Data-driven Decision Making with Probabilistic Guarantees (Part 1):
Theory and Applications of Chance-constrained Optimization

Xinbo Geng, Le Xie
Texas A&M University, College Station, TX, USA.

Abstract
Uncertainties from deepening penetration of renewable energy resources have posed critical challenges to secure and reliable operations of the future electric grid. Among many approaches proposed for decision making in uncertain environments, this paper focuses on chance-constrained optimization, which provides explicit probabilistic guarantees on the feasibility of optimal solutions. Although many methods have been proposed to solve chance-constrained optimization problems, there is a lack of comprehensive review and comparative analysis of the proposed methods. We first review three categories of existing methods to chance-constrained optimization: (1) scenario approach; (2) sample average approximation; and (3) robust optimization based methods. Data-driven methods, which are not constrained by any particular distributions of the underlying uncertainties, are of particular interest. We then provide a comprehensive review on the applications of chance-constrained optimization in power systems. Finally, this paper provides a critical comparison of existing methods based on numerical simulations, which are conducted on standard power system test cases.

Keywords: data-driven, power system, chance constraint, probabilistic constraint, stochastic programming, robust optimization, chance-constrained optimization.

1. Introduction
Real-time decision making in the presence of uncertainties is a classical problem that arises in many contexts. In the context of electric energy systems, a pivotal challenge is how to operate a power grid with an increasing amount of supply and demand uncertainties. The unique characteristics of such operational problem include (1) the underlying distribution of uncertainties is largely unknown (e.g. the forecast error of demand response programs); (2) decisions have to be made in a timely manner (e.g. a dispatch order needs to be given by 5 minutes prior to the real-time); and (3) there is a strong desire to know the risk that the system is exposed to after a decision is made (e.g. the risk of violating transmission constraints after the clearing of real-time market). To meet the challenges of the aforementioned characteristics, a class of optimization problems named “chance-constrained optimization” has received increasing attention in both operations research and practical engineering communities.

The objective of this article is to provide a comprehensive and up-to-date review of mathematical formulations, computational algorithms, and engineering implications of chance-constrained optimization in the context of electric power systems. In particular, this paper focuses on the data-driven approaches to solve chance-constrained optimization without knowing or making specific assumptions on the underlying distribution of the uncertainties.

1.1. An Overview of Chance-constrained Optimization
Chance-constrained optimization (CCO) is an important tool for decision making in uncertain environments. Since its birth in 1950s, CCO has found many successful applications in various fields, e.g. economics (Yaari, 1965), control theory (Calafiore et al., 2006), chemical process (Sahinidis, 2004; Henrion et al., 2001), water management (Dupaev et al., 1991) and recently in machine learning (Sra et al., 2012; Xu et al., 2009; Caramanis et al., 2012). Chance-constrained optimization plays a particularly important role in the context of electric power systems (Ozturk et al., 2004; Wang et al., 2012), applications of CCO can be found in various time-scales of power system operations and at different levels of the system.

The first chance-constrained program was formulated in (Charnes et al., 1958), then was extensively studied in the following 50 years, e.g. (Charnes and Cooper, 1959, 1963; Kataoka, 1963; Pinti, 1989; Sen, 1992; Prekopa et al., 1998; Ruszczynski and Shapiro, 2003, Ben-Tal et al., 2009). Previously, most methods to solve CCO problems deal with specific families of distributions, such as log-concave distributions (Miller and Wagner, 1965; Prekopa, 1995). Many novel methods appeared in the past ten years, e.g. scenario approach (Calafiore et al., 2006), sample average approximation (Pagnoncelli et al., 2009) and convex approximation (Ne-mirovski and Shapiro, 2006). Most of them are generic methods that are not limited to specific distribution families and require very limited knowledge about the uncertainties. In spite of many successful applications of these methods in various fields, there is a lack of comprehensive review and a critical compari-
1. Contributions of This Paper

The main contributions of this paper are threefold:

1. We provide a detailed tutorial on existing algorithms to solve chance-constrained programs and a survey of major theoretical results. To the best of our knowledge, there is no such review available in the literature;
2. We provide a comprehensive review on the applications of chance-constrained optimization in power systems, with focus on various interpretations of chance constraints in the context of power engineering.
3. We implement all the reviewed methods and develop an open-source Matlab toolbox (ConvertChanceConstraint), which is available on Github. We also provide a critical comparison of existing methods based numerical simulations on IEEE standard test systems.

1.3. Organization of This Paper

The remainder of this paper is organized as follows. Section 2 introduces chance-constrained optimization. Section 3 summarizes the fundamental properties of chance-constrained optimization problems. An overview of how to solve the chance-constrained optimization problem is described in Section 4, which outlines Section 5-7. Three major approaches to solve chance-constrained optimization (scenario approach, sample average approximation and robust optimization based methods) are presented in Section 5-7 respectively. Section 8 provides a comprehensive review of applications of CCO in power systems. The structure and usage of the Toolbox ConvertChanceConstraint is in Section 9. Section 10 also conducts numerical simulations and compares existing approaches to solve CCO problems. Concluding remarks are in Section 8.

1.4. Notations

The notations in this paper are standard. All vectors and matrices are in the real field \( \mathbb{R} \). Sets are in calligraphy fonts, e.g. \( \mathcal{S} \). The upper and lower bounds of a variable \( x \) are denoted by \( \bar{x} \) and \( \underline{x} \). The estimation of a random variable \( \epsilon \) is \( \hat{\epsilon} \). We use \( I_n \) to denote an all-one vector in \( \mathbb{R}^n \), and the subscript \( n \) is sometimes omitted for simplicity. The absolute value of vector \( x \) is \(|x|\), and the cardinality of a set \( \mathcal{S} \) is \(|\mathcal{S}|\). Function \([a]_+\) returns the positive part of variable \( a \). The indicator function \( \mathbb{1}_{x \geq 0} \) is one if \( x > 0 \). The floor function \( \lfloor a \rfloor \) returns the largest integer less than or equal to the real number \( a \). The ceiling function \( \lceil a \rceil \) returns the smallest integer greater than or equal to \( a \). \( \mathbb{E}[\xi] \) is the expectation of a random vector \( \xi \), \( \mathbb{V}(x) \) denotes the violation probability of a candidate solution \( x \), and \( \mathbb{P}(\epsilon) \) is the probability taken with respect to \( \epsilon \). The transpose of a vector \( a \) is \( a^\top \). Infimum, supremum and essential supremum are denoted by inf, sup and ess sup. The element-wise multiplication of the same-size vectors \( a \) and \( b \) is denoted by \( a \circ b \).

2. Chance-constrained Optimization

2.1. Introduction

We study the following chance-constrained optimization problem throughout this paper:

\[
\begin{align*}
\text{(CCO): } & \min_{x} c^\top x & \quad (1a) \\
\text{s.t. } & \mathbb{P}(f(x, \xi) \leq 0) \geq 1 - \epsilon & \quad (1b) \\
& x \in \mathcal{X} & \quad (1c)
\end{align*}
\]

where \( x \in \mathbb{R}^n \) is the decision variable and random vector \( \xi \in \mathbb{R}^d \) is the source of uncertainties. Without loss of generality, we assume the objective function is linear in \( x \) and does not depend on \( \epsilon \). Constraint (1b) is the chance (or probabilistic) constraint, it requires the inner constraint \( f(x, \xi) \leq 0 \) to be satisfied with high probability \( 1 - \epsilon \). The inner constraint \( f(x, \xi) : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}^m \) consists of \( m \) individual constraints, i.e. \( f(x, \xi) = (f_1(x, \xi), f_2(x, \xi), \cdots , f_m(x, \xi)) \). The deterministic constraints are represented by set \( \mathcal{X} \). Parameter \( \epsilon \) is called the violation probability of (CCO). Notice that \( f(x, \xi) \) is random due to \( \xi \), the probability \( \mathbb{P} \) is taken with respect to \( \xi \). Sometimes the probability is denoted by \( \mathbb{P}_\xi \) to avoid confusion.

It is worth mentioning that CCO is closely related with the theory of risk management. For example, an individual chance constraint \( \mathbb{P}(f_i(x, \xi) \leq 0) \geq 1 - \epsilon_i \) can be equivalently interpreted as a constraint on the value at risk \( \text{VaR}(f_i(x, \xi); 1 - \epsilon_i) \leq 0 \). This connection can be directly seen from the definition.

**Definition 1** (Value at Risk). Value at risk (VaR) of random variable \( \xi \) at level \( 1 - \epsilon \) is defined as

\[
\text{VaR}(\xi; 1 - \epsilon) := \inf \{ \gamma : \mathbb{P}(\xi \leq \gamma) \geq 1 - \epsilon \} \quad (2)
\]

More details about this can be found in Section 7.4 (Rockafellar and Uryasev, 2000), (Chen et al., 2010) and references therein.

CCO is closely related with two other major tools for decision making with uncertainties: stochastic programming and robust optimization. The idea of sample average approximation, which originated from stochastic programming, can be applied on chance-constrained programs (Section 6). Section 7 demonstrates how to construct uncertainty set in robust optimization with the assistance of chance constraints.

2.2. Joint and Individual Chance Constraints

Constraint (1b) is called a joint chance constraint because of its multiple inner constraints (Miller and Wagner, 1965), i.e.

\[
\mathbb{P}(f_1(x, \xi) \leq 0, f_2(x, \xi) \leq 0, \cdots , f_m(x, \xi) \leq 0) \geq 1 - \epsilon \quad (3)
\]

Alternatively, each one of the following \( m \) constraints is called an individual chance constraint:

\[
\mathbb{P}(f_i(x, \xi) \leq 0) \leq 1 - \epsilon_i , \; i = 1, 2, \cdots , m \quad (4)
\]

\footnote{1Using the epigraph formulation as mentioned in Campi et al., 2009 Boyd and Vandenberghe, 2004.}

\footnote{2github.com/xh00dx/ConvertChanceConstraint-ccc}
Joint chance constraints typically have more modeling power since an individual chance constraint is a special case ($m = 1$) of a joint chance constraint. But individual chance constraints are relatively easier to deal with (see Section 2.2 and 2.3). There are several ways to convert individual and joint chance constraints between each other.

First, a joint chance constraint can be written as a set of individual chance constraints. Notice (9) can be represented as \( \mathbb{P}_{\xi}(\cup_{m=1}^{M} \{ f_i(x,\xi) \geq 0 \}) \leq \varepsilon \). Since \( \varepsilon = \mathbb{P}_{\xi}(\cup_{m=1}^{M} \{ f_i(x,\xi) \geq 0 \}) \leq \sum_{m=1}^{M} \mathbb{P}_{\xi}(\{ f_i(x,\xi) \geq 0 \}) \) if \( \sum_{i} \mathbb{P}_{\xi}(\{ f_i(x,\xi) \geq 0 \}) \) is usually difficult to find such \( \{ \varepsilon_i \}_{i=1}^{M} \). Joint chance constraints typically have more modeling power (Violation Probability)

Definition 2

The critical definitions and assumptions below.

2.3. Critical Definitions and Assumptions

Theoretical results in the following sections are based on the critical definitions and assumptions below.

Definition 2 (Violation Probability). Let \( x^0 \) denote a candidate solution to (CCO), its violation probability is defined as

\[
\mathcal{V}(x^0) := \mathbb{P}_{\xi}(\{ f(x^0,\xi) \leq 0 \})
\]

where \( f(x,\xi) : \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R} \) is the pointwise maximum of functions \( \{ f_i(x,\xi) \}_{i=1}^{m} \) over \( x \) and \( \xi \), i.e.

\[
f(x,\xi) := \max \{ f_1(x,\xi), \ldots , f_m(x,\xi) \}.
\]

It is worth noting that converting \( \{ f_i(x,\xi) \}_{i=1}^{m} \) to \( f(x,\xi) \) could lose nice structures of the original constraints \( f(x,\xi) \leq 0 \) and cause more difficulties.

In this paper, we focus on the chance-constrained optimization problems with a joint chance constraint.

2.3. Critical Definitions and Assumptions

Theoretical results in the following sections are based on the critical definitions and assumptions below.

Definition 2 (Violation Probability). Let \( x^0 \) denote a candidate solution to (CCO), its violation probability is defined as

\[
\mathcal{V}(x^0) := \mathbb{P}_{\xi}(\{ f(x^0,\xi) \leq 0 \})
\]

where \( f(x,\xi) : \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R} \) is the pointwise maximum of functions \( \{ f_i(x,\xi) \}_{i=1}^{m} \) over \( x \) and \( \xi \), i.e.

\[
f(x,\xi) := \max \{ f_1(x,\xi), \ldots , f_m(x,\xi) \}.
\]

It is worth noting that converting \( \{ f_i(x,\xi) \}_{i=1}^{m} \) to \( f(x,\xi) \) could lose nice structures of the original constraints \( f(x,\xi) \leq 0 \) and cause more difficulties.

In this paper, we focus on the chance-constrained optimization problems with a joint chance constraint.

Definition 3. \( x^0 \) is a feasible solution for (CCO) if \( x^0 \in X \) and \( \mathcal{V}(x^0) \leq \varepsilon \). Let \( \mathcal{F}_\varepsilon \) denote the set of feasible solutions to the chance constraint (1b).

\[
\mathcal{F}_\varepsilon := \{ x \in \mathbb{R}^n : \mathcal{V}(x) \leq \varepsilon \} = \{ x \in \mathbb{R}^n : \mathbb{P}_{\xi}(f(x,\xi) \leq 0) \geq 1 - \varepsilon \}
\]

then \( x^0 \) is feasible to (CCO) if \( x^0 \in X \cap \mathcal{F}_\varepsilon \).

Although (CCO) seeks optimal solutions under uncertainties, it is a deterministic optimization problem. To better see this, (CCO) can be equivalently written as \( \min_{x \in X} c^T x, \text{ s.t. } \mathcal{V}(x) \leq \varepsilon \) or \( \min_{x \in X \cap \mathcal{F}_\varepsilon} c^T x \).

Definition 4. Let \( \sigma^* \) denote the optimal objective value of (CCO). For simplicity, we define \( \sigma^* = +\infty \) when (CCO) is infeasible and \( \sigma^* = -\infty \) when (CCO) is unbounded. Let \( x^* \) denote the optimal solution to (CCO) if it exists, and \( \sigma^* = c^T x^* \).

Definition 5. We say a candidate solution \( x^0 \) is conservative if \( \mathcal{V}(x^0) \ll \varepsilon \) or \( c^T x^0 \gg \sigma^* \).

Most existing theoretical results on (CCO) are built upon the following two assumptions.

Assumption 1. Let \( \Xi \) denote the support of random variable \( \xi \), the distribution \( \xi \sim \Xi \) exists and is fixed.

Assumption 1 only assumes the existence of an underlying distribution, but we do not necessarily need to know it to solve (CCO). Removing assumption 1 leads to a more general class of problem named distributionally robust optimization or ambiguous chance constraints. Section 3.4 discusses cases with assumption 1 removed.

Assumption 2. (1) Function \( f(x,\xi) \) is convex in \( x \) for every instance of \( \xi \), and (2) the deterministic constraints define a convex set \( X \).

3. Fundamental Properties

3.1. Hardness

Although CCO is an important and useful tool for decision making under uncertainties, it is very difficult to solve in general. Major difficulties come from two aspects:

\( \text{(D1):} \) It is very difficult to check the feasibility of a candidate solution \( x^0 \). Namely, it is intractable to evaluate the probability \( \mathbb{P}_{\xi}(f(x^0,\xi) \leq 0) \) with high accuracy. More specifically, calculating probability involves multivariate integration, which is NP-Hard (Khachiyan 1989). The only general method might be Monte-Carlo simulation, but it can be computationally intractable due to the curse of dimensionality.

\( \text{(D2):} \) It is very difficult to find the optimal solution \( x^* \) and \( \sigma^* \) to (CCO). The feasible region of (CCO) is often non-convex except a few special cases. For example, Section 3.3 shows the feasible region of (CCO) with separable chance constraints is a union of cones, which is non-convex in general. Although researchers have proved various sufficient conditions such that (CCO) is convex, it remains challenging to solve (CCO) because of difficulty (D1). Most of times, however, we are agnostic about the properties of the feasible region \( \mathcal{F}_\varepsilon \).

Theorem 1 (Luedtke et al. 2010; Qiu et al. 2014). (CCO) is strongly NP-Hard.

Theorem 2 (Ahmed and Xie 2018). Unless \( P = NP \), it is impossible to obtain a polynomial time algorithm for (CCO) with a constant approximation ratio.

Theorem 1 formalizes the hardness results of solving (CCO), Theorem 2 further demonstrates it is also difficult to obtain approximate solutions to (CCO): any polynomial algorithm is not able to find a solution \( x^*(\text{w}ith \sigma^* = c^T x^*) \) such that \( |\sigma^*/\sigma^*| \) is bounded by some constant \( C \). In other words, any polynomial-time algorithm could be arbitrarily worse.
3.2. Special Cases

Although CCO is NP-Hard to solve in general, there are several special cases in which solving (CCO) is relatively easy. The most well-known special case of (CCO) is (8), which was first proved in [Kataoka 1965].

\[
\min_{x \in \mathbb{R}^n} c^T x \quad \text{s.t. } P(a^T x + b^T \xi + e^T D x \leq e) \geq 1 - \epsilon
\]

Parameters \( a \in \mathbb{R}^n, b \in \mathbb{R}^d, D \in \mathbb{R}^{d \times d} \) and \( e \in \mathbb{R} \) are fixed coefficients. \( \xi \sim N(\mu, \Sigma) \) is a multivariate Gaussian random vector with mean \( \mu \) and covariance \( \Sigma \). Notice that (8) is an individual chance constraint with multivariate Gaussian coefficients. Let \( \Phi(\cdot) \) denote the inverse cumulative distribution function (CDF) function of a standard normal distribution. It is easy to show that if \( \epsilon \leq 1/2 \), (8) is equivalent to (9), which is a second order cone program (SOCP) and can be solved efficiently.

\[
\min_{x \in \mathbb{R}^n} c^T x \quad \text{s.t. } e - b^T \mu - (a + D^T \mu)^T x \geq \Phi^{-1}(1 - \epsilon) \sqrt{(b + D \mu)^T \Sigma (b + D \mu)}
\]

Remark 1. Many methods to solve (CCO) (e.g. [Beraldi and Ruszczynski 2002, Prekopa et al. 1998, Kess et al. 2007]) start with a partial or complete enumeration of \( p\)-efficient points. However, the number of \( p\)-efficient points could be astronomical or even infinite. See [Shapiro et al. 2009, Prkopa 1995] and references therein for the finiteness results of \( p\)-efficient points and complete theories and algorithms on \( p\)-efficient points.

3.3. Feasible Region

A chance-constrained program with only right hand side uncertainties (10) is considered in this section. With this example, we provide deeper understandings on the non-convexity of (CCO).

\[
\min_{x \in \mathbb{R}^n} c^T x \quad \text{s.t. } P(f(x) \leq \zeta) \geq p
\]

In (10), the inner function \( f(x) : \mathbb{R}^n \rightarrow \mathbb{R}^n \) is deterministic. The only uncertainty is the right-hand side value, represented by a random vector \( \zeta \in \mathbb{R}^n \). Chance constraints like (10) are also named separable chance constraints (or probabilistic constraints) since the deterministic and random parts are separated. We replace \( 1 - \epsilon \) with \( p \) in (10) to follow the convention in the existing literature.

Definition 6 (\( p\)-efficient points [Shapiro et al. 2009]). Let \( p \in (0,1) \), a point \( v \in \mathbb{R}^n \) is called a \( p\)-efficient point of the probability function \( P_\zeta(\xi \leq z) \), if \( P_\zeta(\xi \leq v) \geq p \) and there is no \( z \leq v \), and \( z \neq v \) such that \( P_\zeta(\xi \leq z) \geq p \).

Theorem 3 (Shapiro et al. 2009, Prkopa 1995). Let \( E \) be the index set of \( p\)-efficient points \( v', i \in E \). Let \( F_p := \{ x \in \mathbb{R}^n : P_\zeta(f(x) \leq \zeta) \geq p \} \) denote the feasible region of (10b), then it holds that

\[
F_p = \bigcup_{i \in E} K_i
\]

where each cone \( K_i \) is defined as \( K_i := v' + \mathbb{R}^n_+ \), \( i \in E \).

Theorem 3 shows the geometric properties of (CCO). Since the finite union of convex sets need not to be convex, the feasible region of (CCO) is generally non-convex.

3.4. Ambiguous Chance Constraints

The Ambiguous chance constraint is a generalization of chance constraints.

\[
P_{\zeta,P}(f(x,\xi) \leq 0) \geq 1 - \epsilon, \forall P \in \mathcal{P}.
\]

It requires the inner chance constraint \( f(x, \xi) \leq 0 \) holds with probability \( 1 - \epsilon \) for any distribution \( P \) belonging to a set of pre-defined distributions \( \mathcal{P} \).

Ambiguous chance constraints are particularly useful in the cases where only partial knowledge on the distribution \( P \) is available, e.g. we know only that \( P \) belongs a given family of \( \mathcal{P} \). However, it is generally more difficult to solve ambiguous chance constraints, and the theoretical results often rely on different assumptions of uncertainties. We only focus on CCO, the study on ambiguous chance constraints is beyond the scope of this paper.

4. An Overview of Solutions to CCO

This paper focuses on solutions to (CCO) with the following properties: (i) dealing with both difficulties (D1) and (D2) mentioned in Section 3.1; (ii) utilizing information from data (only) without making suspicious assumptions on the underlying distribution; and (iii) possessing rigorous guarantees on the feasibility and optimality of the returned solution. Section 4.3 explains the three properties in detail. Section 4.4 provides an overview of the methods with the properties above.

4.1. Classification of Solutions

Existing methods on (CCO) can be roughly classified into four categories [Ahmed and Shapiro 2008]:

(C1) when both difficulties (D1) and (D2) in Section 3.1 are absent, (CCO) is convex and the probability \( P(f(x, \xi) \leq 0) \) is easy to calculate. The only case without both (D1) and (D2) might be (8), which might be the only case of (CCO) that can be easily solved;

(C2) when only difficulty (D2) is absent, it is relatively easy to calculate \( P(f(x, \xi) \leq 0) \) (e.g. finite distributions with not too many realizations). As shown in Theorem 3 (CCO) is often non-convex and solutions typically rely on integer programming and global optimization [Ahmed and Shapiro 2008].
\begin{itemize}
  \item \textbf{(C3)} when only difficulty (D1) is absent, (CCO) is proved to be convex but remains difficult to solve because of (D2). This case often requires approximating the probability via simulation or specific assumptions. All examples mentioned in Section 3.2 except (8) belong to this category.
  \item \textbf{(C4)} when both difficulty (D1) and (D2) are present, it is almost impossible to find the optimal solution $x^*$ and $o^*$. Existing methods focus on obtaining approximate solutions or suboptimal solutions and constructing upper and lower bounds on the true objective value $o^*$ of (CCO).
\end{itemize}

Methods associated with (C1)-(C3) are briefly mentioned in Section 3, the remaining part of this paper focuses on more general and powerful methods in class (C4).

4.2. Prior Knowledge

In order to solve (CCO), a reasonable amount of prior knowledge on the underlying distribution $\xi \sim \Xi$ is necessary. Figure 1 illustrates three categories of prior knowledge:

\begin{itemize}
  \item \textbf{(K1)} we know the exact distribution $\xi \sim \Xi$ thus have complete knowledge on the underlying distribution;
  \item \textbf{(K2)} we know partially on the distribution (e.g. multivariate Gaussian distribution with bounded mean and variance) and thus have partial knowledge;
  \item \textbf{(K3)} we have a finite dataset $\{\xi_i^1\}_{i=1}^N$, this is another case of partial knowledge.
\end{itemize}

It can be seen that prior information in (K2) is a strict subset of (K1), also by sampling we can construct a dataset in (K3) from the exact distribution in (K1). It seems (K1) is the best starting point to solve (CCO). However, probability distributions are not known in practice, they are just models of reality and exist only in our imagination. What exists in reality is data. Therefore (K3) is the most practical case and becomes the focus of this paper. Almost all the data-driven methods to solve (CCO) are based on the following assumption.

\textbf{Assumption 3.} The samples (scenarios) $\xi_i^t$ ($i = 1, 2, \cdots, N$) in the dataset $\{\xi_i^t\}_{i=1}^N$ are independent and identically distributed (i.i.d.).

4.3. Theoretical Guarantees

In this paper, we mainly focus on the theoretical aspect of the reviewed methods. In particular, we pay special attention to the feasibility guarantee and optimality guarantee.

Given a candidate solution $x^*$ to (CCO), the first and possibly the most important thing to do is to check its feasibility, i.e. if $\forall (x^*) \leq \epsilon$. Although (D1) shows the difficulty of calculating $\forall (x^*)$ with high accuracy, there are various feasibility guarantees that either estimate $\forall (x^*)$ or provide upper bound on $\forall (x^*)$. The feasibility results can be classified into two categories: a-priori and a-posteriori guarantees. The a-priori ones typically provide prior conditions on (CCO) and dataset $\{\xi_i^t\}_{i=1}^N$ then guarantee the feasibility of $x^*$ before obtaining $x^*$. Examples of this type include Corollary 1, Theorem 6, 11 and 12. As the name suggests, the a-posteriori guarantees make effects after obtaining $x^*$ and observing some structural features associated with $x^*$. Examples include Theorem 1 and Proposition 1.

Given a candidate solution $x^*$ and the associated objective value $o^* = c^T x^*$, another important question to be answered is about the optimality gap $|o^* - o^*|$. Although finding $o^*$ is often an impossible mission because of difficulty (D2), bounding from below on $o^*$ is relatively easier. Sections 4.5 and 6.4 dedicate to algorithms of constructing lower bounds $o \leq o^*$.

4.4. An Overview

An overview of solutions to (CCO) and their relationships are in Figure 2. Akin methods are plotted in similar colors, and links among two circles indicate the connection of the two methods. Key references of each method are also provided. As shown in Fig. 2, chance-constrained optimization is a special case of ambiguous chance constraints where the set of distributions $\mathcal{P}$ is a singleton (Section 3.4). Therefore methods to solve ambiguous chance constraints can be applied on chance constraints as well. The methods and algorithms to solve chance-constrained optimization are the focus of this paper, we will mention the connection if some methods are closely related with ambiguous chance constraints.

Fig. 2 also outlines the first half of this paper, which dedicates to a review and tutorial on chance-constrained optimization. We summarize key results on the basic properties (Section 3), three main approaches to solve chance-constrained optimization problems, scenario approach (Section 5), sample average approximation (Section 6) and robust optimization (RO) based methods (Section 7).

5. Scenario Approach

5.1. Introduction to Scenario Approach

Scenario approach utilizes a dataset with $N$ scenarios $\{\xi_i^t\}_{i=1}^N$ to approximate the chance-constrained program (1) and obtains the following scenario problem (SP)$_N$:

\begin{align}
\text{(SP)$_N$:} \quad \min_{x \in X} \quad & c^T x \\
\text{s.t.} \quad & f(x, \xi_i^t) \leq 0, \cdots, f(x, \xi_i^N) \leq 0
\end{align}
SP\(_N\) seeks the optimal solution \(x^*_N\) which is feasible for all \(N\) scenarios. The scenario approach is a very simple yet powerful method. The most attractive feature of the scenario approach is its generality. It requires nothing except the convexity of constraints \(f(x, \xi)\) and \(X\). It is purely data-driven and makes no assumption on the underlying distribution.

**Remark 2.** SP\(_N\) is a random program. Both its optimal objective value \(o^*_N\) and optimal solution \(x^*_N\) depend on the random samples \(\xi_i\) and thus become random variables. In consequence, \(V(x^*_N)\) is also a random variable. Let \(N := \{1, 2, \cdots, N\}\) denote the index set of scenarios. The optimal objective value of SP\(_N\) is denoted by \(o^*(N)\) to emphasize its dependence on the random samples.

The theory of scenario approach are built upon the following assumption in addition to Assumptions 1, 2 and 3.

**Assumption 4 (Feasibility and Uniqueness (Campi and Garatti 2008)).** Every scenario problem (SP)\(_N\) is feasible, and its feasibility region has a non-empty interior. Moreover, the optimal solution \(x^*_N\) of (SP)\(_N\) exists and is unique.

If there exist multiple optimal solutions, the tie-break rules in (Calafiore and Campi 2005) can be applied to obtain a unique solution.

**Remark 3 (Sample Complexity \(N\)).** We first provide some intuition on the scenario approach. When solving (SP)\(_N\) with a very large number of scenarios, the solution \(x^*_N\) will be robust to almost every realization of \(\xi\), thus the violation probability goes to zero. Although \(x^*_N\) is a feasible solution to (CCO) as \(N \to +\infty\), it is overly conservative because \(V(x^*) \approx 0 \ll \epsilon\). On the other hand, using too few scenarios for SP\(_N\) might result in infeasible solutions \(x^*_N\) to (CCO). Notice that \(N\) is the only tuning parameter in scenario approach, the most important question in scenario approach theory is: what is the right sample complexity \(N\)? Namely, what is the smallest \(N\) such that \(V(x^*_N) \leq \epsilon\) (with high probability)? Rigorous answers to the
sample complexity question are built upon the structural properties of \( \text{SP}_N \).

5.2. Structural Properties of the Scenario Problem

Among \( N \) scenarios in the dataset \( \{\xi^i\}_{i=1}^N \), there are some important scenarios having direct impacts on the optimal solution \( x_N^* \).

**Definition 7** (Support Scenario ([Calafiore and Campi] 2005)). Scenario \( \xi^i \) is a support scenario for \( \text{SP}_N \) if its removal changes the solution of \( \text{SP}_N \). The set of support scenarios of \( \text{SP}_N \) is denoted by \( S \).

**Theorem 4** ([Calafiore and Campi] 2005, Calafiore, 2010). Under Assumption 2, the number of support scenarios in \( \text{SP}_N \) is at most \( n \), i.e. \( |S| \leq n \).

Theorem 4 is built upon Helly’s theorem and Radon’s theorem ([Rockafellar] 2015) in convex analysis. For non-convex problems, the number of support scenarios could be greater than the number of decision variables \( n \). An example for non-convex problems is provided in ([Campi et al.] 2018).

**Definition 8** (Fully-supported Problem ([Campi and Garatti] 2008)). A scenario problem \( \text{SP}_N \) with \( N \geq n \) is fully-supported if the number of support scenarios is exactly \( n \). Scenario problems with \( |S| < n \) are referred as non-fully-supported problems.

**Definition 9** (Non-degenerate Problem ([Campi and Garatti] 2008, Calafiore, 2010)). Problem \( \text{SP}_N \) is said to be non-degenerate, if \( o^*(N) = o^*(S) \). In other words, \( \text{SP}_N \) is non-degenerate if the solution of \( \text{SP}_N \) with all scenarios in place coincides with solution to the program with only the support scenarios are kept.

**Remark 4.** Distributions with finite support often lead to degenerate scenario problem \( \text{SP}_N \).

5.3. A-priori Feasibility Guarantees

Obtaining a-priori feasibility guarantees on the solution \( x_N^* \) to \( \text{SP}_N \) typically involves the following three steps:

1. exploring problem structure and obtain a upper bound \( \bar{h} \) on the number of support scenarios;
2. choosing a good sample complexity \( N(\epsilon, \beta, \bar{h}) \) using Corollary 1, Theorem 6 or Remark 5;
3. solving scenario problem \( \text{SP}_N \) and obtain optimal solution \( x_N^* \) and \( \sigma_N^* \).

**Theorem 5** ([Campi and Garatti] 2008). Under Assumption 1, 2 and 3 for a non-degenerate problem \( \text{SP}_N \), it holds that

\[
\mathbb{P}^N(\mathcal{V}(x_N^*) > \epsilon) \leq \sum_{i=1}^{n-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i}.
\]  

(14)

The probability \( \mathbb{P}^N \) is taken with respect to \( N \) random samples \( \{\xi^i\}_{i=1}^N \), and the inequality is tight for fully-supported problems.

As mentioned in Remark 2, \( \mathcal{V}(x_N^*) \) is a random variable, its randomness comes from drawing scenarios \( \{\xi^i\}_{i=1}^N \). For fully-supported problems, Theorem 5 shows the exact probability distribution of the violation probability \( \mathcal{V}(x_N^*) \), i.e.

\[
\mathbb{P}^N(\mathcal{V}(x_N^*) > \epsilon) = \sum_{i=1}^{n-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i},
\]  

(15)

the tail of a binomial distribution. We could use Theorem 5 to answer the sample complexity question in Remark 3.

**Corollary 1** ([Campi and Garatti] 2008). If we choose the number of scenarios \( N \) (the smallest such \( N \) is denoted by \( N_{2008} \)) such that

\[
\sum_{i=0}^{n-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i} \leq \beta
\]  

(16)

Let \( x_N^* \) denote the optimal solution to \( \text{SP}_N \), it holds that

\[
\mathbb{P}^N(\mathcal{V}(x_N^*) \leq \epsilon) \geq 1 - \beta
\]  

(17)

For fully-supported problems, \( N_{2008} \) is the tightest upper bound on sample complexity, which cannot be improved. For non-fully supported problems, it turns out \( N_{2008} \) can be further tightened. An improved sample complexity bound is provided in Theorem 6 based on the definition of Helly’s dimension.

**Definition 10** (Helly’s Dimension ([Calafiore] 2010)). Helly’s dimension of \( \text{SP}_N \) is the smallest integer \( h \) such that

\[
\text{ess sup}_{\xi \in \Xi^N} |S(\xi)| \leq h
\]

holds for any finite \( N \geq 1 \). We emphasize the dependence of support scenarios \( S \) on \( \xi \) by \( S(\xi) \).

**Theorem 6** ([Calafiore] 2010). Let \( h \) denote the Helly’s dimension for \( \text{SP}_N \). Under Assumption 1, 2 and 3 for a non-degenerate problem \( \text{SP}_N \), it holds that

\[
\mathbb{P}^N(\mathcal{V}(x_N^*) > \epsilon) \leq \sum_{i=0}^{h-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i}
\]

(18)

Equivalently, for a fixed confidence parameter \( \beta \in (0, 1) \), if the sample complexity \( N \) satisfies

\[
\sum_{i=0}^{h-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i} \leq \beta
\]

(19)

then the following probabilistic guarantee holds

\[
\mathbb{P}^N(\mathcal{V}(x_N^*) > \epsilon) \leq \beta
\]

(20)

The only difference between Theorem 6 and Theorem 5 (and Corollary 1) is replacing \( n \) with Helly’s dimension \( h \) in (18) and (19). Unfortunately, Helly’s dimension is often difficult to calculate, while finding upper bounds \( h \) on Helly’s dimension is usually a much easier task. Similarly we can replace \( h \) by \( \bar{h} \) in (18) and (19), the same theoretical guarantees still hold because of the monotonicity of (18) and (19) in \( N \) and \( h \). The support-
rank defined in [Schildbach et al. (2013)] is an upper bound on Helly’s dimension, some other upper bounds can be obtained by exploiting the structural properties of the problem, e.g. [Zhang et al. (2015)].

Remark 5 (Sample Complexity Revisited). A binary search type algorithm could be used to find $N_{2008}$. And a looser but hand upper bound is provided in (Campi et al. 2009):

$$N_{2009} := \frac{2}{\epsilon} \left( \ln(\frac{1}{\beta}) + n \right)$$  (21)

Notice $n$ in (21) can be replaced by $h$ or $\bar{h}$.

### 5.4. A-posteriori Feasibility Guarantees

When the desired violation probability $\epsilon$ is very small, the sample complexity of the a-priori guarantees grows with $1/\epsilon$ (Remark 5) and could be prohibitive. In other words, the a-priori approach is only suitable for the case where a sufficient amount of scenarios is always available. In many real-world applications (e.g., medical experiments, tests conducted by NASA), however, the amount of data is quite limited, and it could take months or cost a fortune to obtain a data point (experiment). Because of the limitation on the data availability, one of the most fundamental problem in data-driven decision making (e.g., system identification, quantitative finance) is to come up with good decisions or estimates with a moderate or even small amount of data. To overcome this, the scenario approach is extended towards a-posteriori feasibility guarantees.

Similar with the a-priori guarantees, obtaining a-posteriori guarantees typically requires taking the following three steps:

1. given dataset $\{\xi^i\}_{i=1}^N$, solve the corresponding scenario problem $\text{SP}_N$ and obtain $s^*_N$;
2. find support scenarios in $\{\xi^i\}_{i=1}^N$, whose number is denoted as $s^*_N$;
3. calculate the posterior violation probability $\epsilon(\beta, s^*_N, N)$ using Theorem 7

**Theorem 7** (Wait-and-Judge [Campi and Garatti (2016)]. Given $\beta \in (0, 1)$, for any $k = 0, 1, \cdots, n$, the polynomial equation in variable $t$

$$\frac{\beta}{N+1} \sum_{i=k}^N \frac{i}{k}^k - \binom{N}{k} t^{N-k} = 0$$  (22)

has exactly one solution $s(k)$ in the interval $(0, 1)$. Under Assumption 1 and 2 for a non-degenerate problem, it holds that

$$\mathbb{P}(\mathcal{V}(s^*_N) \geq \epsilon(s^*_N)) \leq \beta$$  (23)

Theorem 7 is particularly useful in the following cases: (i) the problem is not fully-support thus difficult to calculate a-priori bounds on number of support scenarios; or (ii) only a moderate or small amount of data points is available, it is difficult to meet the sample complexity from the a-priori guarantees.

Given a candidate solution $x^*$, the most straightforward method is to approximate $\mathcal{V}(x^*)$ by the empirical estimation $\hat{\epsilon}$ through Monte-Carlo simulation with $\hat{N}$ samples, i.e.

$$\hat{\epsilon} = \frac{1}{\hat{N}} \sum_{i=1}^{\hat{N}} I(f(x^*, \xi^i) > 0) = \frac{\hat{V}}{\hat{N}}$$  (24)

where $\hat{V} := \sum_{i=1}^{\hat{N}} I(f(x^*, \xi^i) > 0)$ is the total number of scenarios in which $x^*$ is infeasible. Although (24) only involves $f(x^*, \xi^i) > 0$ which is easy to calculate, it might require an astronomical number $\hat{N}$ to have accurate estimation $\hat{\epsilon}$ because of (D1). [Nemirovski and Shapiro (2006)] shows a method to bound $\mathcal{V}(x^*)$ from above using a dataset of a moderate size $\hat{N}$.

**Proposition 1** ([Nemirovski and Shapiro (2006)].) Given a candidate solution $x^*$ and $\hat{N}$ samples, let $\hat{V} := \sum_{i=1}^{\hat{N}} I(f(x^*, \xi^i) > 0)$ and $1 - \rho$ be the confidence parameter.

$$\tilde{\epsilon} := \max_{y \in [0,1]} \left\{ \sum_{i=0}^{\hat{N}} \binom{\hat{N}}{i} y^i(1 - y)^{\hat{N} - i} \geq \rho \right\}$$  (25)

After finding an upper bound $\tilde{\epsilon}$, so that if $\epsilon \leq \tilde{\epsilon}$, we may be sure that $\mathbb{P}(\mathcal{V}(x^*) \leq \epsilon) \leq 1 - \rho$.

**Remark 6.** Proposition 1 is closely related with scenario approach but with one fundamental difference. Theorem 7 only for solution from scenario approach, while Proposition 1 can evaluate solutions from other methods.

### 5.5. Optimality Guarantees of Scenario Approach

Scenario approach together with order statistics can be used to construct lower bounds $\underline{o}$ on $\alpha^*$ of (CCO).

**Proposition 2** ([Nemirovski and Shapiro (2006)]. Let $\{\xi^i\}_{i=1}^N$ $(i = 1, 2, \cdots, M)$ be $M$ independent datasets of size $N$. For the $j$ th dataset, we solve the associated scenario problem $\text{SP}_N$ and calculate the optimal value $o^*_j$ $(j = 1, 2, \cdots, M)$. Without loss of generality, we assume that $o^*_1 \leq o^*_2 \leq \cdots \leq o^*_M$.

Given $\delta \in (0, 1)$, let us choose positive integers $M, N, L$ in such a way that

$$\sum_{i=0}^{L-1} \binom{M}{i} \left[ (1 - \epsilon)^N \right]^i \left[ (1 - (1 - \epsilon)^N)^M \right]^{M - i} \leq \delta$$  (26)

then with probability of at least $1 - \delta$, the random quantity $o_L^*$ gives a lower bound for the true optimal value $x^*$.

[[Pagnoncelli et al. (2009)]] shows that appropriate $N$ should be the order of $\mathcal{O}(1/\epsilon)$ as $[1 - (1 - \epsilon)^N]^M \approx (1 - \exp(-\epsilon N))^M$. Typically we choose proper values for $N$ and $M$ first, then find out the largest positive integer $L$ that (26) holds true.

Proposition 2 turns out to be a general framework to construct lower bounds on (CCO), ([Pagnoncelli et al. (2009)]) extends the framework towards generating bounds using sample average approximation, which is introduced in Section 6.4.

### 6. Sample Average Approximation

#### 6.1. Introduction to Sample Average Approximation

The idea of using sample average approximation to handle chance constraints first appeared in [Sen (1992)] and was subse-
quently improved with rigorous theoretical results in (Luedtke and Ahmed, 2008).

Let $f(x, \xi) := \max \left\{ f_j(x, \xi), \cdots, f_m(x, \xi) \right\}$, then (CCO) is equivalent to $\min_{x \in X} c^T x$, s.t. $P(\bar{f}(x, \xi) \leq 0) \geq 1 - \epsilon$. Sample Average Approximation (SAA) approximates the true distribution of the random variable $\bar{f}(x, \xi)$ using the empirical distribution from $N$ samples $\{ x^{(i)} \}^N_{i=1}$, i.e. $P(\bar{f}(x, \xi) \leq 0)$ is approximated by $\frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(x^{(i)} \leq 0)$.

(SAA) is also a chance constrained optimization problem, but with two major differences from (CCO): (i) (SAA) is based on the empirical (discrete) distribution from the true distribution of $\xi$ as in (CCO); (ii) (SAA) has the violation probability $\epsilon$ instead of $\epsilon$ in (CCO).

There are two critical questions to be addressed about (SAA). What is the connection of solutions of (SSA) with that of (CCO)? How to solve (SAA)? We first answer the second question in Sections 6.2 then focus on the theoretical results of connecting (SAA) with (CCO).

6.2. Solving Sample Average Approximation

(SAA) can be reformulated as a mixed integer program (MIP) by introducing variables $z \in \{0, 1\}^N$ (Ruszczyski, 2002; Luedtke and Ahmed, 2008). Binary variable $z_i$ is an indicator if $f(x, \xi) \leq 0$ is being violated in sample $i$, i.e.

$$z_i = \mathbb{I}(x, \xi) > 0$$

(28)

can be equivalently written as $f(x, \xi) \leq Mz_i$ with a sufficiently large coefficient $M \in \mathbb{R}^+$. Since $f(x, \xi)$ is the maximum over $m$ functions $f_j(x, \xi)$, $f(x, \xi)$ is maximum over $m$ functions $f_j(x, \xi)$ implies $f_j(x, \xi) \leq Mz_i$, $j = 1, 2, \cdots, m$. Then (SAA) is equivalent to (29), in which $1_m$ is an all one vector with size $m$.

$$\min_{x \in X} c^T x$$

s.t. $f(x, \xi) - Mz_1 \leq 0$

$$\vdots$$

$$f(x, \xi) - Mz_N \leq 0$$

(29a)

(29b)

(29c)

(29d)

(29e)

is equivalent to (SAA) for general function $f(x, \xi)$, but formulations with big-M are typically weak formulations. Introducing big coefficients $M$ might cause numerical issues as well. Stronger formulations of (SAA) are possible by exploiting the structural features of $f(x, \xi)$. A good example is chance-constrained linear program with separable probabilistic constraints: $\min_{x \in X} c^T x$ s.t. $\mathbb{P}(Tx \geq \xi) \geq 1 - \epsilon$. An equivalent but stronger formulation without big M is (Luedtke et al., 2010).

$$\min_{x \in X} c^T x$$

s.t. $Tx = v$

$$v + \xi \geq 0$$

$$\frac{1}{N} \sum_{i=1}^{N} z_i \leq \epsilon$$

$$z_i \in \{0, 1\}, i = 1, 2, \cdots, N$$

(30a)

(30b)

(30c)

(30d)

(30e)

Various strong formulations for (SAA) can be found in (Luedtke et al., 2010) and references therein. (29) and (30) are mixed integer programs, some well-known techniques from integer programming theory can speed up the process of solving (SAA), e.g. adding cuts (Tanner and Ntaimo, 2010; Luedtke et al., 2010; Kkyavuz, 2012) and decompositions (Zeng et al., 2017; Zeng and An, 2014).

6.3. Feasibility Guarantees of SAA

Various feasibility guarantees of (SAA) are proved in (Luedtke and Ahmed, 2008; Pagnoncelli et al., 2009), e.g. the asymptotic behavior of (SAA), when $X$ is finite, the case of separable chance constraints (10b), and when $f(x, \xi)$ is Lipschitz continuous. In this section, we only present the Lipschitz case, which could be used for simulations in Section ??.

Assumption 5. There exists $L > 0$ such that

$$|f(x, \xi) - f(x', \xi)| \leq L |x - x'|_{\infty}, \forall x, x' \in X \text{ and } \forall \xi \in \Xi.$$  (31)

Theorem 8 (Luedtke and Ahmed, 2008). Suppose $X$ is bounded with diameter $D$ and $f(x, \xi)$ is $L$-Lipschitz for any $\xi \in \Xi$ (assumption 5). Let $\epsilon \in (0, \bar{\epsilon}), \theta \in (0, \epsilon - \bar{\epsilon})$ and $\gamma > 0$. Then

$$P(\mathcal{F}^N_{\epsilon, \gamma} \subseteq \mathcal{F}_\epsilon) \geq 1 - \frac{1}{\theta^N} \exp(-2N(\epsilon - \bar{\epsilon})^2)$$

(32)

where the feasible region of (SAA) is defined as

$$\mathcal{F}^N_{\epsilon, \gamma} := \left\{ x \in X : \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(x^{(i)} \leq 0) \geq 1 - \epsilon \right\}.$$ (33)

For fixed $\epsilon$ and $\bar{\epsilon}$, if we choose $\theta = (\epsilon - \bar{\epsilon})/2$ and a small number $\gamma > 0$, then Theorem 8 suggests that using

$$N \geq \frac{2}{(\epsilon - \bar{\epsilon})^2} \left[ \frac{1}{\beta} + n \ln \left( \frac{2LD}{\gamma} \right) + \ln \left( \frac{2}{\epsilon - \bar{\epsilon}} \right) \right]$$

(34)

number of samples, solutions of (SAA) is feasible to (CCO) with high probability $1 - \beta$, i.e. $P(\mathcal{F}^N_{\epsilon, \gamma} \subseteq \mathcal{F}_\epsilon) \geq 1 - \beta$.

The results in Theorem 8 look quite similar to those of scenario approach (e.g. Remark 5). Indeed, (SAA) with $\epsilon = 0$ is exactly the same as the scenario program $\mathcal{SP}_N$. However, one major difference of Theorem 8 from the scenario approach theory is that: Theorem 8 holds for every feasible point of (SAA),
while the scenario approach theory (e.g. Theorem 5) only holds for the optimal solution $x^*$.

Although Theorem 5 provides explicit sample complexity bounds for (SAA) to obtain feasible solution, it requires some efforts to be applied, e.g. tuning parameters $(\epsilon, \theta)$ and calculation of $L$ and $D$. Campi and Garatti [2011] provides a similar but more straightforward theoretical result.

Theorem 9 (Sampling & Discarding [Campi and Garatti 2011]). If we draw $N$ samples and discard any $k$ of them, then use scenario approach with the remaining $N-k$ samples. If $N$ and $k$ satisfy

$$\left(\frac{k+n-1}{k}\right) \cdot \sum_{i=0}^{k-1} \binom{N}{i} (1-\epsilon)^{N-i} \leq \beta$$

then $\Pr\left(\sum_{i=0}^{k-1} \binom{N}{i} (1-\epsilon)^{N-i} \leq \beta \right)$.

Then if we choose $\epsilon > \epsilon$ and assume that (CCO) has an optimal solution. Then

$$\Pr(\hat{\alpha}^N \leq \alpha^*) \geq 1 - \exp(-2N(\epsilon - \epsilon)^2).$$

Theorem 10 directly suggests a method to construct lower bounds on (CBO).

Proposition 3. If we choose $\epsilon > \epsilon$ and $N \geq \frac{1}{\epsilon^2} \log(\frac{1}{\epsilon})$, let $\hat{\alpha}^N$ denote the objective value of (SAA), then $\hat{\alpha}^N$ is a lower bound with probability at least $1-\delta$, i.e. $\Pr(\hat{\alpha}^N \leq \alpha^*) \geq 1 - \delta$.

There is an alternative method using SAA to generate lower bounds of (CCO). Luedtke and Ahmed [2008] extends the framework in Proposition 2 towards SAA.

Proposition 4 ([Luedtke and Ahmed 2008]). Take $M$ sets of $N$ independent samples $(\xi^{j, i}_{\kappa})_{i=1}^{N}$, $(j = 1, 2, \cdots, M)$. For the dataset $(\xi^{j, i}_{\kappa})_{i=1}^{N}$, we solve the associated (SAA) problem and calculate the associated objective value $\hat{\alpha}^N_{j, \kappa}$ (for simplicity $\hat{\alpha}^N_{j}$ and $j = 1, 2, \cdots, M$). Without loss of generality, we assume that $\hat{\alpha}^N_{1} \leq \hat{\alpha}^N_{2} \leq \cdots \leq \hat{\alpha}^N_{M}$.

Given $\delta \in (0, 1)$, $\epsilon \in (0, 1)$, let us choose positive integers $N, L, M (L \leq M)$ such that

$$\sum_{i=0}^{L-1} \binom{M}{i} b(\epsilon, \epsilon, N, N) \left[1 - b(\epsilon, \epsilon, N, N)\right]^{M-i} \geq \delta$$

where $b(\epsilon, \epsilon, N, N) := \sum_{i=0}^{N} \binom{N}{i} (1-\epsilon)^{N-i}$.

Then $\hat{\alpha}^N_{j}$ serves as a lower bound to (CCO) with probability at least $1-\delta$.

7. Robust Optimization Related Methods

7.1. Introduction to Robust Optimization

The last category of solutions to (CCO) is closely related with robust optimization, its typical form is shown in (38).

$$(\text{RC}): \min_{x \in X} c^T x$$

$s.t. \ f(x, \xi) \leq 0, \ \forall \xi \in \mathcal{U}$

(38) finds the optimal solution which is feasible to all realizations of uncertainties in a predefined set $\mathcal{U}$. (38) is called the Robust Counterpart (RC) of the original problem (CCO). By constructing an uncertainty set $\mathcal{U}$ with proper shape and size, there will be theoretical or statistical connections of solutions to (RC) and (CCO).

Designing uncertainty sets lies at the heart of robust optimization. A good uncertainty set should meet the following two standards:

(S1): the resulting (RC) problem is computationally tractable.

(S2): the optimal solution to (RC) is not too conservative or overly optimistic.

(38) can be equivalently written as min$_{x \in X}$ $c^T x$, s.t. $x \in \mathcal{F}_\mathcal{U}$, where $\mathcal{F}_\mathcal{U} := \{x \in \mathbb{R}^n : f(x, \xi) \leq 0, \ \forall \xi \in \mathcal{U}\}$. Unfortunately, (RC) for general convex problems (under Assumption 2) is not always computationally tractable. For example, (RC) of a second order cone program (SOCP) with polyhedral uncertainty set is NP-Hard [Ben-Tal and Nemirovskii 1998; Ben-Tal et al. 2002; Bertsimas et al. 2011].

Since (S2) is directly related with chance constraints, we focus on (S2) in this paper. A good reference of tractability results is [Bertsimas et al. 2011].

7.2. Safe Approximation

Most of RO-related methods to solve (CCO) are built on the idea of safe approximation.

Definition 11 (Safe Approximation). An optimization problem (SA) is called a safe approximation of (CCO) if every feasible solution to (SA) is also feasible to (CCO), i.e. $\mathcal{F} \subseteq \mathcal{F}$.

The safe approximation problem (SA) can have equivalent forms with auxiliary variables $y$: min$_{x \in X, y}$ s.t. $(x, y) \in \mathcal{H}$. In this case, $\mathcal{F} = \{x \in \mathbb{R}^n : \exists y \text{ s.t. } (x, y) \in \mathcal{H}\}$. Since $\mathcal{F} \subseteq \mathcal{F}$, every solution to (CA) is feasible to (CCO). Therefore every optimal solution to (CA) is suboptimal to (CCO) and serves as an upper bound on (CCO).

The first batch of results connecting robust optimization and chance constraints appeared in the literature on robust linear programs (RLPs). Section 7.3 summarizes key results of RLPs. Then a generalized framework is in Section 7.4.
7.3. Robust Linear Optimization with Probabilistic Guarantees

In this subsection, we focus on the chance-constrained linear program (CCLP) with an individual chance constraint. The robust counterpart of CCLP is

\[
\min_{x \in \mathbb{R}^d} c^T x \quad \text{s.t. } \mathbb{P}(\lambda^0 + \sum_{i=1}^{d} \lambda_i \xi_i^T x \leq (\beta^0 + \sum_{i=1}^{d} \beta_i \xi_i), \forall \xi \in \mathcal{U}_e) \geq 1 - \epsilon
\]

where \(\{\lambda_i\}_{i=0}^d\) and \(\{\beta_i\}_{i=0}^d\) are coefficients with proper size. The robust counterpart of CCLP is

\[
\min_{x \in \mathbb{R}^d} c^T x \quad \text{s.t. } (\lambda^0 + \sum_{i=1}^{d} \lambda_i \xi_i^T x \leq (\beta^0 + \sum_{i=1}^{d} \beta_i \xi_i), \forall \xi \in \mathcal{U}_e)
\]

The theoretical results on probabilistic guarantees are built on the following assumption.

**Assumption 6.** \(\{\xi_i\}_{i=0}^d\) are independent of each other with zero mean and take values on \([-1, 1]^d\), i.e. \(E[\xi_i] = 0\) and \(\xi_i \in [-1, 1]\) for \(i = 1, 2, \cdots, d\).

Clearly, under Assumption 6, a natural choice of uncertainty set is the box \(\mathcal{U}^\text{box} = \{\xi \in \mathbb{R}^d : -1 \leq \xi \leq 1\}\). The feasible set \(\mathcal{F}_U^\text{box} = \{x \in \mathbb{R}^d : f(x, \xi) \leq 0, \forall \xi \in \mathcal{U}^\text{box}\}\) is a safe approximation to \(\mathcal{F}_U\), i.e. \(\mathcal{F}_U^\text{box} \subseteq \mathcal{F}_U\). However, using \(\mathcal{U}^\text{box}\) leads to \(\mathbb{P}(f(x, \xi) \geq 0) = 0 \ll \epsilon\), which is overly conservative. The following safe approximations are less conservative.

**Lemma 1** ([Ben-Tal and Nemirovski, 1999]). Under Assumption 6 for every \(\Omega \geq 0\), it holds that

\[
\mathbb{P}\left(\sum_{i=1}^{d} z_i \xi_i \leq \Omega \sqrt{\sum_{i=1}^{d} z_i^2} \right) \geq 1 - \exp\left(-\frac{\Omega^2}{2}\right)
\]

Choosing \(z = \lambda^0 \xi\) and \(\Omega \geq \sqrt{2 \ln(1/\epsilon)}\), Lemma 1 leads to Corollary 2.

**Corollary 2** ([Ben-Tal and Nemirovski, 1999, Ben-Tal et al., 2009]). \(\mathcal{U}^\text{ball} = \{\xi \in \mathbb{R}^d : ||\xi|| \leq \sqrt{2 \ln(1/\epsilon)}\}\) is a safe approximation to \(\mathcal{U}_e\).

\[
(\lambda^0 + \sum_{i=1}^{d} \lambda_i \xi_i^T x \leq (\beta^0 + \sum_{i=1}^{d} \beta_i \xi_i), \forall \xi \in \mathcal{U}_e^\text{ball})
\]

where \(\mathcal{U}_e^\text{ball} = \{\xi \in \mathbb{R}^d : ||\xi|| \leq \sqrt{2 \ln(1/\epsilon)}\}\).

Similar results hold for various uncertainty sets. Theorem 11 summarizes the main results.

**Theorem 11** ([Ben-Tal et al., 2009, Ben-Tal and Nemirovski, 1999, Bertsimas and Sim, 2004]). \(\mathcal{U}_e\) is a safe approximation to \(\mathcal{U}_e\) if \(\mathcal{U}_e\) is one of the following:

\[
\mathcal{U}^\text{box} = \{\xi \in \mathbb{R}^d : -1 \leq \xi \leq 1\}
\]

\[
\mathcal{U}^\text{ball} = \{\xi \in \mathbb{R}^d : ||\xi|| \leq \sqrt{2 \ln(1/\epsilon)}\}
\]

\[
\mathcal{U}^\text{ball-box} = \{\xi \in \mathbb{R}^d : ||\xi|| \leq 1, ||\xi|| \leq \sqrt{2 \ln(1/\epsilon)}\}
\]

\[
\mathcal{U}^\text{budget} = \{\xi \in \mathbb{R}^d : ||\xi|| \leq 2 \ln(1/\epsilon)\}
\]

And the resulting (RC) are all computationally tractable (cf Chapter 2 of [Ben-Tal et al., 2009]).

7.4. Convex Approximation

Results in Section 7.3 can be generalized beyond linear constraints using a generating function based framework. This line of work first appeared in ([Pint [1989]], then was completed in ([Nemirovski and Shapiro, 2006]). In this section, we consider the safe approximation of an individual chance constraint \(\mathbb{P}(f(x, \xi) \leq 0) \geq 1 - \epsilon\), where \(f : \mathbb{R}^n \times \mathbb{R}^d \rightarrow \mathbb{R}\).

**Definition 12** (Generating Function). A function \(\phi : \mathbb{R} \rightarrow \mathbb{R}\) is called a (one-dimensional) generating function if it is nonnegative valued, nondecreasing, convex and satisfying the following property:

\[
\phi(z) > \phi(0) = 1, \forall z > 0
\]

The convex approximation method is built upon the following lemma.

**Lemma 2.** For a positive constant \(t \in \mathbb{R}_+\) and a random variable \(z \in \mathbb{R}\), it holds that

\[
\mathbb{E}[\phi(t^{-1} z)] \geq \mathbb{E}[1_{t^{-1} z \geq 0}] = P_z(t^{-1} z \geq 0) = P(z \geq 0)
\]

Replace \(z\) with \(f(x, \xi)\), then \(\mathbb{E}[\phi(t^{-1} f(x, \xi)\xi)] \geq P_z(f(x, \xi) > 0) = P_z(t^{-1} f(x, \xi) > 0)\). In other words, \(\mathbb{E}[\phi(t^{-1} f(x, \xi))] \leq \epsilon\) is a safe approximation to \(P_z(f(x, \xi) \leq 0) \geq 1 - \epsilon\).

**Theorem 12** (Convex Approximation [Nemirovski and Shapiro, 2006]). Let \(\phi(z)\) be a generating function, then (CA) is a safe approximation to (CCO).

\[
(\text{CA}): \min_{x \in \mathbb{R}^d} c^T x
\]

\[
\text{s.t. } \inf_{t > 0} \left\{ \mathbb{E}[\phi\left(\frac{f(x, \xi)}{t}\right)] - \epsilon t \right\} \leq 0
\]

Under assumption 2 (CA) is convex in \(x\).

**Remark 7.** We can get rid of the strict inequality \(t > 0\) by approximating it using \(t \geq \delta\), where \(\delta\) is very small positive number (e.g. \(\delta = 10^{-5}\)). The price we pay is a bit more conservativeness. Furthermore, we can show that (CA) is equivalent to (48), which is convex in \((x, t)\).

\[
\min_{x \in \mathbb{R}^d} c^T x
\]

\[
\text{s.t. } t \mathbb{E}[\phi\left(\frac{f(x, \xi)}{t}\right)] - \epsilon t \leq 0
\]
Choosing a good generating function lies at the heart of convex approximation framework. Choices of generating functions include Markov bound $\phi(z) = [1 + z]_+$, Chernoff bound $\phi(z) = \exp(z)$, Chebyshev bound $\phi(z) = (z + 1)^2$, and Traditional Chebyshev bound $\phi(z) = (z + 1)^2$.

The least conservative generating function (Nemirovski and Shapiro 2006) is the Markov bound $\phi(z) = [1 + z]_+$. And (Nemirovski and Shapiro 2006) shows the connection of convex approximation to risk theory.

**Definition 13** (Conditional Value at Risk), Conditional value at risk (CVaR) of random variable $z$ at level $1 - \epsilon$ is defined as

$$\text{CVaR}(z; 1 - \epsilon) := \inf_{\gamma} (\gamma + \frac{1}{\epsilon} \mathbb{E}[z - \gamma])$$  \hspace{1cm} (49)

**Proposition 5.** (CA) with Markov bound $\phi(z) = [z + 1]_+$ is equivalent to (50).

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$\text{CVaR}(f(x, \xi) ; 1 - \epsilon) \leq 0$$  \hspace{1cm} (50a)

We sketch the proof of Proposition 5 below, more details can be found in (Ben-Tal et al., 2009). Using Markov bound $\phi(z) = [z + 1]_+$, constraint (47b) becomes

$$\frac{\mathbb{E}[f(x, \xi) + 1]}{\epsilon} \leq 0$$

indeed we could replace $\inf_{f > 0}$ with $\inf_f$ then let $\gamma = -t$, constraint (47b) becomes

$$\frac{\mathbb{E}[f(x, \xi) - \gamma]}{\epsilon}$$

Section 2 shows an individual chance constraint $\mathbb{P}(f(x, \xi) \geq 1 - \epsilon$ is equivalent to $\text{VaR}(f(x, \xi); 1 - \epsilon) \leq 0$. And $\text{CVaR}(z; 1 - \epsilon) \geq \text{VaR}(z; 1 - \epsilon)$. Clearly $\text{CVaR}(f(x, \xi); 1 - \epsilon) \leq 0$ implies $\text{VaR}(f(x, \xi); 1 - \epsilon) \leq 0$. This is the idea of safe approximation.

**Remark 8** (Data-driven Implementation) (Rockafellar and Uryasev 2000). Using the epigraph trick by introducing auxiliary variables $u_k \geq 0$

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$\frac{1}{N} \sum_{i=1}^{N} f(x, \xi^i) + t \leq \epsilon$$  \hspace{1cm} (51b)

Using the epigraph trick by introducing auxiliary variables $u_k \geq 0$

$$\min_{x \in \mathbb{R}^n} c^T x$$

subject to

$$f(x, \xi^i) + t \leq u_i$$  \hspace{1cm} (52b)

$$\frac{1}{N} \sum_{i=1}^{N} u_i \leq \epsilon$$  \hspace{1cm} (52c)

$$u_i \geq 0, i = 1, 2, \ldots, N$$  \hspace{1cm} (52d)

Obviously (52) is a convex program. (52) is a safe approximation to (CCO) with a sufficient number of data points ($N$ is large enough). However, (Chen et al. 2010) points out that the approximating CVaR with data may not necessarily yield a safe approximation. A more rigorous solution can be found in (Chen et al. 2010).

The generating function based framework in (Nemirovski and Shapiro, 2006) was further improved and completed in (Ben-Tal et al., 2009) and (Nemirovski, 2012). But the methods proposed there are mainly analytical and focus on ambiguous chance constraints or distributionally robust problems, which is beyond the scope of this paper.

8. **Concluding Remarks**

This paper consists of two parts. The first part presents a comprehensive review on the fundamental properties, key theoretical results and three classes of algorithms for chance-constrained optimization. An open-source MATLAB toolbox ConvertChanceConstraint is developed to automate the process of translating chance constraints to compatible forms for mainstream optimization solvers. The second part of this paper presents major applications of chance-constrained optimization in power systems. We also present a detailed and critical comparison of existing algorithms to solve chance-constrained programs on IEEE benchmark systems.

Many interesting directions are open for future research. In terms of theoretical investigation, an analytical comparison of existing solutions to CCO is necessary to substantiate the fundamental insights obtained from numerical simulations. In terms of application development, many existing results can be improved by exploiting the structural properties of the problem. The application of chance-constrained optimization in electric energy systems go beyond operational planning practices. For example, it would be worth investigating into the economic interpretation of market power issues through the lens of chance-constrained optimization.

**References**

Ahmed, S., 2018. Relaxations and Approximations of Chance Constraints. URL: https://lamp2018.sciencesconf.org/

Ahmed, S., Shapiro, A., 2008: Solving chance-constrained stochastic programs via sampling and integer programming. Tutorials in Operations Research 10, 261–269.

Ahmed, S., Xie, W., 2018. Relaxations and approximations of chance constraints under finite distributions. Mathematical Programming 170, 43–65.

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

Ben-Tal, A., Nemirovski, A., 1998. Robust convex optimization. Mathematics of operations research 23, 769–805.

Ben-Tal, A., Nemirovski, A., 1999. Robust solutions of uncertain linear programs. Operations research letters 25, 1–13.

Bertsimas, D., Brown, D.B., Caramanis, C., 2011. Theory and applications of robust optimization. SIAM review 53, 464–501.

Bertsimas, D., Gupta, V., Kallus, N., 2018. Data-driven robust optimization. Mathematical Programming.
Zeng, B., An, Y., Kuznia, L., 2017. Chance constrained mixed integer program: Bilinear and linear formulations, and Benders decomposition. Mathematical Programming.

Zhang, X., Grammatico, S., Schildbach, G., Goulart, P., Lygeros, J., 2015. On the sample size of random convex programs with structured dependence on the uncertainty. Automatica 60, 182–188.