cdot (\boldsymbol{B}^2)_i),
$$

for all i such that $p_i + q_i \neq 0$. For the first extreme dual optimal solution, this condition is met for $i = 2, 3$, and thus we have:

$$
\begin{aligned}
(p_2, q_2) \quad \mapsto \quad & \frac{1}{0 + 1}(0 \cdot (1, 1, 1) + 1 \cdot (0, 0, 1)) = (0, 0, 1) \\
(p_3, q_3) \quad \mapsto \quad & \frac{1}{10/7 + 10/7}\left[(10/7) \cdot (1/2, 1/5, 0) \right. \\
& \left. \qquad\qquad + (10/7) \cdot (1/5, 1/2, 0)\right] \\
& = \frac{1}{2}(1/2, 1/5, 0) + \frac{1}{2}(1/5, 1/2, 0) \\
& = (7/20, 7/20, 0).
\end{aligned}
$$

For the second extreme dual optimal solution, the nonzero rows are $i = 1, 3$, and we get:

$$
\begin{aligned}
(p_1, q_1) \quad \mapsto \quad & \frac{1}{1 + 0}(1 \cdot (0, 0, 1) + 0 \cdot (1, 1, 1)) = (0, 0, 1) \\
(p_3, q_3) \quad \mapsto \quad & \frac{1}{20/7}\left[(10/7) \cdot (1/2, 1/5, 0) \right. \\
& \left. \qquad\qquad + (10/7) \cdot (1/5, 1/2, 0)\right] \\
= \quad & (7/20, 7/20, 0).
\end{aligned}
$$

Next, according to Proposition 3, we consider the set of matrices in $\mathcal{P}$ that share one of the nonzero rows of $\tilde{\boldsymbol{B}}_1$, and similarly for $\tilde{\boldsymbol{B}}_2$. These are specified by the convex combination coefficients that form the non-zero rows. The two convex combinations for the first dual solution are formed by the coefficients $\boldsymbol{\alpha}_2 = (0, 1)$ and $\boldsymbol{\alpha}_3 = (1/2, 1/2)$. The second dual solution has convex combination coefficients $\boldsymbol{\alpha}_1 = (1, 0)$ and $\boldsymbol{\alpha}_3 = (1/2, 1/2)$. Therefore, any strictly improving partition must be such that no single region contains both matrices $\{\boldsymbol{B}^2, \frac{1}{2}\boldsymbol{B}^1 + \frac{1}{2}\boldsymbol{B}^2\}$, nor the two matrices $\{\boldsymbol{B}^1, \frac{1}{2}\boldsymbol{B}^1 + \frac{1}{2}\boldsymbol{B}^2\}$. Evidently, no such partition into 2 (convex) regions exists. Therefore 2-adaptability cannot satisfy these two necessary conditions, and thus (in this example) is no better than the static robust solution of (VI.20). This illustrates point (2) above: the necessary conditions corresponding to the two extreme points of $\mathcal{C}_0$ are alone sufficient to prove that 2-adaptability is no better than the static robust solution.

Next we consider the more general case $\mathcal{C}_\eta$ and $\mathcal{M}_\eta$. We consider a few different values of η: $\eta_1 = (27/7) - 3.8$, $\eta_2 = (27/7) - 3.2$, and $\eta_3 = (27/7) - 2.9$. We generate the extreme points $(\boldsymbol{p}, \boldsymbol{q})$ of $\mathcal{C}_{\eta_1}$, and the points of $\mathcal{M}_{\eta_1}$ to which they map. The polytope $\mathcal{C}_{\eta_1}$ has 12 extreme points[4]. These yield four non-redundant necessary conditions:

$$\mathcal{N}_1 = \{\boldsymbol{B}^2, \tfrac{10}{21}\boldsymbol{B}^1 + \tfrac{11}{21}\boldsymbol{B}^2\};$$
$$\mathcal{N}_2 = \{\boldsymbol{B}^1, \tfrac{11}{21}\boldsymbol{B}^1 + \tfrac{10}{21}\boldsymbol{B}^2\};$$
$$\mathcal{N}_3 = \{\tfrac{1}{50}\boldsymbol{B}^1 + \tfrac{49}{50}\boldsymbol{B}^2, \tfrac{1}{2}\boldsymbol{B}^1 + \tfrac{1}{2}\boldsymbol{B}^2\};$$
$$\mathcal{N}_4 = \{\tfrac{49}{50}\boldsymbol{B}^1 + \tfrac{1}{50}\boldsymbol{B}^2, \tfrac{1}{2}\boldsymbol{B}^1 + \tfrac{1}{2}\boldsymbol{B}^2\}.$$

While there exists no partition into only two regions that can simultaneously satisfy these four necessary conditions, the three-region split $[0, 1] = [0, 1/6] \cup [1/6, 5/6] \cup [5/6, 1]$ does satisfy $\mathcal{N}_1 - \mathcal{N}_4$; we can check that none of the sets $\mathcal{N}_i$, $1 \leq i \leq 4$, are contained within any single region of the proposed partition. In fact, this partition decreases the cost by $0.4672 \geq \eta_1$. The polytope $\mathcal{C}_{\eta_2}$ has 12 vertices. The non-redundant constraints generated by points of $\mathcal{M}_{\eta_2}$ corresponding to the extreme points of $\mathcal{C}_{\eta_2}$, are

$$\mathcal{N}_1 = \{\boldsymbol{B}^1, \tfrac{28}{33}\boldsymbol{B}^1 + \tfrac{5}{33}\boldsymbol{B}^2\};$$
$$\mathcal{N}_2 = \{\boldsymbol{B}^2, \tfrac{28}{33}\boldsymbol{B}^2 + \tfrac{5}{33}\boldsymbol{B}^1\}$$
$$\mathcal{N}_3 = \{\tfrac{77}{100}\boldsymbol{B}^1 + \tfrac{23}{100}\boldsymbol{B}^2, \tfrac{1}{2}\boldsymbol{B}^1 + \tfrac{1}{2}\boldsymbol{B}^2\};$$
$$\mathcal{N}_4 = \{\tfrac{23}{100}\boldsymbol{B}^1 + \tfrac{77}{100}\boldsymbol{B}^2, \tfrac{1}{2}\boldsymbol{B}^1 + \tfrac{1}{2}\boldsymbol{B}^2\}.$$

It is easy to check that these four necessary conditions are not simultaneously satisfiable by any partition with only three (convex) regions. Indeed, at least 5 are required. This is another illustration of point (2) from above: a small set of four necessary conditions suffices to prove that 3-adaptability cannot improve the static robust solution by more than $\eta_2 = (27/7) - 3.2$.

The smallest η at which the necessary conditions corresponding to the extreme points of $\mathcal{C}_\eta$ provide a certificate that at least 5 regions are required for any partition to achieve an η-improvement or greater, is $\hat{\eta} \approx 27/7 - 3.2770 \approx 0.5801$. This illustrates point (3) above: examining values of $\eta \in [0, \hat{\eta}]$, the four necessary conditions implied by the extreme points of $\mathcal{C}_\eta$ are sufficient to reveal that two-adaptability is no better than the static robust solution, and in addition, they reveal the limit of 3-adaptability. Furthermore, they reveal the optimal 3-partition to be: $[0, 1] = [0, \hat{\lambda}] \cup [\hat{\lambda}, 1 - \hat{\lambda}] \cup [1 - \hat{\lambda}, 1]$, for $\hat{\lambda} \approx 0.797$.

Finally, let us consider $\eta_3 = (27/7) - 2.9$. In this case, we are asking for more improvement than even the completely adaptable formulation could provide (recall $\mathrm{CompAdapt}(\mathcal{P}) = 3$). In short, such improvement is not possible within our framework of a deterministic adversary. Proposition 2 tells us how the polytope $\mathcal{C}_{\eta_3}$ and the set $\mathcal{M}_{\eta_3}$ witness this impossibility. The polytope $\mathcal{C}_{\eta_3}$ has 31 vertices. It is enough to consider one of these vertices in particular: $\boldsymbol{v} = (9/10, 1/10, 9/5), (0, 0, 0)$. The corresponding necessary condition is: $\mathcal{N} = \{\boldsymbol{B}^1, \boldsymbol{B}^1, \boldsymbol{B}^1\}$. Evidently, no number of partitions can ever satisfy this necessary condition. Indeed,

[4]These computations were done using the software CDD by Komei Fukuda [16]. This is an implementation of the double description method. See also http://www.cs.mcgill.ca/ fukuda/soft/cddman/node2.html for further details.

this is precisely what Proposition 2 says: if progress η is not possible, it must be because $\bar{\mathcal{M}}_\eta \cap \mathcal{P} \neq \emptyset$.

Next we illustrate point (4), by showing that for the problem (VI.20) above, the affine adaptability proposal of Ben-Tal et al. ([2]) is no better than the static robust formulation, even though 3-adaptability significantly improves the static robust solution, and thus outperforms affine adaptability. In Figure 2 on the left, we have the actual optimal solutions for the com-
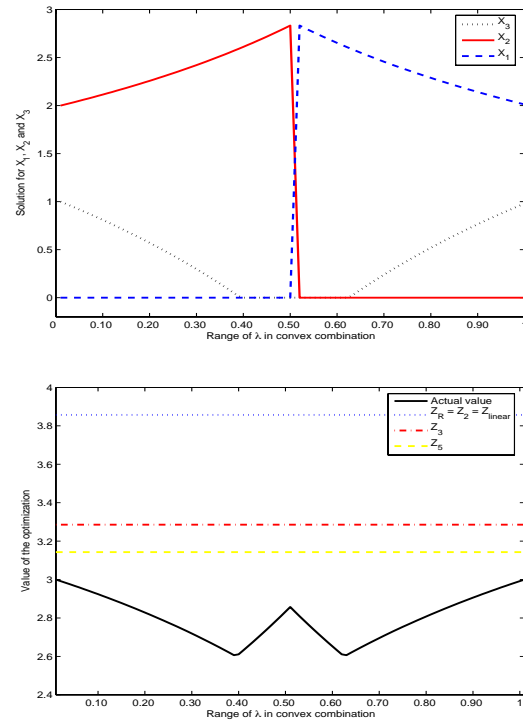

Fig. 2. The figure on the left illustrates the optimal response policy for the decision-maker. The optimal response function is far from linear. In the figure to the right the lowest curve is the value of the nominal LP as a function of the realization of the uncertainty. The next three lines, Z_5, Z_3, Z_R, illustrate the value of 5,3-adaptability, and the static robust value, respectively. The static robust value coincides with the value of affine adaptability.

pletely adaptable problem. For every $\lambda \in [0, 1]$, the decision-maker has an optimal response, $\boldsymbol{y}(\lambda) = (y_1(\lambda), y_2(\lambda), y_3(\lambda))$. The figure on the right illustrates the optimal completely adaptable cost as a function of λ, as well as the optimal static robust cost (the line at the top) and then the cost when the decision-maker selects 3 and 5 contingency plans, respectively. $\mathrm{CompAdapt}(\mathcal{P})$ is given by

$$\mathrm{CompAdapt}(\mathcal{P}) = \begin{array}{ll} \max: & y_1(\lambda) + y_2(\lambda) + y_3(\lambda) \\ \text{s.t.}: & \lambda \in [0, 1]. \end{array}$$

We can see from the figure that indeed this value is 3.

Next, consider the optimal affine adaptability. In (II.8) we define affine adaptability for the two stage problem, however we can easily apply this to single stage optimization by allowing all the decision-variables to depend affinely on the uncertainty. Here the uncertainty is one-dimensional, parameterized by $\lambda \in [0, 1]$, so we let $\boldsymbol{y}^{\mathrm{aff}}(\lambda)$ denote the optimal affine solution. The third component, $y_3^{\mathrm{aff}}(\lambda)$ must satisfy:

$y_3^{\mathrm{aff}}(0), y_3^{\mathrm{aff}}(1) \geq 1$. Therefore, by linearity, we must have $y_3^{\mathrm{aff}}(\lambda) \geq 1$ for all $\lambda \in [0,1]$. Furthermore, for $\lambda = 1/2$, we must also have

$$\frac{1}{2}\left(\frac{1}{2}y_1^{\mathrm{aff}}(1/2)+\frac{1}{5}y_2^{\mathrm{aff}}(1/2)\right)+\frac{1}{2}\left(\frac{1}{5}y_1^{\mathrm{aff}}(1/2)+\frac{1}{2}y_2^{\mathrm{aff}}(1/2)\right)$$
$$\geq 1,$$

which implies, in particular, that $y_1^{\mathrm{aff}}(1/2) + y_2^{\mathrm{aff}}(1/2) \geq \frac{20}{7}$. The cost obtained by affine adaptability is

$$\mathrm{Affine}(\mathcal{P}) = \max: \quad y_1^{\mathrm{aff}}(\lambda) + y_2^{\mathrm{aff}}(\lambda) + y_3^{\mathrm{aff}}(\lambda)$$
$$\mathrm{s.t.}: \quad \lambda \in [0,1].$$

This is at least the value at $\lambda = 1/2$. But this is: $y_1^{\mathrm{aff}}(1/2) + y_2^{\mathrm{aff}}(1/2) + y_3^{\mathrm{aff}}(1/2) \geq 20/7 + 1 = 27/7$, which is the static robust value. Therefore, in this case, affine adaptability is no better than the static robust value. On the other hand, as illustrated in Figure 2, 3-adaptability is sufficient to significantly improve the cost to the decision-maker, and 5-adaptability is better still. Moreover, since this problem satisfies the Continuity Assumption, by Proposition 1, $\mathrm{Adapt}_k(\mathcal{P}) \to \mathrm{CompAdapt}(\mathcal{P})$ as k increases, so we can further improve the cost with more adaptability. Thus, we illustrate point (4) from above. $\blacktriangle$

Next we illustrate points (5) and (6) above by presenting a modification of the previous example. Consider:

$$\min: \quad y_2$$
$$\mathrm{s.t.}: \quad \begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \geq \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$$
$$\forall \begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \end{pmatrix} \in \mathrm{conv}\left\{ \begin{pmatrix} \frac{1}{2} & \frac{1}{5} \\ -\frac{1}{2} & -\frac{1}{5} \end{pmatrix}, \begin{pmatrix} \frac{1}{5} & \frac{1}{2} \\ -\frac{1}{5} & -\frac{1}{2} \end{pmatrix} \right\}$$
$$(\mathrm{VI.21})$$

The static robust solution to (VI.21) is $\boldsymbol{y}_R = (10/7, 10/7)$, and hence $\mathrm{Static}(\mathcal{P}) = 10/7$. On the other hand, for any realization of the uncertain matrix,

$$\begin{pmatrix} a_1 & a_2 \\ b_1 & b_2 \end{pmatrix} = \lambda\begin{pmatrix} \frac{1}{2} & \frac{1}{5} \\ -\frac{1}{2} & -\frac{1}{5} \end{pmatrix} + (1-\lambda)\begin{pmatrix} \frac{1}{5} & \frac{1}{2} \\ -\frac{1}{5} & -\frac{1}{2} \end{pmatrix},$$

the solution $(y_1, y_2) = (5 - 3\lambda, 0)$ is feasible, and hence optimal for the nominal problem. The optimal response function in this case is affine. Here, $\mathrm{CompAdapt}(\mathcal{P}) = 0$, and the gap is $10/7$. Consider now any partition of the uncertainty set (i.e., the interval $[0,1]$) into finitely (or even countably many) regions. At least one region of the partition must contain more than one point of the interval, otherwise we would have uncountably many regions. Let $\tilde{\mathcal{P}}$ denote this region, with $\lambda_1 < \lambda_2$ both elements of $\tilde{\mathcal{P}}$. The static robust problem over this set $\tilde{\mathcal{P}}$, is lower bounded by

$$\min: \quad y_2$$
$$\mathrm{s.t.}: \quad X_1(\lambda_1)\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \geq \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$
$$X_2(\lambda_2)\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \geq \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$
$$y_2 \geq 0$$

where

$$X_1(\lambda_1) = \left[\lambda_1\begin{pmatrix} \frac{1}{2} & \frac{1}{5} \\ -\frac{1}{2} & -\frac{1}{5} \end{pmatrix} + (1-\lambda_1)\begin{pmatrix} \frac{1}{5} & \frac{1}{2} \\ -\frac{1}{5} & -\frac{1}{2} \end{pmatrix}\right],$$
$$X_2(\lambda_2) = \left[\lambda_2\begin{pmatrix} \frac{1}{2} & \frac{1}{5} \\ -\frac{1}{2} & -\frac{1}{5} \end{pmatrix} + (1-\lambda_2)\begin{pmatrix} \frac{1}{5} & \frac{1}{2} \\ -\frac{1}{5} & -\frac{1}{2} \end{pmatrix}\right].$$

As $\lambda_1 \neq \lambda_2$, the point $\boldsymbol{y}_R = (10/7, 10/7)$ is the only point in the feasible region, and thus it must also be optimal; hence the value is not improved from $10/7$. Note, moreover, that this example violates the Continuity Assumption: for any two (even infinitesimally close) realizations of the uncertainty, the only common feasible point is $\boldsymbol{y}_R = (10/7, 10/7)$, which is not within ε of optimality for any $\varepsilon < 10/7$. Thus, we illustrate point (5), and show that the Continuity Assumption may not be removed from Proposition 1. Recall that Proposition 2 says that finite adaptability can strictly improve the solution if and only if $\mathcal{P} \cap \mathcal{M} = \emptyset$. Here, we can indeed check that $\mathcal{P} \cap \mathcal{M} = \emptyset$. However, the set of dual optimal solutions to (VI.21) is unbounded, and the set $\mathcal{M}$ is not closed. With some work, we can check that, e.g.,

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{5} \\ -\frac{1}{2} & -\frac{1}{5} \\ 0 & 1 \end{pmatrix} \in \bar{\mathcal{M}}.$$

Thus, the conclusion of Proposition 2 holds, and in particular, as we point out in (6) above, taking the closure of $\mathcal{M}_\eta$ cannot be relaxed. $\blacktriangle$

It turns out (see [11]) that with quartic polynomial adaptability, or with piecewise affine adaptability, one can recover the optimal solution.

VII. HEURISTIC ALGORITHMS

In large scale optimization problems such as the one discussed in Section VIII-C, we seek practically efficient and implementable solutions. In this section, we propose a heuristic tractable algorithm. We restrict ourselves to an infinite class of partitions from which selecting the optimal partition can be done efficiently.

The algorithm is motivated by the results of Section III. There, the necessary conditions we derive say that good partitions divide points of $\mathcal{P}$ which must be separated. We try to do exactly that. The algorithm is based on the following observation, whose proof is immediate.

Lemma 3: Consider the set of partitions $\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2$ given by a hyperplane division of $\mathcal{P}$. If the orientation (i.e., the normal vector) of the hyperplane is given, then selecting the optimal partitioning hyperplane with this normal can be done efficiently.

Algorithm 1: Let $\mathcal{P} = \mathrm{conv}\{(\boldsymbol{A}, \boldsymbol{B})^1, \ldots, (\boldsymbol{A}, \boldsymbol{B})^K\}$.

1. For every pair (i,j), $1 \leq i \neq j \leq K$, let $\boldsymbol{v}_{ij} = (\boldsymbol{A}, \boldsymbol{B})^j - (\boldsymbol{A}, \boldsymbol{B})^i$ be the unique vector they define.
2. Consider the family of hyperplanes with normal $\boldsymbol{v}_{ij}$.
3. Solve the quasi-convex problem, and let $\boldsymbol{H}_{ij}$ be the hyperplane that defines the optimal hyperplane partition of $\mathcal{P}$ within this family.
4. Select the optimal pair (i,j) and the corresponding optimal hyperplane partition of $\mathcal{P}$.

This algorithm can be applied iteratively as a heuristic approach to computing 2^d-adaptability. In Section VIII, we implement this algorithm to compute 2,4-adaptability.

Section III-B provides an approach to strengthen the above algorithm. The algorithm selects the optimal hyperplane from the set of hyperplanes that have normal vector defined by a pair of extreme points of $\mathcal{P}$. By adding explicitly more points that are in the interior of $\mathcal{P}$, we enlarge the space of hyperplanes over which the algorithm searches. In Section III-B, we illustrate a procedure for obtaining necessary conditions that any "good" partition must satisfy. These conditions essentially contain requirements that certain collections of points of $\mathcal{P}$ should not be contained within any single region of the partition. By including (a partial set of) the points corresponding to a list of necessary conditions, we guarantee that the set of partitions considered include partitions that meet the necessary conditions. In effect, this gives a structured approach to increasing the size of the family of partitions considered.

Algorithm 2: Let the uncertainty set be given by inequalities: $\mathcal{P} = \{(\boldsymbol{A}, \boldsymbol{B}) : \boldsymbol{\alpha}_i' \mathrm{vec}((\boldsymbol{A}, \boldsymbol{B})) \leq 1, 1 \leq i \leq N\}$, where $\mathrm{vec}(\boldsymbol{B})$ is vector consisting of the rows of $\boldsymbol{B}$.

1. For every defining facet $\boldsymbol{\alpha}_i$ of $\mathcal{P}$, let $\boldsymbol{v}$ be the unique normal vector.
2. Consider the family of hyperplanes with normal $\boldsymbol{v}$.
3. Solve the quasi-convex problem, and let $\boldsymbol{H}_i$ be the hyperplane that defines the optimal hyperplane partition of $\mathcal{P}$ within this family.
4. Select the optimal index i and the corresponding optimal hyperplane partition of $\mathcal{P}$.

VIII. COMPUTATIONAL EXAMPLES

In this section, we report on the performance of the heuristic algorithm of Section VII. In Section VIII-A, we consider a minimum cost robust scheduling problem with integer constraints. These randomly generated examples are meant to illustrate the applicability of k-adaptability, and some types of problems that can be considered. In the final part of this section, VIII-B, we explore a large collection of randomly generated instances of the scheduling problem without integer constraints. We consider different problem size, and types and level of uncertainty, in an effort to obtain some appreciation in the generic case, for the benefit of the first few levels of the adaptability hierarchy, and for the behavior of the algorithm of Section VII.

Finally, we discuss the problem of Air Traffic Control. We discuss why finite adaptability may be an appropriate framework for adaptability, both in terms of theoretical and practical considerations. The full details of the model, and the numerical computations are in [7].

A. Robust Scheduling: Integer Constraints

Suppose we have m products, and each product can be completed partially or fully at one of n stations, and the stations work on many products simultaneously so that no product blocks another. Thus the decision variables, y_j, are

for how long to operate station j. The matrix $\boldsymbol{B} = \{b_{ij}\}$ gives the rate of completion of product i at station j. Running station j for one hour we incur a cost c_j. To minimize the cost subject to the constraint that the work on all products is completed, we solve:

$$\begin{aligned} \min : \quad & x \\ \mathrm{s.t.} : \quad & x \geq \textstyle\sum_{j=1}^{n} c_j y_j \\ & \textstyle\sum_{j=1}^{n} b_{ij} y_j \geq 1, \quad 1 \leq i \leq m \\ & y_j \geq 0, \quad\quad\quad\; 1 \leq j \leq n. \end{aligned}$$

In the static robust version of the problem, the rate matrix $\boldsymbol{B}$ is only known to lie in some set $\mathcal{P}$. How much can we reduce our cost if we can formulate 2 (in general k) schedules rather than just one? Particularly in the case where we have to make binary decisions about which stations to use, there may be some cost in having k contingency plans prepared, as opposed to just one. It is therefore natural to seek to understand the value of k-adaptability, so the optimal trade-off may be selected.

In Section VIII-B, we generate a large ensemble of these problems, varying the size and the generation procedure, and we report average results. Here, we consider only one instance from one of the families below, and impose binary constraints, so that each station must be either on or off: $y_i \in \{0, 1\}$.

The heuristic algorithms proposed in Section VII are tractable because of the quasi-convexity of the search for the optimal dividing hyperplane and by the limited set of normal directions considered. Both these factors are independent of the continuous or discrete nature of the underlying problem. Indeed, all that is required for the algorithms is a method to solve the static robust problem.

We consider an instance with 6 products and 6 stations, where the uncertainty set is the convex hull of six randomly generated rate matrices. Without the integer constraints, the value of the static robust problem is 3.2697, and the completely adaptable value is bounded below by 2.8485. The value of the 2-adaptability solution is 3.1610, and for 4-adaptability the value is 3.0978. Thus, 2-adaptability covers 25.8% of the gap, and 4-adaptability covers just over 40% of the gap. As we see from the results of the next section, these numbers are typical in our ensemble. When we add integer constraints, the static robust cost is 5, i.e., 5 stations must be turned on. The completely adaptable value is 4. The 2-adaptability solution also improves the static robust cost, lowering it to 4. Thus, in this case a single split of the uncertainty region reduces the cost as much as the full completely adaptable formulation.

B. Robust Scheduling

We consider a large collection of randomly generated instances of the scheduling problem above, without integer constraints. First, we suppose that the extreme points of $\mathcal{P}$ are generated uniformly at random, their elements drawn *iid* from a uniform distribution. Next, we consider another random instance generation procedure, where the extreme points of $\mathcal{P}$ come from a specific degrading of some number of products. That is, we may have nominal values $\{b_{ij}\}$, but in actuality some collection (typically small) of the m products may take longer to complete on each machine, than

indicated by the nominal values. Here each extreme point of $\mathcal{P}$ would be constructed from the nominal matrix B, degraded at some small number of rows. We generate random instances of this problem by generating a nominal matrix B, and then degrading each row individually. This corresponds to choosing robustness that protects against a single product being problematic and requiring more time at the stations.

We are interested in several figures of merit. We consider the gap between the static robust problem and complete adaptability. As we have remarked above, we note that complete adaptability is typically difficult to compute exactly ([2]). Therefore for all the computations in this section, we compute upper bounds on the gap between the static robust and the completely adaptable values. Thus, we present lower bounds on the benefit of adaptability and the performance of the heuristic algorithm. We obtain upper bounds on the gap by approximating the completely adaptable value by random sampling. We sample 500 points independently and uniformly at random from the uncertainty set. Since the truly worst case may not be close to one of these sampled points, the completely adaptable value may in fact be worse than reported, thus making the gap *smaller*. Thus our random approximation gives a conservative bound on the true gap. Next, we compute the extent to which 2- and 4-adaptability, as computed by the algorithm of Section VII, close this gap.

We summarize the computational examples by reporting the size of the instances and some statistics of the simulations. In each category, every number represents the average of 50 independently generated problem instances of size as shown. These results are contained in Table I. There, we give the average, minimum, and maximum gap between the static robust and the completely adaptable values. We give this as a fraction of the static robust value, that is, GAP $= (\text{Static}(\mathcal{P}) - \text{CompAdapt}(\mathcal{P}))/\text{Static}(\mathcal{P})$. Then we report the average percentage of this gap covered by 2-adaptability and 4-adaptability, as computed by the heuristic algorithm.

Matrix Size	Size of $\mathcal{P}$	Avg Gap %	2-Adapt %	4-Adapt %
6×6	$K = 3$	10.10	63.22	70.70
6×6	$K = 6$	14.75	42.72	52.33
6×6	$K = 8$	18.45	39.15	47.42
10×10	$K = 3$	10.12	50.67	63.29
10×10	$K = 5$	14.22	38.58	49.36
10×10	$K = 10$	18.27	31.17	40.18
15×25	$K = 3$	8.06	39.27	54.53
15×25	$K = 5$	10.73	25.12	35.52
15×25	$K = 7$	13.15	18.21	26.84

TABLE I

THE MATRICES IN THESE INSTANCES WERE GENERATED INDEPENDENTLY. THE FIRST GROUP OF TWO COLUMNS IDENTIFIES THE SIZE OF THE PROBLEM, WHERE BY MATRIX SIZE WE MEAN THE "NUMBER OF PRODUCTS BY NUMBER OF STATIONS," AND BY SIZE OF $\mathcal{P}$ WE INDICATE THE NUMBER OF EXTREME POINTS. WE NOTE THAT THE AVERAGE GAP BETWEEN THE STATIC AND ADAPTABLE FORMULATIONS INCREASES WITH THE SIZE OF THE UNCERTAINTY SET $\mathcal{P}$. ALSO, THE BENEFIT OF 2,4-ADAPTABILITY DECREASES AS THE SIZE OF THE SET $\mathcal{P}$ INCREASES.

The table illustrates several properties of the gap, and of adaptability. We have considered several examples where we fix the number of products and the number of stations (i.e., we fix the size of the matrices) and then vary the size of

the uncertainty set, i.e., the number of extreme points. In all such examples, we see that the average gap increases as the level of the uncertainty grows. Indeed, this is as one would expect. Furthermore, we see that the quality of 2,4-adaptability decreases as the size of the uncertainty set grows. Again this is as one would expect, as we are keeping the amount of adaptability, and the problem dimension constant, while increasing the number of extreme points of the uncertainty set. For the 6×6 matrices, 4-adaptability covers, on average, over 70% of the gap. That is, with only 4 contingency plans, on average we do over 70% as well as the best possible attainable by any amount of adaptability. When we double the size of $\mathcal{P}$, the average performance of 2-adaptability drops from over 63% to just over 42%, while the performance of 4-adaptability drops from over 70% to about 52%. A similar phenomenon occurs in the other examples as well.

We also report the results of the computations for the case where the uncertainty set $\mathcal{P}$ corresponds to the case where at most one product is degraded. That is, we form $\mathcal{P}$ by degrading each row of a matrix B individually. The results from this random generation procedure are comparable to the first procedure. The results are reported in Table II.

Matrix Size	Size of $\mathcal{P}$	Avg Gap %	2-Adapt %	4-Adapt %
3×6	$K = 3$	9.08	70.61	78.36
6×6	$K = 6$	14.24	54.62	66.34
3×20	$K = 3$	10.61	28.67	45.16
5×20	$K = 5$	15.90	33.78	47.60
10×20	$K = 10$	21.17	22.35	31.50
3×25	$K = 3$	10.92	52.94	65.16
5×25	$K = 5$	15.81	32.83	45.90
3×50	$K = 3$	10.66	44.04	59.06

TABLE II

THE MATRICES IN THESE INSTANCES WERE GENERATED WITH DEPENDENT MATRICES, AS EXPLAINED ABOVE. IN THIS EXAMPLE AGAIN WE NOTE THE SAME TRENDS AS FOR THE FIRST EXAMPLE: THE GAP BETWEEN THE STATIC AND THE ADAPTABLE INCREASES WITH THE SIZE OF THE UNCERTAINTY SET, AND THE VALUE OF 2,4-ADAPTABILITY IS BETTER FOR LOW-DIMENSIONAL UNCERTAINTY SETS THAN FOR HIGH-DIMENSIONAL UNCERTAINTY.

C. An Example from Air Traffic Control

There are about 30,000 flights daily over the United States National Air Space (NAS). These flights must be scheduled so that they do not exceed the takeoff or landing capacity of any airport, or when they are the capacity of any sector of the NAS while they are in-flight. These capacities are uncertain, as they are impacted by the weather. Currently, there is no centralized, optimization-based approach implemented to obtain a schedule that respects the capacity constraints while minimizing delays. The primary challenges are (a) the large scale nature of the problem, with over a million variables and constraints; (b) the variables are inherently discrete; (c) the problem is naturally multi-stage: scheduling decisions are made sequentially, and the uncertainty is also revealed throughout the day, as we have access to the current forecast at every point in time. Because of the discrete variables, continuous adaptability cannot work. Also, because of the large-scale nature of the problem, there is very little leeway to increase the size of the problem.

Finite Adaptability, is an appropriate framework to address all three of the above challenges. We given a small example (see [11] for more details and computations) to illustrate the application of adaptability, showing finite adaptability can significantly decrease the impact of a storm on flight delay and cancellation.

Figure 3 depicts a major airport (e.g., JFK) that accepts heavy traffic from airports to the West and the South. In this figure, the weather forecast precits major disruption due to an approaching storm; the timing of the impact, however, is uncertain, and at question is which of the 50 (say) Northbound and 50 Easthbound flights to hold on the ground, and which in the air.
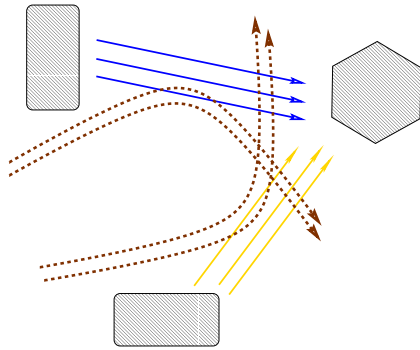

Fig. 3.   We have planes arriving at a single hub such as JFK in NYC. Dashed lines express uncertainty in the weather.

The minimum flight time is 2 hours. Each plane may be held either on the ground, in the air, or both, for a total delay not exceeding 60 minutes. Therefore all 50 Northbound and 50 Eastbound planes land by the end of the three hour window under consideration.

We discretize time into 10-minute intervals. We assume that the impact of the storm lasts 30 minutes. The uncertainty is in the timing of the storm, and the order in which it will affect the capacity of the southward and westward approaches. There is essentially a single continuous parameter here, controls the timing of the storm, and whether the most severe capacity impact hits the approach from the south before, after, or at the same time as it hits the approach from the west. Because we are discretizing time into 10 minute intervals, there are four possible realizations of the weather-impacted capacities in the second hour of our horizon. These four scenarios are as follows. We give the capacity in terms of the number of planes per 10-minute interval:

$$(1) \quad \begin{bmatrix} \text{West:} & 15 & 15 & 15 & \underline{5} & \underline{5} & \underline{5} \\ \text{South:} & \underline{5} & \underline{5} & \underline{5} & 15 & 15 & 15 \end{bmatrix}$$

$$(2) \quad \begin{bmatrix} \text{West:} & 15 & 15 & \underline{5} & \underline{5} & \underline{5} & 15 \\ \text{South:} & 15 & \underline{5} & \underline{5} & \underline{5} & 15 & 15 \end{bmatrix}$$

$$(3) \quad \begin{bmatrix} \text{West:} & 15 & \underline{5} & \underline{5} & \underline{5} & 15 & 15 \\ \text{South:} & 15 & 15 & \underline{5} & \underline{5} & \underline{5} & 15 \end{bmatrix}$$

$$(4) \quad \begin{bmatrix} \text{West:} & \underline{5} & \underline{5} & \underline{5} & 15 & 15 & 15 \\ \text{South:} & 15 & 15 & 15 & \underline{5} & \underline{5} & \underline{5} \end{bmatrix}$$

In the utopic set-up (not implementable) the decision-maker can foresee the future (of the storm). Thus we get a bound on performance. We also consider a nominal, no-robustness scheme, where the decision-maker assumes the storm will behave exactly according to the first scenario. We also consider adaptabiliy formulations: 1-adaptable (static robust) solution, then the 2- and 4-adaptable solution.

The cost is computed from the total amount of ground holding and the total amount of air holding. Each 10-minute interval that a single flight is delayed on the ground, contributes 10 units to the cost. Each 10-minute interval of air-delay contributes 20 units.

	Delay Cost	Ground Holding	Air Holding
Utopic:	2,050	205	0
Static:	4,000	400	0
2-Adaptable:	3,300	170	80
4-Adaptable:	2,900	130	80

TABLE III

RESULTS FOR THE COST OF TOTAL DELAY, AS WELL AS THE TOTAL GROUND-HOLDING TIME, AND AIR-HOLDING TIME, FOR THE UTOPIC, ROBUST, 2-ADAPTABLE, AND 4-ADAPTABLE SCHEMES, FOR THE AIR TRAFFIC CONTROL EXAMPLE. THE GROUND- AND AIR-HOLDING TIME IS GIVEN AS THE NUMBER OF 10 MINUTE SEGMENTS INCURRED BY EACH FLIGHT (SO IF A SINGLE FLIGHT IS DELAYED BY 40 MINUTES, IT CONTRIBUTES 4 TO THIS COUNT).

In Table IV, we give the cost of the nominal solution, depending on what the actual realization turns out to be.

	Real'n 1	Real'n 2	Real'n 3	Real'n 4
Nominal Cost:	2,050	2,950	3,950	4,750

TABLE IV

RESULTS FOR THE COST OF TOTAL DELAY FOR EACH SCENARIO, WHEN THE FIRST-STAGE SOLUTION IS CHOSEN WITHOUT ROBUSTNESS CONSIDERATIONS, ASSUMING THAT THE FIRST REALIZATION IS IN FACT THE TRUE REALIZATION.

IX. CONCLUSION

We have proposed a notion of finite adaptability. This corresponds to choosing a finite number of contingency plans, as opposed to a single static robust solution. We have shown that this is equivalent to partitioning the uncertainty space, and receiving ahead of time coarse information about the realization of the uncertainty, corresponding to one of the chosen partitions.

The structure of this adaptability is designed to reduce the geometric gap between $\mathcal{P}$ and $(\mathcal{P})_R$, which is exactly the reason the static robust solution may be conservative. In this paper, we have focused on exploiting non-constraintwise uncertainty. We consider elsewhere the value of adaptability in the face of non-convex uncertainty sets. This notion of finite adaptability establishes a hierarchy of adaptability that bridges the gap between the static robust formulation, and the completely adaptable formulation. Thus, we introduce the concept of the value of adaptability. We believe that the finiteness of the proposal, as well as the hierarchy of increasing adaptability, are central to the paper. The finiteness of the adaptability is appropriate in many application areas where infinite adjustability, and infinitesimal sensitivity, are either

impossible due to the constraints of the problem, or undesirable because of the structure of the optimization, i.e., the cost. In addition to this, the inherent finiteness, and hence discrete nature of the proposal, makes it suitable to address adaptability problems with discrete variables. We expect that this benefit should extend to problems with non-convex constraints.

In problems where adaptability, or information is the scarce resource, the hierarchy of finite adaptability provides an opportunity to trade off the benefits of increased adaptability, versus its cost.

On the other hand, as we demonstrate, obtaining optimal partitions of the uncertainty space can be hard. Thus, there is a need for efficient algorithms. We have proposed a tractable algorithm for adaptability. Numerical evidence indicates that its behavior is good.

REFERENCES

[1] A. Atamtürk and M. Zhang. Two-stage robust network flow and design under demand uncertainty. Technical Report BCOL.04.03, IEOR, University of California–Berkeley, December 2004.

[2] A. Ben-Tal, A. Goryashko, E. Guslitzer, and A. Nemirovski. Adjustable robust solutions of uncertain linear programs. *Math. Programming*, 99:351–376, 2003.

[3] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Math. Oper. Res.*, 23:769–805, 1998.

[4] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Math. Programming*, 88:411–421, 2000.

[5] D. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.

[6] D. Bertsekas and J.N. Tsitsiklis. *Neuro Dynamic Programming*. Athena Scientific, 1996.

[7] D. Bertsimas and C. Caramanis. Multistage finite adaptability: Application to air traffic control. Technical Report In preparation, M.I.T., 2006.

[8] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Mathematical Programming Series B*, 98:49–71, 2003.

[9] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[10] J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer-Verlag, 1997.

[11] C. Caramanis. *Adaptable Optimization: Theory and Algorithms*. PhD dissertation, Massachusetts Institute of Technology, June 2006.

[12] X. Chen, M. Sim, and P. Sun. A robust optimization perspective of stochastic programming. *Submitted*, 2005.

[13] G.B. Dantzig and A. Madansky. On the solution of two-stage linear programs under uncertainty. In *Proceedings of the Fourth Berkeley Symposium on Statistics and Probability*, volume 1, pages 165–176. University of California Press, 1961.

[14] E. Erdoğan, D. Goldfarb, and G. Iyengar. Robust portfolio management. Technical Report CORC TR-2004-11, IEOR, Columbia University, November 2004.

[15] C. Floudas and V. Visweswaran. *Quadratic Optimization*, pages 217–270. Handbook of Global Optimizaion. Kluwer Academic Publishers, 1994.

[16] K. Fukuda. CDD. See http://www.cs.mcgill.ca/~/fukuda/soft

[17] M. Garey and D. Johnson. *Computers and Intractability*. W.H. Freeman, 1979.

[18] L. El Ghaoui, F. Oustry, and H. Lebret. Robust solutions to uncertain semidefinite programs. *Siam J. Optimization*, 9(1), 1998.

[19] W.K.K. Haneveld and M.H. van der Vlerk. Stochastic integer programming. *Annals of Operations Research*, 85:39–57, 1998.

[20] B. Korte and J. Vygen. *Combinatorial Optimization*. Springer-Verlag, 2002.

[21] A. Prékopa. Programming under probabilistic constraint and maximizing a probability under constraints. Technical Report RRR 35-93, Rutgers University, December 1993.

[22] A. Prékopa. *Stochastic Programming*. Kluwer, 1995.

[23] R. T. Rockafellar. Optimization under uncertainty. Lecture Notes.

[24] A. Shapiro. On complexity of multistage stochastic programs. *Optimization Online*, 2005.

[25] A. Shapiro and A. Nemirovski. On complexity of stochastic programming problems. Technical Report available from: http://www.optimization-online.org, Georgia Tech, 2005.

[26] H. Sherali and A. Alameddine. A new reformulation-linearization technique for bilinear programming problems. *Journal of Global Optimization*, 2:379–410, 1992.

[27] H. Sherali and C.H. Tuncbilek. A reformulation-convexification approach for solving nonconvex quadratic programming problems. *Journal of Global Optimization*, 7:1–31, 1995.

[28] A.K. Singh, K. He, C. Caramanis, and M. Orshansky. Mitigation of intra-array sram variability using adaptive voltage architecture. In *Proceedings of ICCAD*, 2009.

[29] A. Takeda, S. Taguchi, and R. Tütüncü. Adjustable robust optimization models for nonlinear multi-period optimization. *Submitted*, 2004.

Dimitris Bertsimas Dimitris Bertsimas is currently the Boeing Professor of Operations Research and the codirector of the Operations Research Center at the Massachusetts Institute of Technology. He has received a BS in Electrical Engineering and Computer Science at the National Technical University of Athens, Greece in 1985, a MS in Operations Research at MIT in 1987, and a Ph.D in Applied Mathematics and Operations Research at MIT in 1988. Since 1988, he has been in the MIT faculty.

His research interests include optimization, stochastic systems, data mining, and their application. In recent years he has worked in robust and adaptive optimization, health care and finance. He has co-authored more than 120 scientific papers and 3 graduate level textbooks. He is a member of the National Academy of Engineering, and area editor for optimization in Management Science. He has received numerous research awards including the Farkas prize (2008), the Erlang prize (1996), the SIAM prize in optimization (1996), the Bodossaki prize (1998) and the Presidential Young Investigator award (1991-1996).

Constantine Caramanis Constantine Caramanis obtained the A.B. degree in Mathematics from Harvard University, and the MS and PhD degrees from the Massachusetts Institute of Technology. Since 2006, he has been on the faculty in Electrical and Computer Engineering at The University of Texas at Austin. His research interests include robust and adaptable optimization, combinatorial optimization, statistics, machine learning and control, with applications to large scale networks.