

# Distributed Robust Optimization (DRO)

## Part I: Framework and Example

Kai Yang<sup>1,4</sup>, Jianwei Huang<sup>2</sup>, Yihong Wu<sup>3</sup>, Xiaodong Wang<sup>1</sup>, Mung Chiang<sup>3</sup>

<sup>1</sup>Dept. of Electrical Engineering, Columbia University, New York, NY, USA

<sup>2</sup>Dept. of Information Engineering, The Chinese University of Hong Kong, China

<sup>3</sup>Dept. of Electrical Engineering, Princeton University, Princeton, NJ, USA

<sup>4</sup>NEC Laboratories America, Princeton, NJ, USA

{kyee, wangx}@ee.columbia.edu, {yihongwu, chiangm}@princeton.edu,

jwhuang@ie.cuhk.edu.hk

### Abstract

Robustness of optimization models for networking problems has been an under-explored area. Most existing algorithms for solving robust optimization problems are centralized, thus not suitable for many communication networking problems that demand distributed solutions. This paper represents the first step towards a systematic theory for designing distributed *and* robust optimization models and algorithms. We first discuss several models for describing parameter uncertainty sets that can lead to decomposable problem structures and thus distributed solutions. These models include general ellipsoid, polyhedron, and  $D$ -norm. We then apply these models in solving a robust rate control problem in wireline networks. Tradeoffs among performance, robustness, and distributiveness are illustrated both analytically and through simulations. In Part II of this two-part paper, extensive applications to wireless power control are presented using the framework of DRO.

J. Huang is supported by the Competitive Earmarked Research Grants (Project Number 412308) established under the University Grant Committee of the Hong Kong Special Administrative Region, China, the Direct Grant (Project Number C001-2050398) of The Chinese University of Hong Kong, and the National Key Technology R&D Program (Project Number 2007BAH17B04) established by the Ministry of Science and Technology of the People's Republic of China.

## I. INTRODUCTION

Despite the importance and success of using optimization theory to study communication and networking problems, most work in this area makes the unrealistic assumption that the data defining the constraints and objective function of the optimization problem can be obtained precisely. We call the corresponding problems “nominal”. However, in many practical problems, these data are typically inaccurate, uncertain, or time-varying. Solving the nominal optimization problems may lead to poor or even infeasible solutions of the real problem.

Over the last ten years, robust optimization has emerged in the Operations Research community as a field that tackles optimization problems under data uncertainty (e.g., [1], [2], [3], [4], [5]). The basic idea of robust optimization is to seek a solution which remains feasible and near-optimal under the perturbation of parameters in the optimization problem. Each robust optimization is defined by three-tuple: *a nominal formulation, a definition of robustness, and a representation of the uncertainty set*. The process of making an optimization formulation robust can be viewed as a mapping that maps from one optimization problem to another. A central question is as follows: when will important properties, such as convexity and decomposability, be preserved under such mapping? In particular, what kind of nominal formulation and uncertainty set representation will preserve convexity and decomposability in the robust version of the optimization problem?

So far, almost all of the work on robust optimization focuses on determining what representations of data uncertainty preserves convexity, thus tractability through a centralized solution, in the robust counter part of the nominal problem for a given definition of robustness. For example, for worst-case robustness, it has been shown that under the assumption of ellipsoid set of data uncertainty, a robust linear optimization problem can be converted into a second-order cone problem; and a robust second-order cone problem can be reformulated as a semi-definite optimization problem [6]. In general, the previous focus in this research area is to formulate the robust optimization problem such that it preserves the *convexity* of the original nominal problem, such that we can use efficient centralized algorithm (e.g., interior point method) to solve it. In this paper, we will focus instead on the *distributiveness*-preserving formulation of the robust optimization, which is desirable for many practical problems in networking. The driving question is: how much more communication overhead is introduced in making the problem robust?

To develop a systematic theory of Distributed Robust Optimization, we first show how to represent an uncertainty set, which not only captures the data uncertainty in the model but also leads to a distributively solvable optimization problem. Second, in the case where fully distributed algorithm is not obtainable, we focus on the investigation between the *tradeoff* between robustness and distributiveness. Distributed algorithms are often developed based on decomposability structure of the problem, which may disappear as the optimization formulation is made robust. While distributed computation has long been studied [7], unlike convexity of a problem, distributiveness of an algorithm does not have a widely-agreed definition. It is often quantified by the amount of communication overhead required: how far and how frequent do the nodes have to pass message around? Zero communication overhead is obviously the “most distributed”, and we will see how the amount of overhead trades-off with the degree of robustness.

In Section II, we develop the framework of DRO, with a focus on the characterization of uncertainty sets that are useful for designing distributed algorithms. An example on robust rate control is given in Section III, where we discuss various tradeoffs between robustness, distributiveness, and performance through both analysis and numerical studies. Conclusions to this part are given in Section IV. Due to space limitation, extensive applications of DRO to wireless power control are presented in Part II of the paper.

## II. GENERAL FRAMEWORK

To make our discussions concrete, we will focus on a class of optimization problems with the following nominal form: maximization of a *concave* objective function over a given data set characterized by *linear* constraints,

$$\begin{aligned} & \text{maximize } f_0(\mathbf{x}) & (1) \\ & \text{subject to } \mathbf{Ax} \preceq \mathbf{b} \\ & \text{variables } \mathbf{x}, \end{aligned}$$

where  $\mathbf{A}$  is an  $M \times N$  matrix,  $\mathbf{x}$  is an  $N \times 1$  vector, and  $\mathbf{b}$  is an  $M \times 1$  vector. This class of problems can model a wide range of engineering systems (e.g., [8], [9], [10], [11], [12]). Generalizing to nonlinear and convex constraint sets presents a set of major future work.

The uncertainty of Problem (1) may exist in the objective function  $f_0$ , matrix parameter  $\mathbf{A}$ ,

and vector parameter  $\mathbf{b}$ . In many cases, the uncertainty in objective function  $f_0$  can be converted into uncertainty of the parameters defining constraints [13]. And later in Section III we show that it is also possible to convert the uncertainty in  $\mathbf{b}$  into uncertainty in  $\mathbf{A}$  (although this could be difficult in general). Therefore, in the rest of the paper, we will focus on studying the uncertainty in  $\mathbf{A}$ . In many networking problems, structures and physical meaning of matrix  $\mathbf{A}$  readily leads to distributed algorithms, e.g., in rate control where  $\mathbf{A}$  is a given routing matrix, distributed subgradient algorithm is well-known to solve the problem, with an interesting correspondence with the practical protocol of TCP. Making the optimization robust may turn the linear constraints into nonlinear ones and increase the amount of message passing. Quantifying the tradeoff between robustness and distributedness is a main subject to study in this paper.

In the robust counterpart of Problem (1), we require the constraints  $\mathbf{A}\mathbf{x} \preceq \mathbf{b}$  to be valid for any  $\mathbf{A} \in \mathcal{A}$ , where  $\mathcal{A}$  denotes the uncertainty set of  $\mathbf{A}$ , and the definition of robustness is in the *worst-case* sense [14]. If we allow an arbitrary uncertainty set  $\mathcal{A}$ , then the robust optimization problem is difficult to solve even in a centralized manner [15]. In this paper, we will focus on the study of *constraint-wise* (i.e. *row-wise*) uncertainty set, where the uncertainties between different rows in matrix  $\mathbf{A}$  are decoupled. This restricted class of uncertainty set characterizes the data uncertainty in many practical problems, and it also allows us to convert the robust optimization problem into a formulation that is distributively solvable. Tackling more general forms of uncertainties is another direction of extension in future work.

Denote the  $j^{\text{th}}$  row of  $\mathbf{A}$  be  $\mathbf{a}_j^T$ , which lies in a compact uncertainty set  $\mathcal{A}_j$ . Then the *robust* optimization problem that we focus on in this paper can be written in the following form:

$$\begin{aligned} & \text{maximize } f_0(\mathbf{x}), \\ & \text{subject to } \mathbf{a}_j^T \mathbf{x} \leq b_j, \forall \mathbf{a}_j \in \mathcal{A}_j, \forall 1 \leq j \leq M \\ & \text{variables } \mathbf{x}. \end{aligned} \tag{2}$$

We show that the robust optimization problem (2) can be equivalently written in a form represented by *protection functions* instead of uncertainty sets. Denote the nominal counterpart of problem (2) with a coefficient matrix  $\bar{\mathbf{A}}$  (i.e., the values when there is no uncertainty), with the  $j^{\text{th}}$  row's coefficient  $\bar{\mathbf{a}}_j \in \mathcal{A}_j$ . Then

**Proposition 1.** *Problem (2) is equivalent to the following convex optimization problem:*

$$\begin{aligned}
& \text{maximize } f_0(\mathbf{x}), \\
& \text{subject to } \bar{\mathbf{a}}_j^T \mathbf{x} + g_j(\mathbf{x}) \leq b_j, \quad \forall 1 \leq j \leq M \\
& \text{variables } \mathbf{x},
\end{aligned} \tag{3}$$

where

$$g_j(\mathbf{x}) = \sup_{\mathbf{a}_j \in \mathcal{A}_j} (\mathbf{a}_j - \bar{\mathbf{a}}_j)^T \mathbf{x} \tag{4}$$

is the protection function for the  $j^{\text{th}}$  constraint, which depends on the uncertainty set  $\mathcal{A}_j$  and the nominal row  $\bar{\mathbf{a}}_j$ .  $g_j(\mathbf{x})$  is a convex function for each  $j$ .

Different forms of  $\mathcal{A}_j$  will lead to different protection function  $g_j(\mathbf{x})$ , which results in different robustness and performance tradeoff of the formulation. Next we consider several approaches in terms of modeling  $\mathcal{A}_j$  and the corresponding protection function  $g_j(\mathbf{x})$ .

#### A. Robust Formulation Defined By General Polyhedron

In this case, the uncertainty set  $\mathcal{A}_j$  is a polyhedron characterized by a set of linear inequalities, i.e.,  $\mathcal{A}_j \triangleq \{\mathbf{a}_j : \mathbf{D}_j \mathbf{a}_j \preceq \mathbf{c}_j\}$ . The protection function is

$$g_j(\mathbf{x}) = \max_{\mathbf{a}_j : \mathbf{D}_j \mathbf{a}_j \preceq \mathbf{c}_j} (\mathbf{a}_j - \bar{\mathbf{a}}_j)^T \mathbf{x}, \tag{5}$$

which involves a linear program (LP). We next show that the uncertainty set can be translated into a set of linear constraints. In the  $j^{\text{th}}$  constraint in (2), with  $\mathbf{x} = \hat{\mathbf{x}}$  fixed, we can characterize the set  $\forall \mathbf{a}_j \in \mathcal{A}_j$  by comparing  $b_j$  with the outcome of the following LP:

$$v_j^* = \max_{\mathbf{a}_j : \mathbf{D}_j \mathbf{a}_j \preceq \mathbf{c}_j} \mathbf{a}_j^T \hat{\mathbf{x}}. \tag{6}$$

If  $v_j^* \leq b_j$ , then  $\hat{\mathbf{x}}$  is feasible for (2). However, this approach is not very useful since it requires solving one LP in (6) for each possible  $\hat{\mathbf{x}}$ . Alternatively, we take the Lagrange dual problem of the LP in (6),

$$v_j^* = \min_{\mathbf{p}_j : \mathbf{D}_j^T \mathbf{p}_j \succeq \hat{\mathbf{x}}, \mathbf{p}_j \succeq \mathbf{0}} \mathbf{c}_j^T \mathbf{p}_j. \tag{7}$$

If we can find a feasible solution  $\hat{\mathbf{p}}_j$  for (7), and  $\mathbf{c}_j^T \hat{\mathbf{p}}_j \leq b_j$ , then we must have  $v_j^* \leq \mathbf{c}_j^T \hat{\mathbf{p}}_j \leq b_j$ . We can thus replace constraint in (2) by the following constraints:

$$\mathbf{c}_j^T \mathbf{p}_j \leq b_j, \mathbf{D}_j^T \mathbf{p}_j \succeq \mathbf{x}, \mathbf{p}_j \succeq \mathbf{0}, \forall 1 \leq j \leq M, \quad (8)$$

and we now have an equivalent and *deterministic* formulation for Problem (2), where all the constraints are linear.

### B. Robust Formulation Defined by D-norm

*D*-norm approach [13] is another method to model the uncertainty set, and has advantages such as guarantee of feasibility independent of uncertainty distributions and flexibility in terms of tradeoff between robustness and performance.

Consider the  $j^{\text{th}}$  constraint  $\mathbf{a}_j^T \mathbf{x} \leq b_j$  in (2). Denote the set of all uncertain coefficients in  $\mathbf{a}_j$  as  $\mathcal{E}_j$ . The size of  $\mathcal{E}_j$  is  $|\mathcal{E}_j|$ , which might be smaller than the total number of coefficients  $N$  (i.e.,  $a_{ij}$  for some  $i$  might not have uncertainty). For each  $a_{ij} \in \mathcal{E}_j$ , assume the actual value falls into the range of  $[\bar{a}_{ij} - \hat{a}_{ij}, \bar{a}_{ij} + \hat{a}_{ij}]$ , in which  $\hat{a}_{ij}$  is a given error bound. Also choose a nonnegative integer  $\Gamma_i \leq |\mathcal{E}_j|$ . The definition of robustness associated with the *D*-norm formulation is to maintain feasibility if at most  $\Gamma_i$  out of all possible  $|\mathcal{E}_j|$  parameters are perturbed. Let's denote  $\mathcal{S}_i$  as the set of  $\Gamma_i$  uncertain coefficients. The above robustness definition can be characterized by the following protection function,

$$g_j(\Gamma_j, \mathbf{x}) = \max_{\mathcal{S}_j: \mathcal{S}_j \subseteq \mathcal{E}_j, |\mathcal{S}_j| = \Gamma_j} \sum_{i \in \mathcal{S}_j} \hat{a}_{ij} |x_i|. \quad (9)$$

If  $\Gamma_j = 0$ , then  $g_j(\Gamma_j, \mathbf{x}) = 0$  and the  $j^{\text{th}}$  constraint is reduced to the nominal constraint, i.e., no protection against uncertainty. If  $\Gamma_j = |\mathcal{E}_j|$ , then  $g_j(\Gamma_j, \mathbf{x}) = \sum_{i \in \mathcal{E}_j} \hat{a}_{ij} |x_i|$  and the  $j^{\text{th}}$  constraint becomes Soyster's worst-case formulation [13]. The tradeoff between robustness and performance can be obtained by adjusting  $\Gamma_j$ .

Note that the nonlinearity of  $g_j(\Gamma_j, \mathbf{x})$  is difficult to deal with in the constraint. We reformulate it into the following LP problem,

$$\max_{\{0 \leq s_{ij} \leq 1\} \forall i \in \mathcal{E}_j} \sum_{i \in \mathcal{E}_j} \hat{a}_{ij} |x_i| s_{ij}, \quad \text{s.t.} \quad \sum_{i \in \mathcal{E}_j} s_{ij} \leq \Gamma_j. \quad (10)$$

Taking the dual of Problem (10), we have

$$\min_{\{p_{ij} \geq 0\}_{\forall i \in \mathcal{E}_j, q_j \geq 0}} q_j \Gamma_j + \sum_{i \in \mathcal{E}_j} p_{ij}, \quad \text{s.t. } q_j + p_{ij} \geq \hat{a}_{ij} |x_i|, \forall i \in \mathcal{E}_j. \quad (11)$$

Similar to Section II-A, we can substitute (11) into the robust Problem (2) to obtain an equivalent formulation:

$$\begin{aligned} & \text{maximize } f_0(\mathbf{x}) \\ & \text{subject to } \sum_i \hat{a}_{ij} x_i + q_j \Gamma_j + \sum_{i \in \mathcal{E}_k} p_{ij} \leq b_j, \forall j, \\ & \quad q_j + p_{ij} \geq \hat{a}_{ij} y_i, \quad \forall i \in \mathcal{E}_k, \forall j, \\ & \quad -y_i \leq x_i \leq y_i, \forall i, \\ & \text{variables } \mathbf{x}, \mathbf{y} \succeq \mathbf{0}, \mathbf{p} \succeq \mathbf{0}, \mathbf{q} \succeq \mathbf{0}. \end{aligned} \quad (12)$$

The new problem only has linear constraints. We provide such an example in Section III.

### C. Robust Formulation Defined by Ellipsoid

Ellipsoid is commonly used to approximate complicated uncertainty sets based on statistical reasons [15] and to succinctly describe a set of discrete points in Euclidean geometry [14]. Here we consider the case where coefficient  $\mathbf{a}_j$  falls in an ellipsoid centered at the nominal  $\bar{\mathbf{a}}_j$ . Specifically,

$$\mathcal{A}_j = \{\bar{\mathbf{a}}_j + \Delta \mathbf{a}_j : \sum_i |\Delta a_{ij}|^2 \leq \epsilon_j^2\}. \quad (13)$$

By (4), the protection function is given by

$$g_j(\mathbf{x}) = \max \left\{ \sum_i \Delta a_{ij} x_i : \sum_i |\Delta a_{ij}|^2 \leq \epsilon_j^2 \right\}, \quad (14)$$

where the supremum is replaced by maximum because the uncertainty sets  $\mathcal{A}_j$  are compact. Denote by  $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$  the  $\ell_2$ -norm (or the Euclidean norm) of  $\mathbf{x}$ . By Cauchy-Schwartz inequality,

$$\sum_i \Delta a_{ij} x_i \leq \sqrt{\sum_i |\Delta a_{ij}|^2} \|\mathbf{x}\|_2 \leq \epsilon_j \|\mathbf{x}\|_2,$$

and the equality is attained by choosing

$$\Delta a_{ij} = \frac{x_i \epsilon_j}{\|\mathbf{x}\|_2}.$$

Therefore we conclude that

$$g_j(\mathbf{x}) = \epsilon_j \|\mathbf{x}\|_2. \quad (15)$$

Although the resulting constraint in Problem (2) is not readily decomposable using standard decomposition techniques, we will show in Part II of this paper that this leads to tractable formulations in some important applications (e.g., wireless power control) where users can obtain network information through local measurements without global message passing.

#### *D. Distributed Algorithm for Robust Optimization under Linear Constraints*

We now consider possible distributed algorithms to solve the general robust optimization problem with linear constraints in (2). Notice, however, the design of a truly distributed algorithm needs to take into account the setup of the practical system. In many cases, very similar mathematical model may represent totally different systems and requires different distributed algorithms. Our motivation here is not to design a general distributed algorithm which is guaranteed to work equally well for all applications. Instead, we try to show how to solve (2) by exploring its special structures in DRO, and the resulting algorithm may lead to distributed algorithm for each of the individual applications, such as those in Section III below and Part II of the paper.

***Nonlinear Gauss-Seidel and Jacobi Method:*** The Gauss-Seidel (GS) and Jacobi methods were originally proposed as efficient means to solve linear systems, and have been extended to tackle unconstrained as well as constrained optimization problems [7]. GS methods and Jacobi methods have a variety of applications including CDMA multiuser detection [16] and decoding of low-density parity-check code [17]. The general principle of both methods is to decompose a multidimensional optimization problem into multiple one-dimensional optimization problems, and solve these one-dimensional optimization problems in a successive or parallel manner.

Let  $x_i(k)$  denote the tentative value of the  $i^{th}$  variable obtained at the  $k^{th}$  iteration of the GS algorithm. Also Let  $X_i(x_1(k), \dots, x_{i-1}(k), x_{i+1}(k), \dots, x_N(k))$  be the feasible region of the  $i^{th}$

variable when the values of other variables are set as  $x_1(k), \dots, x_{i-1}(k), x_{i+1}(k), \dots, x_N(k)$ , i.e.,

$$X_i(x_1(k), \dots, x_{i-1}(k), x_{i+1}(k), \dots, x_N(k)) \triangleq \{x_i : \bar{\mathbf{a}}_j^T \tilde{\mathbf{x}} + g_j(\tilde{\mathbf{x}}) \leq b_j, \forall 1 \leq j \leq M\}, \quad (16)$$

where  $\tilde{\mathbf{x}} = [x_1(k), \dots, x_{i-1}(k), x_i, x_{i+1}(k), \dots, x_N(k)]$ .  $x_i(k+1)$  is found by the following GS iteration,

$$x_i(k+1) = \arg \max_{x_i \in X_i(x_1(k+1), \dots, x_{i-1}(k+1), x_{i+1}(k), \dots, x_N(k))} f_0(x_1(k+1), \dots, x_{i-1}(k+1), x_i, x_{i+1}(k), \dots, x_N(k)).$$

Likewise, in the Jacobi iteration,  $x_i(k+1)$  is found by

$$x_i(k+1) = \arg \max_{x_i \in X_i(x_1(k), \dots, x_{i-1}(k), x_{i+1}(k), \dots, x_N(k))} f_0(x_1(k), \dots, x_{i-1}(k), x_i, x_{i+1}(k), \dots, x_N(k)). \quad (17)$$

The difference between the GS method and the Jacobi method lies in the order of solving the one-dimensional problems. The GS method solves the one-dimensional problems sequentially, while the Jacobi method solve in parallel. They can also be combined together to form hybrid algorithms. We later show a distributed robust power control application, which is closely related to the nonlinear GS and Jacobi method.

**Cutting-plane Method:** we consider next the cutting-plane method to solve the robust optimization problem with linear constraints. The cutting-plane method has been used for solving the general nonlinear programming problems and mixed linear integer programming problems [18]. We assume the protection function here is convex and bounded. Also, we assume the feasible region of the robust optimization with linear constraints, i.e.,  $X$  is a convex and bounded set. At the  $k^{\text{th}}$  iteration of this method, we use the following function to approximate  $g_j(\mathbf{b}_j, \mathbf{x})$ ,

$$\bar{g}_j(\mathbf{b}_j, \mathbf{x}) = \max_{\ell} \tilde{\mathbf{a}}_{j\ell} \mathbf{x} + \tilde{b}_{j\ell}, \quad 1 \leq \ell \leq k \quad (18)$$

such that  $\forall \mathbf{x} \in X$ , we have  $\bar{g}_j(\mathbf{b}_j, \mathbf{x}) \geq g_j(\mathbf{b}_j, \mathbf{x})$ . The key is to appropriately choose  $\tilde{\mathbf{a}}_{j\ell}$  and  $\tilde{b}_{j\ell}$  for all  $\ell$ .

Substituting the protection function  $g_j(\mathbf{b}_j, \mathbf{x})$  with  $\bar{g}_j(\mathbf{b}_j, \mathbf{x})$ , we obtain the following approximating problem.

$$\text{maximize } f_0(\mathbf{x}), \quad (19)$$

subject to  $\bar{\mathbf{a}}_j^T \mathbf{x} + \hat{\mathbf{a}}_{j\ell} \mathbf{x} + \tilde{b}_{j\ell} \leq b_j, \forall 1 \leq j \leq M, 1 \leq \ell \leq k,$   
variables  $\mathbf{x}$ .

The feasible set of (19) is still a polytope, and therefore we assume it can be distributively solved. Based on the obtained solution, we can generate a new cutting-plane. The new cutting-plane, together with previously obtained constraints, can help to attain a better approximation of the protection function. Notice, however, how to efficiently find a new cutting-plane depends on which type of protection function we use. Details about how to find cutting-planes for different convex set can be found in [14].

Among the three aforementioned uncertainty sets, it is straightforward to generate cutting-planes for polyhedra and D-norm uncertainty set. For the ellipsoid uncertainty set, [19] has proposed an efficient approach to find a good polytope approximation for a given ellipsoid. A drawback of the cutting-plane approach is that it may result in a large number of linear constraints and bring about a lot of extra message passings. In practice, however, we can employ the efficient numerical algorithms such as the active set method to reduce the number of message passing. As an example, we will show a distributed robust rate control algorithm in the following section, which is inspired by the cutting-plane method.

### E. Definition of Robustness

In all previous discussions, we assume that the uncertainty set  $\mathcal{A}$  is either an accurate description or a conservative estimation of the uncertainty in practice. In other words, it is not possible to have parameters outside set  $\mathcal{A}$ . In this case, the definition of robustness is the *worst case robustness*, i.e., the solution of the robust optimization problem is always feasible. However, this approach might be too conservative. A more meaningful choice of robustness is the *chance-constrained robustness*, i.e., the probability of infeasibility (or outage) is upper bounded. We can flexibly adjust the chance-constrained robustness of the robust solution by solving the worst-case robust optimization problem over a properly selected subset of the exact uncertainty set. We will discuss such an example in details in Section III-D.

### III. ROBUST MULTIPATH RATE CONTROL

#### A. Nominal and Robust Formulations

Consider a wireline network where some links might fail due to reasons such as human mistakes, software bugs, hardware defects, or natural hazard. Network operators typically reserve some bandwidth for backup paths. When the primary paths fail, some or all of the traffic will be re-routed to the corresponding disjoint backup paths. Fast system recovery schemes are required to guarantee service availability in the presence of link failure. There are three key components for fast system recovery [20]: identifying a backup path disjoint from the primary path, computing network resource (such as bandwidth) in reservation prior to link failure, detecting the link failure in real-time and re-route the traffic. The first component has been investigated extensively in graph theory. The third component has been extensively studied in system research community. Here we consider the robust rate control and bandwidth reservation in the face of possible failure of primary path, which is related to the second component.

First consider the nominal problem with no link failures. Following similar notation as in Kelly's seminal work [11], we consider a network with  $S$  users,  $L$  links and  $T$  paths, indexed by  $s$ ,  $l$  and  $t$ , respectively. Each user is a unique flow from one source node to one destination node. There could be multiple users between the same source-destination node pair. The network is characterized by the  $L \times T$  path-availability 0 – 1 matrix

$$[\mathbf{D}]_{lt} = \begin{cases} d_{lt} = 1, & \text{if link } l \text{ is on path } t, \\ 0, & \text{otherwise.} \end{cases}$$

and  $T \times S$  primary-path-choice nonnegative matrix

$$[\mathbf{W}]_{ts} = \begin{cases} w_{ts}, & \text{if user } s \text{ uses path } t \text{ as the primary path,} \\ 0, & \text{otherwise.} \end{cases}$$

where  $w_{ts}$  indicates the percentage that user  $s$  allocates its rate to primary path  $t$ , and satisfies  $w_{ts} > 0$  and  $\sum_t w_{ts} = 1$ . Let  $\mathbf{x}$ ,  $\mathbf{c}$ , and  $\mathbf{y}$  denote source rates, link capacities, and aggregated path rates, respectively. The nominal multi-path rate control problem is

$$\text{maximize } \sum_s f_s(x_s) \tag{20}$$

$$\begin{aligned} & \text{subject to } \mathbf{D}\mathbf{y} \preceq \mathbf{c}, \quad \mathbf{W}\mathbf{x} \preceq \mathbf{y}, \\ & \text{variables } \mathbf{x} \succeq \mathbf{0}, \mathbf{y} \succeq \mathbf{0}, \end{aligned}$$

where  $f_s(x_s)$  is the utility of user  $s$ , which is increasing and strictly concave in  $x_s$ .

In order to guarantee the data transmission is robust against the link failure, each user also determines a backup path when it joins the network. The nonnegative backup path choice matrix is

$$[\mathbf{B}]_{ts} = \begin{cases} b_{ts}, & \text{if user } s \text{ uses path } t \text{ as the backup path,} \\ 0, & \text{otherwise.} \end{cases}$$

where  $b_{ts}$  indicates the maximum percentage that user  $s$  allocates its rate to path  $t$  and satisfies  $b_{ts} > 0$ . The actual rate allocation will be a random variable between 0 and  $b_{ts}$ , depending on whether the primary paths fail. We further assume that a path can only be used as either a primary path or a backup path for the same user by not both. The corresponding robust multi-path routing rate allocation problem is given by

$$\begin{aligned} & \text{maximize } \sum_s f_s(x_s) & (21) \\ & \text{subject to } \mathbf{D}\mathbf{y} \preceq \mathbf{c}, \\ & \sum_s w_{ts}x_s + g_t(\mathbf{b}_t, \mathbf{x}) \leq y_t, \quad \forall t. \\ & \text{variables } \mathbf{x} \succeq \mathbf{0}, \mathbf{y} \succeq \mathbf{0}. \end{aligned}$$

Here  $\sum_s w_{ts}x_s$  denotes the aggregate rate from users who utilize path  $t$  as their primary path, and  $g_t(\mathbf{b}_t, \mathbf{x})$  corresponds to the protection function for the traffic from users who use path  $t$  as their backup path, and  $\mathbf{b}_t$  is the  $t^{\text{th}}$  row of matrix  $\mathbf{B}$ . There are many ways of characterizing the protection function. Here we consider the choice of  $D$ -norm.

Let  $\mathcal{E}_t = \{s : b_{ts} > 0, \forall s\}$  denote the set of users who utilize path  $t$  as the backup path, and  $\mathcal{F}_{t,\Gamma_t}$  denote a subset of  $\mathcal{E}_t$  with size  $\Gamma_t$ , where  $0 \leq \Gamma_t \leq |\mathcal{E}_t|$  and controls the tradeoff between robustness and performance. Notice that there  $\Gamma_t$  is a parameter of problem (21). Then the protection function is

$$g_t(\mathbf{b}_t, \mathbf{x}) = \max_{\mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s, \forall t. \quad (22)$$

## B. Distributed Active Set Algorithms

Following the approach in Section II-B, we can convert the robust optimization problem into an equivalent problem with only linear constraints and solve it distributively by dual-based decompositions.

This approach, however, leads to a large amount of extra message passing (due to the new auxiliary variables and constraints) and is computationally expensive to calculate local projections. In this section, we propose a fast distributed algorithm based on a combination of active set method [21] and dual-based decomposition method.

We first show that the nonlinear constraints in Problem (21) can be replaced by a set of linear constraints:

**Proposition 2.** *For any path  $t$ , the constraint*

$$\sum_s w_{ts}x_s + g_t(\mathbf{b}_t, \mathbf{x}) \leq y_t, \quad (23)$$

*is equivalent to the following set of constraints*

$$\sum_s w_{ts}x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \leq y_t, \quad \forall \mathcal{F}_{t,\Gamma_t} \in \mathcal{E}_t. \quad (24)$$

*Proof:* From (22), we know  $\sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \leq g_t(\mathbf{b}_t, \mathbf{x})$  for all  $\mathcal{F}_{t,\Gamma_t}$ . If  $\mathbf{x}^*$  satisfies constraint (23), we have

$$\sum_{s \in P_t} w_{ts}x_s^* + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s^* \leq \sum_{s \in P_t} w_{ts}x_s^* + g_t(\mathbf{b}_t, \mathbf{x}^*) \leq y_t, \quad (25)$$

i.e., it also satisfies the set of constraints in (24). On the other hand, if

$$\sum_{s \in P_t} w_{ts}x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \leq y_t, \quad \forall \mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t, |\mathcal{F}_{t,\Gamma_t}| = \Gamma_t, \forall t, \quad (26)$$

we have

$$\begin{aligned} & \sum_{s \in P_t} w_{ts}x_s + g_t(\mathbf{b}_t, \mathbf{x}) \\ &= \max_{\forall \mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t, |\mathcal{F}_{t,\Gamma_t}| = \Gamma_t} \left( \sum_{s \in P_t} w_{ts}x_s^* + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \right) \leq y_t. \end{aligned} \quad (27)$$

Therefore these two constraints are equivalent. ■

Based on Proposition 2, we can convert robust optimization problem (21) into a problem with only linear constraints. However, the number of new linear constraints grows in the order of  $M\Gamma_t$ , where  $M$  is the number of linear constraints in the nominal problem. More importantly, the resulting new optimization problem is difficult to solve by the dual decomposition method in a distributed fashion. This motivates us to design an alternative sequential optimization algorithm.

Let  $\mathcal{H}_t = \{\mathcal{F}_{t,\Gamma_t} | \mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t\}$  denote the set of all subsets of  $\mathcal{E}_t$  with size  $\Gamma_t$ . The basic idea is to iteratively generate a set  $\bar{\mathcal{H}}_t \subseteq \mathcal{H}_t$ , and use the following set of constraints to approximate (24):

$$\sum_s w_{ts}x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \leq y_t, \quad \forall \mathcal{F}_{t,\Gamma_t} \in \bar{\mathcal{H}}_t. \quad (28)$$

This leads to a relaxation of Problem (21):

$$\begin{aligned} & \text{maximize} \quad \sum_s f_s(x_s) & (29) \\ & \text{subject to} \quad \mathbf{D}\mathbf{y} \preceq \mathbf{c}, \\ & \quad \quad \quad \sum_s w_{ts}x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}x_s \leq y_t, \quad \forall \mathcal{F}_{t,\Gamma_t} \in \bar{\mathcal{H}}_t, \quad \forall t., \\ & \quad \quad \quad \text{variables } \mathbf{x} \succeq \mathbf{0}, \mathbf{y} \succeq \mathbf{0}. \end{aligned}$$

Let  $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$  denote an optimal solution of (29) and  $(\mathbf{x}^*, \mathbf{y}^*)$  denote an optimal solution of (21). If  $\bar{\mathcal{H}}_t = \mathcal{H}_t$ , then we have  $\sum_s f_s(\bar{\mathbf{x}}_s) = \sum_s f_s(\mathbf{x}_s^*)$ . Even if  $\bar{\mathcal{H}}_t \subset \mathcal{H}_t$ , the two optimal objective values can still be the same as shown in the following theorem:

**Theorem 1.**  $\sum_s f_s(\bar{\mathbf{x}}_s) = \sum_s f_s(\mathbf{x}_s^*)$  if the following condition holds

$$g_t(\mathbf{b}_t, \bar{\mathbf{x}}) = \max_{\mathcal{F}_{t,\Gamma_t} \in \bar{\mathcal{H}}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts}\bar{x}_s, \quad \forall t. \quad (30)$$

*Proof:* Since  $\bar{\mathcal{H}}_t \subseteq \mathcal{H}_t$  and  $\bar{\mathbf{x}}$  is the optimal source rate allocation of the relaxed problem in (29), we have  $\sum_s f_s(\bar{\mathbf{x}}_s) \geq \sum_s f_s(\mathbf{x}_s^*)$ . However, the condition in (30)

implies that

$$\begin{aligned} & \sum_s w_{ts} \bar{x}_s + g_t(\mathbf{b}_t, \bar{\mathbf{x}}) \\ &= \sum_s w_{ts} \bar{x}_s + \max_{\mathcal{F}_{t,\Gamma_t} \in \bar{\mathcal{H}}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} \bar{x}_s \leq \bar{y}_t, \quad \forall t, \end{aligned} \quad (31)$$

hence  $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$  is a feasible solution of (21), and  $\sum_s f_s(\bar{x}_s) \leq \sum_s f_s(x_s^*)$ . Thus we have  $\sum_s f_s(\bar{x}_s) = \sum_s f_s(x_s^*)$ . ■

Next we develop a distributed algorithm (Algorithm 1) to solve Problem (29) for a fixed  $\bar{\mathcal{H}}_t$  for each  $t$ , which is suboptimal for solving Problem (21). We then design an optimal distributed algorithm (Algorithm 2) that achieves the optimal solution of Problem (21) by iteratively using Algorithm 1.

We first give an equivalent representation of Problem (29) to facilitate the presentation of our distributed algorithms. For each path  $t$ , we let  $\mathcal{F}_{t,\Gamma_t}(i)$  represent the  $i^{\text{th}}$  element in set  $\bar{\mathcal{H}}_t$ , and define a group of auxiliary variables  $\{y_{ti}, 1 \leq i \leq |\bar{\mathcal{H}}_t|\}$ .

**Proposition 3.** *Consider the case where link  $l$  is on path  $t$ , i.e.,  $d_{lt} = 1$ . Then, given  $l$  and  $t$ , the set of constraints,*

$$\begin{aligned} & \sum_{j:j \neq t} d_{lj} y_j + d_{lt} y_t \leq c_l, \\ & \sum_s w_{ts} x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} x_s \leq y_t, \quad \forall \mathcal{F}_{t,\Gamma_t} \in \bar{\mathcal{H}}_t, \end{aligned} \quad (32)$$

is equivalent to the following set of constraints

$$\begin{aligned} & \sum_{j:j \neq t} d_{lj} y_{ji} + d_{lt} y_{ti} \leq c_{lji}, c_{lji} = c_l, \quad 1 \leq i \leq |\bar{\mathcal{H}}_t|, \\ & \sum_s w_{ts} x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} x_s \leq y_{ti}, \quad 1 \leq i \leq |\bar{\mathcal{H}}_t|. \end{aligned} \quad (33)$$

With the auxiliary variables  $\{y_{ti}\}$  and  $\{c_{lji}\}$ , we can convert (29) into the following form,

$$\begin{aligned} & \text{maximize} \quad \sum_s f_s(x_s) \\ & \text{subject to} \quad \bar{D} \bar{\mathbf{y}} \preceq \bar{\mathbf{c}}, \end{aligned} \quad (34)$$

$$\sum_s w_{ts}x_s + \sum_{s \in \mathcal{F}_{t,\Gamma_t}(i)} b_{ts}x_s \leq y_{ti}, \quad 1 \leq i \leq |\bar{\mathcal{H}}_t|,$$

variables  $\mathbf{x} \succeq \mathbf{0}, \bar{\mathbf{y}} \succeq \mathbf{0}$ ,

where  $\bar{\mathbf{y}} = \{\{y_{ti}\}_{i=1}^{|\bar{\mathcal{H}}_t|}\}_{t=1}^T$ , and  $\bar{\mathbf{c}} = \{\{\{c_{lti}\}_{i=1}^{|\bar{\mathcal{H}}_t|}\}_t\}_l$  where link  $l$  is on path  $t$ .  $\bar{\mathbf{D}}$  is a 0-1 matrix representing the first group of constraints in (33).

By relaxing the constraints in Problem (34) using dual variables  $\boldsymbol{\lambda} = \{\{\{\lambda_{lti}\}_{i=1}^{|\bar{\mathcal{H}}_t|}\}_{t=1}^T\}_{l=1}^L$ ,  $\boldsymbol{\mu} = \{\{\{\mu_{ti}\}_{i=1}^{|\bar{\mathcal{H}}_t|}\}_{t=1}^T\}$ , we obtain the following Lagrangian,

$$\begin{aligned} Z(\boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{x}, \bar{\mathbf{y}}) &= \sum_s f_s(x_s) + \boldsymbol{\lambda}^T (\bar{\mathbf{c}} - \bar{\mathbf{D}}\bar{\mathbf{y}}) + \\ &\quad \sum_t \mu_{ti} \sum_{i=1}^{|\bar{\mathcal{H}}_t|} \left( y_{ti} - \sum_s w_{ts}x_s - \sum_{s \in \mathcal{F}_{t,\Gamma_t}(i)} b_{ts}x_s \right), \end{aligned}$$

and the dual function is

$$Z(\boldsymbol{\lambda}, \boldsymbol{\mu}) = \max_{\mathbf{x} \succeq \mathbf{0}, \bar{\mathbf{y}} \succeq \mathbf{0}} Z(\boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{x}, \bar{\mathbf{y}}). \quad (35)$$

The optimization over  $\mathbf{x}$  in (35) can be decomposed into one problem for each user  $s$ :

$$\max_{x_s \geq 0} \left( f_s(x_s) - \sum_t \left( \sum_{i=1}^{|\bar{\mathcal{H}}_t|} \mu_{ti} w_{ts} + \sum_i \sum_{s: s \in \mathcal{F}_t(i)} \mu_{ti} b_{ts} \right) x_s \right). \quad (36)$$

Due to the problem reformulation in Proposition 3, link  $l$  is now associated with a group of dual variables  $\lambda_{lti}$ . Likewise, path  $t$  is associated with a group of dual variables  $\mu_{ti}$ . Each user  $s$  determines its transmission rate  $x_s$  by considering prices from both its primary path and backup path.

The optimization over  $\bar{\mathbf{y}}$  in (35) leads to the following relationship between dual variables,

$$\boldsymbol{\mu} = \bar{\mathbf{D}}^T \boldsymbol{\lambda},$$

otherwise the dual function is unbounded.

The master dual problem is

$$\max_{\boldsymbol{\lambda} \succeq \mathbf{0}, \boldsymbol{\mu} \succeq \mathbf{0}} Z(\boldsymbol{\lambda}, \boldsymbol{\mu}), \quad (37)$$

which can be solved by the subgradient method. For each dual variable  $\lambda_{lti}$ , its subgradient can be calculated as

$$\zeta_{lti}(\lambda_{lti}) = c_l - \sum_s w_{ts}x_s - \sum_{s \in \mathcal{F}_t(i)} b_{ts}x_s, \quad (38)$$

and  $\boldsymbol{\mu} = \bar{\mathbf{D}}^T \boldsymbol{\lambda}$ . The value of  $\lambda_{lti}$  will be updated using the subgradient information correspondingly. The complete algorithm is given as in Algorithm 1.

**Algorithm 1.** (*Suboptimal Distributed Algorithm*)

- 1) Set time  $k = 0$ ,  $\boldsymbol{\lambda}(0) = \mathbf{0}$ , and  $\boldsymbol{\mu}(0) = \mathbf{0}$ .
- 2) Let  $k = k + 1$ .
- 3) Each user  $s$  determines  $x_s(k)$  by solving Problem (36).
- 4) Each user  $s$  passes its tentative rate  $x_s(k)$  to each link associated with this user.
- 5) Each link  $l$  calculates the subgradients  $\boldsymbol{\zeta}_l(\boldsymbol{\lambda}_l(k)) = \{\zeta_{lti}(\lambda_{lti}(k)), \forall t, i\}$  as in (38).
- 6) If  $|\boldsymbol{\zeta}(\boldsymbol{\lambda})| \leq \epsilon$ , stop. Otherwise, each link  $l$  updates the dual variables

$$\boldsymbol{\lambda}_l(k+1) = \max\{\boldsymbol{\lambda}_l(k) + \theta(k)\boldsymbol{\zeta}_l(\boldsymbol{\lambda}_l(k)), 0\}.$$

- 7) Each user  $s$  using path  $t$  as primary or backup path calculates the associated dual prices  $\mu_{ti}(k+1) = \sum_l d_{lt}\lambda_{lti}(k+1)$ ,  $1 \leq i \leq |\bar{\mathcal{H}}_t|$ , by passing messages over path  $t$  from the destination to source.

Here  $\theta(k)$  is the step-size at time  $k$  and  $\epsilon$  is the stopping criterion. Building on Algorithm 1, we propose the optimal distributed algorithm in Algorithm 2 to find the optimal solution of Problem (21).

**Algorithm 2.** (*Optimal Distributed Algorithm*)

- 1) Each path  $t$  randomly generates a set  $\mathcal{F}_{t,\Gamma_t}$  and let  $\bar{\mathcal{H}}_t = \{\mathcal{F}_{t,\Gamma_t}\}$ .
- 2) Each path  $t$  passes  $\bar{\mathcal{H}}_t$  to every link associated with it.
- 3) Run Algorithm 1 to obtain a tentative result  $\mathbf{x}^*$ .
- 4) Each user  $s$  passes the tentative data rate  $x_s^*$  to every path associated with this user.
- 5) Rank  $\{b_{ts}x_s^*\}_{s \in \mathcal{E}_t}$  in descending order for path  $t$  and take the  $\Gamma_t$  biggest items to obtain a new set  $\mathcal{F}_{t,\Gamma_t}$ .

- 6) For path  $t$ , if every new generated set  $\mathcal{F}_{t,\Gamma_t}$  is already contained in the corresponding set  $\bar{H}_t$ , then the stopping criterion stated in Theorem 1 is satisfied, stop.
- 7) Otherwise, path  $t$  passes the new generated set  $\mathcal{F}_{t,\Gamma_t}$  to every link associated with this path. Every link in the path  $t$  adds  $\mathcal{F}_{t,\Gamma_t}$  into  $\bar{H}_t$ , and go to step 3.

Algorithm 2 iteratively generates a group of relaxed problems to approximate the original problem (21), and eventually converges to optimal solution. Note in the worst case we may need to generate all  $\mathcal{F}_{t,\Gamma_t} \in H_t$ . In practice, however, only a small portion of constrains are required to achieve the optimal solution.

### C. Numerical Results

Here we consider a simple network model with three nodes, 13 links and 13 paths, as shown in Fig. 1. Paths 1 – 12 are single link paths, and use links 1 – 12, respectively. Path 13 consists of links 12 and 13. The first 11 paths are used as primary paths by 11 users in the network. Path 12 is used as the backup path by users 1 – 8, and path 13 is used as the backup path by user 9 – 11. Each user  $s$  has a logarithmic utility function  $\log(x_s)$ , where unit of  $x_s$  is kbps. The capacity of each link is fixed at 1 Mbps.

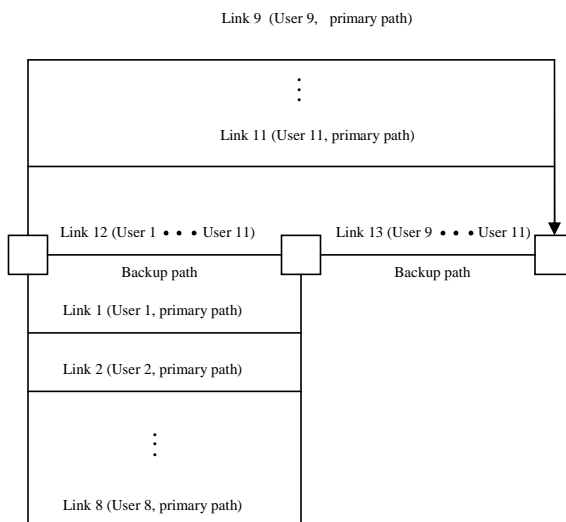


Fig. 1. Network Topology.

Figure 2 shows the convergence behavior of the proposed distributed optimal algorithm 1.

Here  $\Gamma_{12} = \Gamma_{13} = 3$ . It is seen the distributed method can converge to the optimal solution.

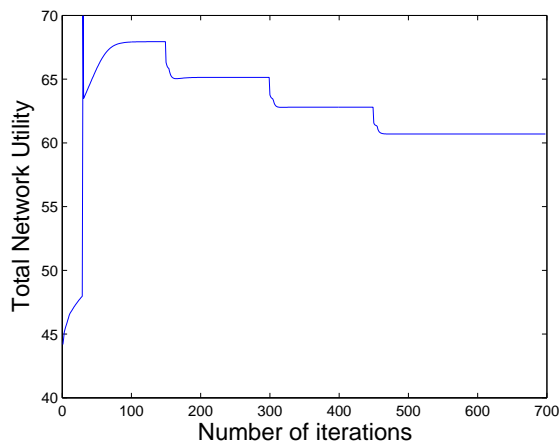


Fig. 2. Convergence of the suboptimal algorithm 1

#### D. Performance-Robustness Tradeoff

Besides the worst case robustness, we can also study the performance-robustness tradeoff by using *outage probability* as the definition of robustness. Let  $\{\mathbf{x}^*, \mathbf{y}^*\}$  denote the optimal solution to (21). An outage of the path  $t$  is defined as the occurrence of link failures which makes the constraint on the  $t^{\text{th}}$  path is violated, i.e.,  $\sum_{s \in P_t} w_{ts} x_s^* + \sum_{s \in \mathcal{E}_t} \bar{b}_{ts} x_s^* > y_t^*$ . Notice here  $\bar{b}_{ts}$  denotes the actual percentage of rate that user  $s$  allocates to the  $t^{\text{th}}$  path. Here we assume  $\bar{b}_{ts}$  is a Bernoulli random number which takes value  $b_{ts}$  with probability  $P_{ts}$  and 0 with probability  $1 - P_{ts}$ , i.e., if the primary path fails  $\bar{b}_{ts}$  equals to  $b_{ts}$ , otherwise  $\bar{b}_{ts}$  is 0. The outage can be measured by a probability function  $P_o(t)$  such that  $P_o(t) = Pr(\sum_{s \in P_t} w_{ts} x_s^* + \sum_{s \in \mathcal{E}_t} \bar{b}_{ts} x_s^* > y_t^*)$ . In addition, we use  $P_t$  to denote the upper bound to the probability of failure of the primary paths in  $\mathcal{E}_t$ , i.e.,  $P_t \geq P_{ts}, \forall s \in \mathcal{E}_t$ . We further assume failures of primary paths are independent of each other.

Clearly, the proposed rate control scheme becomes increasingly robust against link failures if we enforce stronger protections, i.e., increasing  $\Gamma_t$ . On the other hand, a larger  $\Gamma_t$  needs to reserve more bandwidth, and consequently reduces the maximum achievable rate. Various uncertainty sets give rise to different tradeoffs between the achievable rate and outage probability. In the sequel, we will show that the proposed approach can attain an excellent tradeoff between rate

and robustness. If less than  $\Gamma_t$  primary paths fail to work, the obtained rate control scheme remains feasible *deterministically*. Furthermore, the rate control scheme will be feasible with *high probability* if a proper  $\Gamma_t$  is selected, as shown in Theorem 2.

**Theorem 2.** Let  $\{\mathbf{x}^*, \mathbf{y}^*\}$  denote the optimal solution to (21). Recall  $\mathcal{E}_t$  denote the set of users who utilize path  $t$  as the backup path, and  $P_o(t)$  is the outage probability that the rate control on  $t^{\text{th}}$  path is violated, i.e.,  $\Pr(\sum_{s \in P_t} w_{ts} x_s^* + \sum_{s \in \mathcal{E}_t} \bar{b}_{ts} x_s^* > y_t^*)$ . Then,

- 1)  $P_o(t) = 0$  if  $|\bar{t}| \leq \Gamma_t$ , where  $|\bar{t}|$  is the number of users in  $\mathcal{E}_t$  whose primary paths fail.
- 2)  $P_o(t) \leq \sum_{k=\Gamma_t+1}^{|\mathcal{E}_t|} \binom{|\mathcal{E}_t|}{k} (P_t)^k (1 - P_t)^{|\mathcal{E}_t|-k}$ .
- 3)  $P_o(t) \leq -2 \frac{(\Gamma_t+1-|\mathcal{E}_t|P_t)^2}{|\mathcal{E}_t|}$ , if  $\frac{\Gamma_t+1}{|\mathcal{E}_t|} \geq P_t$ . In addition, let  $f_t \triangleq \frac{\Gamma_t}{|\mathcal{E}_t|}$ . If  $f_t > P_t$ , we have  $P_o(t) \leq e^{-D_v(f_t||P_t)|\mathcal{E}_t|}$  where  $D_v(f_t||P_t) = f_t \log\left(\frac{f_t}{P_t}\right) + (1-f_t) \log\left(\frac{1-f_t}{1-P_t}\right)$  is the Kullback-Leibler (KL) divergence for two Bernoulli random variables.

*Proof:* Since  $|\bar{t}| \leq \Gamma_t$ , we have

$$\sum_s w_{ts} x_s^* + \sum_{s \in \mathcal{F}_{t,|\bar{t}|}, |\mathcal{F}_{t,|\bar{t}}|=|\bar{t}|, \mathcal{F}_{t,|\bar{t}} \subseteq \mathcal{E}_t} \bar{b}_{ts} x_s^* \quad (39)$$

$$\leq \sum_s w_{ts} x_s^* + \max_{\mathcal{F}_{t,\Gamma_t}} \left( \sum_{s \in \mathcal{F}_{t,\Gamma_t}, \mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t, |\mathcal{F}_{t,\Gamma_t}|=\Gamma_t} \bar{b}_{ts} x_s^* \right) \quad (40)$$

$$\leq y_t^* \quad (41)$$

It follows that  $P_o(t) = 0$ . ■

Because the failures of primary paths are independent of each other, the total number of failed paths follows Binomial distribution. In addition, the failure probability of each single primary path is upper bounded by  $P_t$ . It follows that

$$P_o(t) \leq \sum_{k=\Gamma_t+1}^{|\mathcal{E}_t|} \binom{|\mathcal{E}_t|}{k} (P_t)^k (1 - P_t)^{|\mathcal{E}_t|-k}. \quad (42)$$

This bound can be difficult to compute when  $|\mathcal{E}_t|$  is large. We can further upperbound the outage probability as follows.

$$P_o(t) \leq \sum_{k=\Gamma_t+1}^{|\mathcal{E}_t|} \binom{|\mathcal{E}_t|}{k} (P_t)^k (1 - P_t)^{|\mathcal{E}_t|-k}, \quad (43)$$

$$= \sum_{k'=0}^{|\mathcal{E}_t|-\Gamma_t-1} \binom{|\mathcal{E}_t|}{|\mathcal{E}_t|-k'} (P_t)^{|\mathcal{E}_t|-k'} (1-P_t)^{k'}, \quad (44)$$

$$= \sum_{k'=0}^{|\mathcal{E}_t|-\Gamma_t-1} \binom{|\mathcal{E}_t|}{k'} (P_t)^{|\mathcal{E}_t|-k'} (1-P_t)^{k'}, \quad (45)$$

$$\leq e^{-2 \frac{[|\mathcal{E}_t|(1-P_t)-(\mathcal{E}_t-\Gamma_t-1)]^2}{|\mathcal{E}_t|}}, \quad (46)$$

$$= e^{-2 \frac{(\Gamma_t+1-|\mathcal{E}_t|P_t)^2}{|\mathcal{E}_t|}}, \quad (47)$$

provided that  $\frac{\Gamma_t+1}{|\mathcal{E}_t|} \geq P_t$ . (44) is obtained by letting  $k' \triangleq |\mathcal{E}_t| - k$ . (45) is due to the fact that  $\binom{|\mathcal{E}_t|}{|\mathcal{E}_t|-k'}$  equals to  $\binom{|\mathcal{E}_t|}{k'}$ . (46) follows from the Hoeffding inequality.

Let  $f_t \triangleq \frac{\Gamma_t}{|\mathcal{E}_t|}$ . Assume  $f_t > P_t$ , it follows from the Chernoff-Hoeffding theorem [22] that,

$$P_o(t) \leq \sum_{k=\Gamma_t+1}^{|\mathcal{E}_t|} \binom{|\mathcal{E}_t|}{k} (P_t)^k (1-P_t)^{|\mathcal{E}_t|-k}, \quad (48)$$

$$\leq \left(\frac{P_t}{f_t}\right)^{f_t} \left(\frac{1-P_t}{1-f_t}\right)^{1-f_t}, \quad (49)$$

$$\leq e^{-D_v(f_t||P_t)|\mathcal{E}_t|}. \quad (50)$$

Here  $D_v(f_t||P_t) = f_t \log\left(\frac{f_t}{P_t}\right) + (1-f_t) \log\left(\frac{1-f_t}{1-P_t}\right)$  denotes the Kullback-Leibler (KL) divergence for two Bernoulli random variables.

Figure 3 shows the tradeoff between robustness and performance. The performance is measured by the total network utility  $\sum_s \log(x_s)$ , and the robustness level is measured by the number of failures that is guaranteed to be protected on path 12, i.e.,  $\Gamma_{12}$ . The value of  $\Gamma_{13}$  is fixed at 3 in this example. As we see, the performance decreases as the robustness ( $\Gamma_{12}$ ) increases. Also the centralized algorithm and distributed Algorithm 2 achieve the same performance.

The outage probability of path 12 and its upper bounds are illustrated in Figure 4. Here  $P_t$  is set as 0.1, i.e., the probability of failures of users in  $\mathcal{E}_{12}$  is upper bounded by 0.1. As evident from the figure, the outage probability decreases *exponentially* with the increase of  $\Gamma_{12}$ . The Chernoff and Hoeffding bounds can be used as efficient means to estimate the outage probability in case the exact outage probability is difficult to calculate.

Fig. 5 illustrates the probabilistic tradeoff between robustness and total network utility. It is seen that by using the D-norm uncertainty set, the outage probability of a path can be reduced in an exponential rate by slightly decreasing the total network utility. In practice, we can select

a proper  $\Gamma_{12}$  to strike a balance between the robustness (outage probability) and the total system throughput.

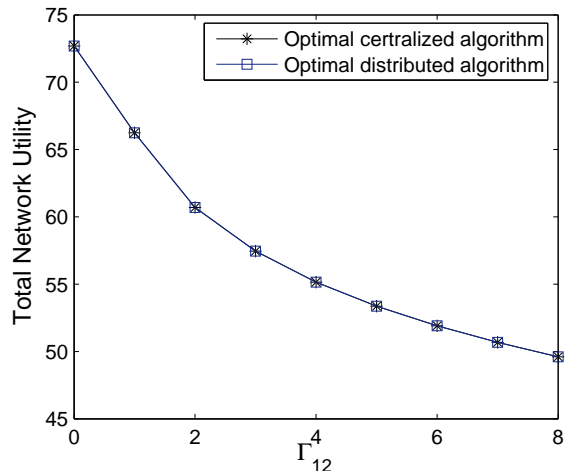


Fig. 3. Deterministic robustness-rate tradeoff under D-norm uncertainty set

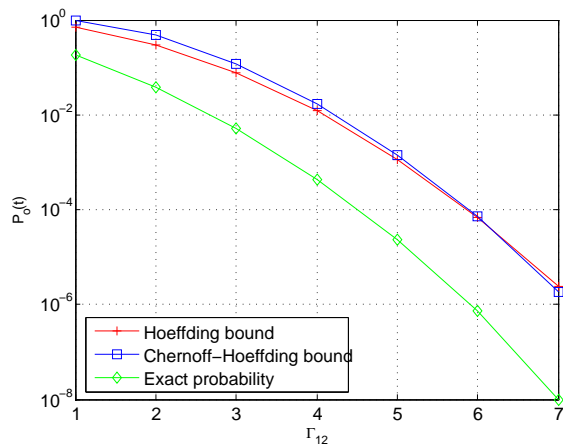


Fig. 4. Outage probability and upper bounds for different  $\Gamma_{12}$

### E. Robustness-Distributiveness Tradeoff

We next consider the tradeoff between the robustness and distributiveness of the proposed rate control algorithm. Here the robustness of the  $t^{\text{th}}$  path is quantified by the parameter  $\Gamma_t$ , which can control the outage probability  $P_o(t)$ . We use the total number of message passing required

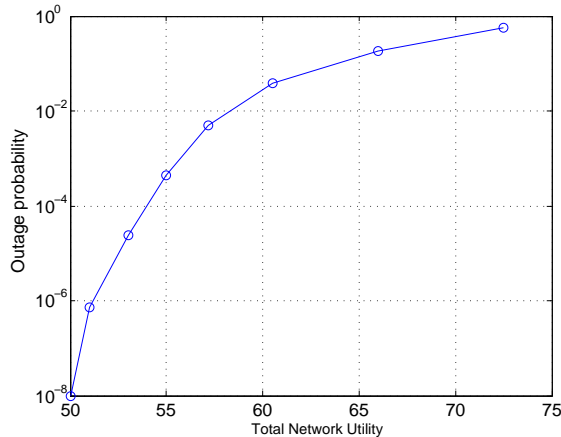


Fig. 5. Probabilistic robustness-rate tradeoff under D-norm uncertainty set

at one iteration in Algorithm 1 to measure the distributiveness. At each iteration of Algorithm 1, every link will collect tentative decision rate  $\mathbf{x}^*$  from sources (Step 4) and perform the subgradient projection as in (38). Then every path passes aggregate dual prices ,i.e.,  $\mu_{ti}(k+1) = \sum_l d_{lt} \lambda_{lti}(k+1)$ ,  $1 \leq i \leq |\bar{\mathcal{H}}_t|$ , from the destination back to the source (Step 7). The number of message passing required at Step 4 is *independent* of the parameter  $\Gamma_t$ , and the number of message passing for Step 7 increases only linearly with  $|\bar{\mathcal{H}}_t|$ . So the total number of message passing during one iteration for one user is  $O(1 + |\bar{\mathcal{H}}_t|)$ .

Figure 6 shows the robustness-distributiveness tradeoff of the distributed robust rate control algorithm. The upper bound to the number of message passing is derived from the fact that  $|\bar{\mathcal{H}}_t|$  is upper bounded by  $\binom{|\mathcal{E}_t|}{\Gamma_t}$ . We also calculate the actual number of message passing at the final iteration of Algorithm 2 through simulation studies. The actual curve presents a better tradeoff. These curves quantify the intuition that the number of required message passing is larger when  $\Gamma_t$  is around  $\frac{|\mathcal{E}_t|}{2}$  and becomes small when  $\Gamma_t$  approaches 0 or  $|\mathcal{E}_t|$ .

#### F. Rate-Distributiveness Tradeoff with Guaranteed Outage Bound

We conclude this section with a stronger result than generally obtainable in DRO: due to the special property of the D-norm protection function in rate control, for a given path  $t$ , we can tradeoff between the total network utility and distributiveness without loss of robustness. This tradeoff is achieved by choosing different linear constraints to approximate the D-norm uncertainty set.

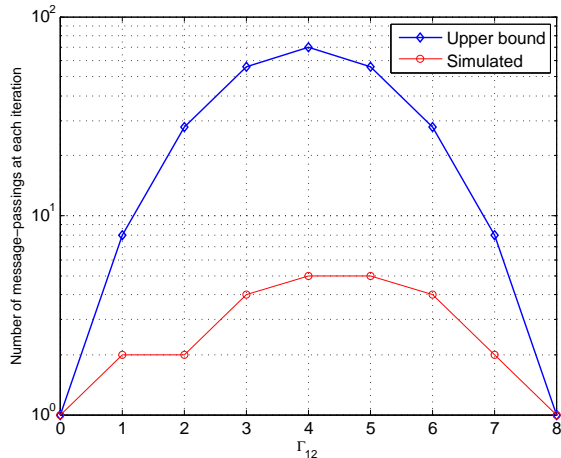


Fig. 6. Robustness-distributiveness tradeoff of the distributed robust rate control algorithm

Recall the D-norm protection function is  $g_t(\mathbf{b}_t, \mathbf{x}) = \max_{\mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} x_s, \forall t$ . Further, it has been seen in the preceding section that the number of required message passing at each step is proportional to  $\binom{|\mathcal{E}_t|}{\Gamma_t}$ , which equals to the number of linear constraints required to fully characterize  $g_t(\mathbf{b}_t, \mathbf{x})$ . The motivation here is to find a new protection function with larger object function value than that of  $g_t(\mathbf{b}_t, \mathbf{x})$  and can be represented by fewer linear constraints.

Consider, for example, that we want to protect the rate control scheme against single-link failure for the path  $t$ , i.e.,  $\Gamma_t = 1$ . We assume  $\mathcal{E}_t = \{e_t(1), e_t(2), e_t(3), e_t(4), e_t(5), e_t(6)\}$ . The protection function is chosen to be

$$g_t(\mathbf{b}_t, \mathbf{x}) = \max_{\mathcal{F}_{t,1} \subseteq \mathcal{E}_t} \sum_{s \in \mathcal{F}_{t,1}, |\mathcal{F}_{t,1}|=1} b_{ts} x_s. \quad (51)$$

Since  $|\mathcal{E}_t| = 6$ , this protection function can be equivalently represented by six linear constraints. Let  $\bar{\mathcal{E}}_t = \{[e_t(1), e_t(2)], [e_t(3), e_t(4)], [e_t(5), e_t(6)]\}$ . We can use the following protection function

$$\bar{g}_t(\mathbf{b}_t, \mathbf{x}) = \max_{\bar{\mathcal{F}}_t \in \bar{\mathcal{E}}_t} \sum_{s \in \bar{\mathcal{F}}_t} b_{ts} x_s, \forall t. \quad (52)$$

For any  $\mathcal{F}_{t,1}$ , we are able to find a corresponding set  $\bar{\mathcal{F}}_t$  such that  $\mathcal{F}_{t,1} \subseteq \bar{\mathcal{F}}_t$ . Further,  $\bar{\mathcal{E}}_t$  contains only three sets i.e.,  $|\bar{\mathcal{E}}_t| = 3$ . We can therefore reduce the number of required message passings at each iteration of Algorithm 1 by replacing the protection function  $g_t(\mathbf{b}_t, \mathbf{x})$  with  $\bar{g}_t(\mathbf{b}_t, \mathbf{x})$ . We can further obtain the following general result. Recall the D-norm protection function is given

by

$$g_t(\mathbf{b}_t, \mathbf{x}) = \max_{\mathcal{F}_{t,\Gamma_t}, |\mathcal{F}_{t,\Gamma_t}=\Gamma_t|, \mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} x_s, \forall t, \quad (53)$$

where  $\mathcal{E}_t = \{s : b_{ts} > 0, \forall s\}$  is the set of users who utilize path  $t$  as the backup path.  $\mathcal{F}_{t,\Gamma_t}$  denotes a subset of  $\mathcal{E}_t$  with size  $\Gamma_t$ . Let  $\bar{g}_t(\mathbf{b}_t, \mathbf{x})$  denote another protection function such that

$$\bar{g}_t(\mathbf{b}_t, \mathbf{x}) = \max_{\bar{\mathcal{F}}_t \in \bar{\mathcal{E}}_t} \sum_{s \in \bar{\mathcal{F}}_t} b_{ts} x_s, \forall t. \quad (54)$$

**Theorem 3.** *Suppose  $\bar{\mathcal{E}}_t$  is properly designed such that for any  $\mathcal{F}_{t,\Gamma_t}$ , we can always find  $\bar{\mathcal{F}}_t \in \bar{\mathcal{E}}_t$  which satisfies  $\mathcal{F}_{t,\Gamma_t} \subseteq \bar{\mathcal{F}}_t$ . The outage probability of path  $t$  by using the protection function  $\bar{g}_t(\mathbf{b}_t, \mathbf{x})$  is no larger than that by using the D-norm protection function.*

*Proof:* Assume  $\{\bar{\mathbf{x}}, \bar{\mathbf{y}}\}$  is the optimal solution to the robust rate control problem where the protection function is set as  $\bar{g}_t(\mathbf{b}_t, \mathbf{x})$ . For any  $\mathcal{F}_{t,\Gamma_t}$ , there exist  $\bar{\mathcal{F}}_t$  such that  $\mathcal{F}_{t,\Gamma_t} \subseteq \bar{\mathcal{F}}_t$ , it follows that

$$\begin{aligned} g_t(\mathbf{b}_t, \bar{\mathbf{x}}) &= \max_{\mathcal{F}_{t,\Gamma_t} \subseteq \mathcal{E}_t} \sum_{s \in \mathcal{F}_{t,\Gamma_t}} b_{ts} \bar{x}_s \\ &\leq \max_{\bar{\mathcal{F}}_t \subseteq \bar{\mathcal{E}}_t} \sum_{s \in \bar{\mathcal{F}}_t} b_{ts} \bar{x}_s = \bar{g}_t(\mathbf{b}_t, \bar{\mathbf{x}}). \end{aligned} \quad (55)$$

Consequently, we have  $\sum_s w_{ts} \bar{x}_s + \bar{g}_t(\mathbf{b}_t, \bar{\mathbf{x}}) \leq \sum_s w_{ts} \bar{x}_s + g_t(\mathbf{b}_t, \bar{\mathbf{x}}) \leq \bar{y}_t$ , which means  $\{\bar{\mathbf{x}}, \bar{\mathbf{y}}\}$  is a feasible solution to the robust power control problem with D-norm protection function and directly prove the claim. ■

The tradeoff between the total network utility and the robustness is shown in Figure 7. Here single-path protection is assumed, i.e.,  $\Gamma_{12} = 1$ . To make sure the rate control scheme is robust against single-path failure we can choose the D-norm protection function with  $\Gamma_t = 1$ . However, such a protection function brings about eight additional linear constraints and requires considerable message passing. To reduce the number of message passing, we can use alternative protection function. Here we consider three other protection functions satisfying conditions given in Theorem 3. Since  $|\mathcal{E}_{12}| = 8$ . The first protection function is to let  $|\bar{\mathcal{F}}_{12}| = 2$  and  $\bar{\mathcal{E}}_t = \{[e_t(1), e_t(2)], [e_t(3), e_t(4)], [e_t(5), e_t(6)], [e_t(5), e_t(6)]\}$ . Such a protection function can be represented by four linear constraints. We can obtain the other two protection functions in a

similar manner.

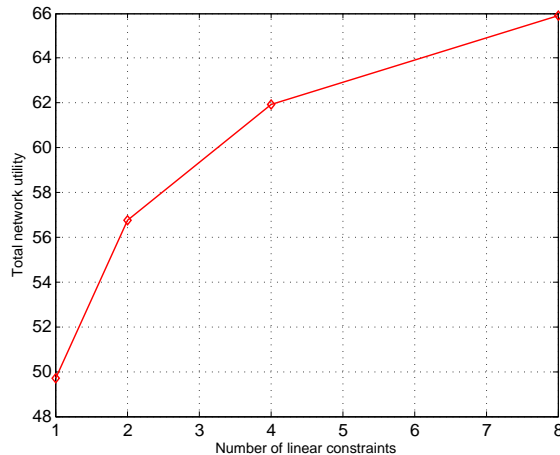


Fig. 7. Rate-distributiveness tradeoff with guaranteed outage bound

#### IV. CONCLUSIONS

Making optimization models of communication network design robust and distributed at the same time is an under-explored area. This paper initiates the study of DRO through robust formulations that preserve a large degree of distributiveness of solution algorithms. We first describe several models for describing parameter uncertainty sets that can lead to distributed solutions for linearly constrained nominal problems. These models include general polyhedron,  $D$ -norm, and ellipsoid. We then apply these models in the example of distributed rate control. The tradeoff between robustness (i.e., the maximum of link failures allowed), performance (total network utility), and distributiveness (i.e., the amount of message passing needed) is demonstrated. In Part II of this two-part paper, extensive applications of DRO methodology to wireless power control will be presented.

The study of distributed robust optimization in general remains wide open, with many challenging issues and possible applications where robustness to uncertainty is as important as optimality in the nominal model. Extensions of the basic results in this paper include nonlinear constraint sets in the nominal problem or uncertainties in other parts of the nominal formulations.

#### REFERENCES

- [1] A. Ben-Tal and A. Nemirovski, “Robust solutions to uncertain programs,” *Operations Research Letter*, vol. 25, pp. 1–13, Feb. 1999.

- [2] ———, “Selected topics in robust convex optimization,” *Mathematical Programming*, vol. 1, no. 1, pp. 125–158, July 2007.
- [3] A. Nemirovski, “On tractable approximations of randomly perturbed convex constraints,” *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 2419–2422, 2003.
- [4] L. El-Ghaoui and H. Lebret, “Robust solutions to least-square problems to uncertain data matrices,” *SIAM J. Matrix Anal. Appl.*, vol. 18, pp. 1035–1064, 1997.
- [5] L. El-Ghaoui, F. Oustry, and H. Lebret, “Robust solutions to uncertain semidefinite programs,” *SIAM J. Optim.*, vol. 9, pp. 33–52, 1998.
- [6] A. Ben-Tal and A. Nemirovski, “Robust solutions to linear programming problems contaminated with uncertain data,” *Mathematical Programming*, vol. 88, pp. 411–424, 2000.
- [7] D. P. Bertsekas and J. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Englewood Cliffs, NJ: Prentice Hall, 1999.
- [8] G. Foschini and Z. Miljanic, “A simple distributed autonomous power control algorithm and its convergence,” *Vehicular Technology, IEEE Transactions on*, vol. 42, no. 4, pp. 641–646, Nov. 1993.
- [9] J. Huang, R. Berry, and M. L. Honig, “Distributed interference compensation in wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 5, pp. 1074–1084, May 2006.
- [10] J. W. Lee, M. Chiang, and R. A. Calderbank, “Utility-optimal random-access control,” *Wireless Communications, IEEE Transactions on*, vol. 6, no. 7, pp. 2741–2751, 2007.
- [11] F. Kelly, A. Maulloo, and D. Tan, “Rate control for communication networks: shadow prices, proportional fairness and stability,” *Journal of the Operational Research Society*, vol. 49, no. 3, pp. 237–252, 1998.
- [12] K. Yang and X. Wang, “Cross-layer network planning for multi-radio cognitive wireless networks,” *Communications, IEEE Transactions on*, vol. 7, no. 11, pp. 1363–1373, November 2008.
- [13] B. Dimitris and M. Sim, “The price of robustness,” *Operations Research*, vol. 52, no. 1, pp. 35–53, Feb. 2004.
- [14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
- [15] A. Ben-Tal and A. Nemirovski, “Robust solutions to uncertain linear problems,” Research Report 6/95, Optimization Laboratory, Faculty of Industrial Engineering and Management, Technion, Tech. Rep., 1995.
- [16] A. Yener, R. Yates, and S. Ulukus, “CDMA multiuser detection: A nonlinear programming approach,” *IEEE Transactions on Communications*, vol. 50, no. 6, pp. 1016–1024, June 2002.
- [17] K. Yang, J. Feldman, and X. Wang, “Non-linear programming approaches to decoding low-density parity-check codes,” *IEEE J. Select. Areas Commun.*, vol. 24, no. 6, pp. 1603–1613, Aug. 2006.
- [18] D. Bertsimas and J. N. Tsitsiklis, *Linear Optimization*, 1st ed. Belmont, MA: Athena Scientific, 1997.
- [19] A. N. A. Ben-Tal, “On polyhedral approximations of the second-order cone,” *Mathematics of Operations Research*, vol. 26, no. 2, pp. 193–205, May 2001.
- [20] D. Xu, Y. Li, M. Chiang, and A. R. Calderbank, “Optimal provision of elastic service availability,” *Proc. IEEE INFOCOM*, 2007.
- [21] S. Leyffer, “The return of the active set method,” *Oberwolfach Reports*, vol. 2, no. 1, pp. 107–109, January 2005.
- [22] W. Hoeffding, “Probability inequalities for sums of bounded random variables.”