A General Scenario Theory For Security-Constrained Unit Commitment With Probabilistic Guarantees

Xinbo Geng, Le Xie, and M. Sadegh Modarresi

Abstract

This paper addresses the challenges of security-constrained unit commitment (SCUC) under uncertainties from high renewables. We propose a chance-constrained SCUC (c-SCUC) framework, which ensures that the risk of violating constraints is within an acceptable threshold. Using the scenario approach, c-SCUC is reformulated to the scenario-based SCUC (s-SCUC) problem. By choosing an appropriate number of scenarios, we provide theoretical guarantees on the posterior risk level of the solution to s-SCUC. Inspired by the latest progress of the scenario approach on non-convex problems, we demonstrate the structural properties of general scenario problems and analyze the specific characteristics of s-SCUC. Those characteristics were exploited to benefit the scalability and computational performance of s-SCUC. Case studies performed on the IEEE 118-bus system suggest that the scenario approach could be an attractive solution to practical power system applications.

I. INTRODUCTION

Security-constrained Unit commitment (SCUC) is one of the most important decisions made in power system operational planning. The SCUC problem seeks the most cost-efficient on/off decisions and dispatch schedules for generators, considering various security constraints such as generation and transmission capacity limits under contingencies.

X. Geng and L. Xie are with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, 77840. Email: {xbgeng, le.xie}@tamu.edu. M.S. Modarressi is with Burns & McDonnell, Houston, TX, 77027. Email: mmsmodarresi@burnsmcd.com. This work is supported in part by Power Systems Engineering Research Center, and in part by NSF OAC-1934675 and ECCS-1839616.
SCUC is a decision making problem in uncertain environments by its nature. Conventional SCUC problems ensure the system is secured for a number of outages in generation, transmission, or other elements within the system. As the generation portfolio is shifting towards renewable resources, SCUC, a crucial part of power system day-ahead scheduling, needs to evolve to address the flexibility concerns.

Stochastic optimization (SO) and robust optimization (RO) are two common approaches for decision making under uncertainties. Both SO and RO have been successfully applied in various areas. SO relies on probabilistic models to depict uncertainties and often optimizes the objective function in the presence of randomness. SO has found many successful applications in power system operations and planning problems. For instance, references [1]–[3] formulate and solve the stochastic unit commitment problem, which minimizes the expected commitment and dispatch costs. RO takes an alternative approach, in which the uncertainty model is set-based and typically deterministic [4]. Recently, researchers in [5] formulated and solved the robust unit commitment problem, which minimizes the commitment and dispatch costs for the worst case in a predefined uncertainty set.

This paper provides a perspective of solving SCUC in uncertain environments through the lens of chance-constrained optimization (CCO), which is akin to both SO and RO [6]. The main distinction between CCO and SO/RO is the chance constraint (see (1b) and (2b) in Section II), which explicitly considers the feasibility of solutions under uncertainties. Various formulations of chance-constrained SCUC (c-SCUC) have been proposed, e.g. [7]–[15]. As mentioned in [6], chance-constrained optimization problems can be solved using the scenario approach, sample average approximation, or robust optimization based techniques. We take c-SCUC as an example. It is solved via sample average approximation in [9]–[15] and via robust optimization based techniques in [16]–[18].

The scenario approach is a well-known algorithm to solve CCO problems [21]–[23]. It was mainly targeted at convex problems (see Assumption 3), whereas SCUC is non-convex by nature due to on/off commitment decisions. Consequently, the scenario approach was considered not

\(^1\)The method used in [17], [18] is based on [19]. It utilizes the sample complexity bound by an earlier version of the scenario approach [20], however, it is more closely related with robust optimization. Furthermore, the results in [17], [18] might be overly conservative, since the sample complexity bound by [20] could be significantly tightened by in-depth analysis of the scenario approach, see Theorem 1 for more details.
applicable for c-SCUC. An extended version of the scenario approach was proposed recently in [24], which makes it applicable for non-convex problems such as SCUC.

Our previous paper [25] might be the first attempt to apply the scenario approach on unit commitment. However, the formulation therein is greatly simplified by ignoring some critical constraints such as transmission capacities. Enabled by this limiting assumption, [25] shows that the original scenario approach remains applicable in spite of the non-convexities from binary decision variables. Nonetheless, its main limitation is that the nice results in [25] only hold in the absence of transmission capacity constraints. This paper significantly improves [25] by considering additional security constraints such as line flow limits in the presence of uncertainties, and provides theoretical analysis on the results of the scenario approach.

The main contributions of this paper are threefold. (1) We contribute to the non-convex scenario approach theory by proving salient structural properties of non-convex scenario problems, which extends the classical results for convex scenario problems published in [23]. (2) We formulate c-SCUC, which is later reformulated to scenario-based SCUC (s-SCUC) and solved via the scenario approach. To the best of our knowledge, this paper is the first to solve c-SCUC using the scenario approach while considering critical constraints such as transmission limits. (3) We design efficient algorithms to explore the structural properties of s-SCUC, which enables rigorous guarantees on the optimal solution returned by the scenario approach.

The remainder of this paper is organized as follows. Section II summarizes the key results of the scenario approach for both convex and non-convex problems. Section III proves the structural properties of non-convex scenario problems. Section IV formulates chance-constrained SCUC, which is solved via the scenario approach. Numerical results and discussions are in Section V and VI respectively. Section VII presents the concluding remarks. All proofs are available in the full-length version [26] of this paper.

The notations in this paper are standard. All vectors are in the real field $\mathbb{R}$. We use $\mathbf{1}$ to represent an all-one vector of appropriate size. The transpose of a vector $a$ is $a^\top$. The element-wise multiplication of the same-size vectors $a$ and $b$ is denoted by $a \odot b$. For instance, $[a_1; a_2] \odot [b_1; b_2] = [a_1 b_1; a_2 b_2]$. Sets are in calligraphy fonts, e.g. $\mathcal{S}$. The cardinality of a set $\mathcal{S}$ is $|\mathcal{S}|$. Removal of element $i$ from set $\mathcal{N}$ is represented by $\mathcal{N} - i$. The essential supremum is $\text{ess sup}$.

\footnote{We call it unit commitment instead of SCUC because no security constraints are considered.}
II. INTRODUCTION TO THE SCENARIO APPROACH

This section first provides a brief introduction to chance-constrained optimization. Section II-B presents the main results of the scenario approach for convex problems. Recent progress in the scenario approach for non-convex problems are summarized in Section II-C.

A. Chance-constrained Optimization

Chance-constrained optimization is a major approach for decision making in an uncertain environment. Since its birth in 1950s [27], chance-constrained optimization has been widely studied and successfully applied in various fields [6]. A typical formulation of chance-constrained optimization is presented below.

\[
\begin{align*}
\min_{x} & \quad c^\top x \\
\text{s.t.} & \quad \mathbb{P}_\xi \left( f(x, \xi) \leq 0 \right) \geq 1 - \epsilon \\
& \quad g(x) \leq 0
\end{align*}
\]

We could write (1) in a more compact form by defining \( X_\xi := \{ x \in \mathbb{R}^n : f(x, \xi) \leq 0 \} \) and \( \chi := \{ x \in \mathbb{R}^n : g(x) \leq 0 \} \)

\[
\begin{align*}
\min_{x \in \chi} & \quad c^\top x \\
\text{s.t.} & \quad \mathbb{P}_\xi \left( x \in X_\xi \right) \geq 1 - \epsilon
\end{align*}
\]

Without loss of generality, we assume that the objective is a linear function of decision variables \( x \in \mathbb{R}^d \) [28]. Random vector \( \xi \in \Xi \) denotes the source of uncertainties and \( \Xi \) is the support of \( \xi \). Deterministic constraints (1c) are represented by set \( \chi \) in (2). Constraint (1b) or (2b) is the chance constraint. The chance constraint (2b) requires the inner constraint \( x \in X_\xi \) to be satisfied with probability at least \( 1 - \epsilon \), where the violation probability \( \epsilon \) is typically a small number (e.g. 1%, 5%). In (2b), the set \( X_\xi \) depends on the realization of \( \xi \) and the probability is taken with respect to \( \xi \).

Researchers have proposed many methods to solve chance-constrained optimization problems, e.g. sample average approximation [29], convex approximation [30], and scenario approach [21]–[23]. A detailed review and tutorial on chance-constrained optimization is in [6]. Compared with other methods, the scenario approach has many advantages such as computationally efficient and are applicable for a broad range of optimization problems.
B. The Scenario Approach for Convex Problems

The scenario approach (sometimes referred as scenario approximation) is one of the well-known solutions to chance-constrained optimization, but its strength is not well-understood until recently \[6\]. The scenario approach utilizes \( N \) independent and identically distributed (i.i.d.) scenarios \( \mathcal{N} := \{ \xi^1, \xi^2, \cdots, \xi^N \} \) to convert the chance-constrained program (1) to the scenario problem below:

\[
\text{SP}(\mathcal{N}) : \min_x c^T x \tag{3a}
\]
\[
\text{s.t. } f(x, \xi^i) \leq 0 : \mu^i \tag{3b}
\]
\[
\vdots
\]
\[
f(x, \xi^N) \leq 0 : \mu^N \tag{3c}
\]
\[
g(x) \leq 0 : \lambda \tag{3d}
\]

The scenario problem \( \text{SP}(\mathcal{N}) \) seeks the optimal solution \( x^*_N \) that is feasible for all \( N \) scenarios. The Lagrangian multiplier associated with the \( i \)-th scenario constraint \( f(x, \xi^i) \leq 0 \) is denoted by \( \mu^i \in \mathbb{R}^m \). We can write the scenario problem \( \text{SP}(\mathcal{N}) \) in a similar way with (2) by defining \( \mathcal{X}_i := \{ x \in \mathbb{R}^n : f(x, \xi^i) \leq 0 \} \).

\[
\text{SP}(\mathcal{N}) : \min_{x \in \mathcal{X}} c^T x \tag{4a}
\]
\[
\text{s.t. } x \in \bigcap_{i=1}^N \mathcal{X}_i \tag{4b}
\]

**Definition 1** (Violation Probability). The violation probability of a candidate solution \( x^\circ \) is defined as the probability that \( x^\circ \) is infeasible:

\[
\mathbb{V}(x^\circ) := \mathbb{P}_\xi(x^\circ \notin \mathcal{X}). \tag{5}
\]

The scenario approach theory aims at answering the following sample complexity question: what is the smallest sample size \( N \) such that \( x^*_N \) is feasible (i.e. \( \mathbb{V}(x^*_N) \leq \epsilon \)) to the original chance-constrained program (2)? Reference [22], [23] provide in-depth analysis based on the concept of support scenarios.

**Definition 2** (Support Scenario [22], [23]). Scenario \( \xi^i \) is a support scenario for the scenario problem \( \text{SP}(\mathcal{N}) \) if its removal changes the solution of \( \text{SP}(\mathcal{N}) \).
Let $x^*_N$ and $x^*_{N-i}$ stand for the optimal solution to scenario problems $SP(N)$ and $SP(N-i)$, respectively. Then scenario $\xi^i$ is a support scenario if $c^T x^*_{N-i} < c^T x^*_N$. We use $S(N)$ (S in short) to represent the set of all support scenarios of $SP(N)$.

**Definition 3** (Non-degenerate Scenario Problem [22, 23]). Let $x^*_N$ and $x^*_S$ be the optimal solutions to the scenario problems $SP(N)$ and $SP(S)$, respectively. The scenario problem $SP(N)$ is said to be non-degenerate, if $c^T x^*_N = c^T x^*_S$.

**Assumption 1** (Non-degeneracy [22, 23]). For every $N$, the scenario problem $SP(N)$ is non-degenerate with probability 1 with respect to scenarios $N = \{\xi^1, \xi^2, \cdots, \xi^N\}$.

**Assumption 2** (Feasibility [22]). Every scenario problem $SP(N)$ is feasible, and its feasibility region has a non-empty interior. The optimal solution $x^*_N$ of $SP(N)$ exists.

**Definition 4** (Helly’s Dimension [23]). Helly’s dimension of the scenario problem $SP(N)$ is the least integer $h$ that $h \geq \text{ess sup}_{N \subseteq \Xi} |S(N)|$ holds for any finite $N \geq 1$, where $|S(N)|$ is the number of support scenarios.

Theorem 1 presents one of the most important results in the scenario approach theory, which is based on the non-degeneracy and feasibility assumptions.

**Theorem 1** (Exact Feasibility [22, 23]). Under Assumptions 1 (non-degeneracy) and 2 (feasibility), let $x^*_N$ be the optimal solution to the scenario problem $SP(N)$, it holds that

$$
P^N \left( \forall (x^*_N) > \epsilon \right) \leq \sum_{i=1}^{h-1} \binom{N}{i} \epsilon^i(1-\epsilon)^{N-i}.
$$

The probability $P^N$ is taken with respect to $N$ random scenarios $N = \{\xi^i\}_{i=1}^N$, and $h$ is the Helly’s dimension of $SP(N)$.

Stronger results without the feasibility assumption are in [22, 23]. Based on Theorem 1, the scenario approach answers the sample complexity question in Corollary 1.

**Corollary 1** (Sample Complexity [22, 23]). Under Assumptions 1 (non-degeneracy) and 2 (feasibility), given a violation probability $\epsilon \in (0, 1)$ and a confidence parameter $\beta \in (0, 1)$, if
we choose the smallest number of scenarios \( N \) such that

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

then it holds that

\[
\mathbb{P}^N \left( \forall x^*_N \leq \epsilon \right) \geq 1 - \beta,
\]

where \( x^*_N \) is the optimal solution to \( SP(N) \), and \( h \) is the Helly’s dimension of \( SP(N) \) (0 \( h \leq N \)).

The scenario approach is essentially a randomized algorithm to solve chance-constrained optimization problems. The randomness of the scenario approach comes from drawing i.i.d. scenarios. The confidence parameter \( \beta \) quantifies the risk of failure due to drawing scenarios from a “bad” set. Corollary 1 shows that by choosing a proper number of scenarios, the corresponding optimal solution \( x^*_N \) is feasible (i.e. \( \forall (x^*_N) \leq \epsilon \)) with confidence at least \( 1 - \beta \).

**Assumption 3** (Convexity). The deterministic constraint \( g(x) \leq 0 \) is convex, and the random constraint \( f(x, \xi) \) is convex in \( x \) for every instance of \( \xi \). In other words, the sets \( \chi \) and \( \chi_i \) in (4) are convex.

**Theorem 2** ([21], [23]). Under Assumption 2 and 3 the number of support scenarios \( |S| \) for \( SP(N) \) is at most \( n \). In other words, \( h \leq n \), where \( n \) is the number of decision variables \( x \in \mathbb{R}^n \) and \( h \) is Helly’s dimension.

For convex scenario problems \( SP(N) \), we could replace \( h \) by \( n \) in Theorem 1 and Corollary 1. This leads to the classical results of the scenario approach in [21]–[23].

**Remark 1** (Towards Non-convexity). Theorem 1 and Corollary 1 do not assume convexity of \( f(x, \xi) \) and \( g(x) \). In theory, Theorem 1 and Corollary 1 are applicable for non-convex scenario problems if a feasible non-convex \( SP(N) \) is proved to be non-degenerate with probability 1 (e.g. [25]). In practice, however, the scenario approach was considered not applicable for non-convex problems. Comprehensive analysis are presented in Section II-C.

**C. The Scenario Approach for Non-convex Problems**

The scenario approach was considered not applicable for non-convex problems for the following three reasons: (1) non-convexity causes degeneracy; (2) non-trivial bounds on \( |S| \) may not
exist for non-convex SP(\(\mathcal{N}\)); and (3) it is computationally intractable to find optimal solutions.

First, degeneracy is a common issue for non-convex problems, e.g. the scenario-based SCUC problem in Section IV-C. Since the non-degeneracy assumption lies at the heart of the scenario approach theory, almost all results in the literature are for non-degenerate problems.

Second, it is almost impossible to prove non-trivial and practical bounds on the number of support scenarios \(|\mathcal{S}|\) for non-convex problems. Reference [24] presents one extreme case, in which every scenario is a support scenario thus \(|\mathcal{S}| = N^3\). In addition, a loose bound typically leads to an astronomical sample complexity \(N\), which make the scenario approach unpractical. For instance, loose bounds on \(|\mathcal{S}|\) for scenario-based unit commitment will require \(10^3 \sim 10^4\) times more scenarios than necessary [25].

Furthermore, the most attractive feature of convex optimization is that any local minimum is a global minimum. And there exist a broad family of efficient algorithms that compute global optimal solutions for convex problems. Hence, \(x_{x^*_{\mathcal{N}}}\) in Section II-B refers to the global optimal solution by default. It is worth noting that \(x_{x^*_{\mathcal{N}}}\) is solely determined by the scenario problem SP(\(\mathcal{N}\)) and it is not algorithm-dependent.

For non-convex problems, however, it is often computationally intractable to find global optimal solutions. There are many algorithms that are capable of finding local optimal solutions in a relatively short time. Therefore, it is more reasonable and practical to analyze the characteristics of local solutions for non-convex scenario problems. Algorithm \(\mathcal{A}_\mathcal{N} : \Xi^\mathcal{N} \rightarrow \mathbb{R}^n\) stands for the process of finding solutions to SP(\(\mathcal{N}\)), e.g. primal-dual interior-point method. We use opx\(_\mathcal{A}_\mathcal{N}(\mathcal{N})\) to represent a (possibly suboptimal) solution to SP(\(\mathcal{N}\)) obtained via algorithm \(\mathcal{A}_\mathcal{N}\). The corresponding optimal objective value is denoted by \(\text{opt}_{\mathcal{A}_\mathcal{N}}(\mathcal{N})\). The subscript \(\mathcal{A}\) emphasizes the fact that the solution is algorithm-dependent. And we use SP\(_{\mathcal{A}_\mathcal{N}}(\mathcal{N})\) to represent a scenario problem solved by algorithm \(\mathcal{A}_\mathcal{N}\).

Consequently, the scenario approach was considered not applicable for non-convex problems until very recently. By removing the non-degeneracy assumption and analyzing any feasible solutions of non-convex scenario problems, reference [24] develops a general theory for the scenario approach. This subsection summarizes its key results.

Identical to the convex case in Section II-B, the scenario approach converts (2) to the scenario problem

\[\textbf{3} \text{Using the trivial bound } |\mathcal{S}| \leq N, \text{ Theorems 1 and 3 provide guarantees } P(V(x^*_{\mathcal{N}}) > \epsilon) \leq 1, \text{ which is useless.}\]
problem (4) using $N$ scenarios $\mathcal{N} = \{\xi^1, \xi^2, \ldots, \xi^N\}$ for non-convex problems. The sets $\chi$ and $\mathcal{X}_\xi$ here could be non-convex.

**Definition 5** (Invariant Set). Let $\text{opt}_A(\mathcal{M})$ be the optimal value of $\text{SP}(\mathcal{M})$ found by algorithm $A$ for a scenario problem $\text{SP}(\mathcal{M})$. A set of scenarios $\mathcal{I}$ is an invariant (scenario) set for $\text{SP}_A(\mathcal{N})$ if $\text{opt}_A(\mathcal{I}) = \text{opt}_A(\mathcal{N})$.

The concept of invariant set is an extension of support scenarios for (possibly degenerate) non-convex scenario problems. A trivial invariant set is $\mathcal{I} = \mathcal{N}$. Algorithm $B: \Xi^N \to \mathcal{I}$ represents the process of finding non-trivial invariant sets. Examples of Algorithm $B$ can be found in Section III and Appendix A.

**Theorem 3** (Posterior Guarantees for Non-convex Scenario Problems [24]). Suppose Assumption 2 (feasibility) holds true and $\beta \in (0, 1)$ is given. Algorithm $A$ solves the scenario problem $\text{SP}(\mathcal{N})$ and obtains an optimal solution $\text{opx}_A(\mathcal{N})$. Algorithm $B$ finds an invariant set $\mathcal{I}$ of cardinality $|\mathcal{I}|$. The following probabilistic guarantee holds

$$\mathbb{P}^N\left(\bigvee \left(\text{opx}_A(\mathcal{N})\right) \leq \epsilon(N, |\mathcal{I}|, \beta)\right) \geq 1 - \beta,$$

where the function $\epsilon(k, N, \beta)$ is defined as

$$\epsilon(k, N, \beta) := \begin{cases} 1 & \text{if } k = N, \\ 1 - \left(\frac{\beta^k}{N^k}\right)^\frac{1}{\beta} & \text{otherwise.} \end{cases} \quad (9)$$

**Lemma 1.** The $\epsilon(N, k, \beta)$ function defined in (9) has the following properties: (1) $\epsilon(N, k, \beta)$ is monotonically decreasing in $\beta$; (2) $\epsilon(N, k, \beta)$ is monotonically increasing in $k$; (3) $\epsilon(N, k, \beta)$ is monotonically decreasing in $N$.

In order to achieve an $\epsilon$-level solution with confidence $1 - \beta$, Lemma 1 shows that the least conservative result (i.e. smallest sample complexity $N$) is achieved with the invariant set of minimal cardinality, which is defined as the essential set.

---

$^4$Theorem 3 is a simplified version of the main result in [24], the feasibility assumption 2 is a simplified version of the admissible assumption in [24].
**Definition 6** (Essential Set [23]). A set of scenarios $\mathcal{E} \subseteq \mathcal{N}$ is an essential (scenario) set for $\text{SP}_A(\mathcal{N})$ if

$$\mathcal{E} := \arg \min \{|\mathcal{E}| : \text{opt}_A(\mathcal{E}) = \text{opt}_A(\mathcal{N}), \mathcal{E} \subseteq \mathcal{N}\}.$$  \hspace{1cm} (10)

In other words, $\mathcal{E}$ is an invariant set of minimal cardinality.

One key step in the non-convex scenario approach is designing algorithms to search for essential sets. Section III reveals the structure of general non-convex scenario problems, which lays the cornerstone for algorithms to obtain essential sets. Section III also gives one example of designing more efficient algorithms by exploiting the structural properties of specific problems.

### III. Structural Properties of General Scenario Problems

Searching for essential sets is an important step in the non-convex scenario approach. However, the only known general algorithm to obtain essential sets is enumerating all $2^\mathcal{N}$ possibilities by solving $2^\mathcal{N}$ non-convex problems. This implies that searching for essential sets is in general computationally prohibitive. Section III-A first demonstrates the structural properties for general non-convex scenario problems, and proves a few special cases that finding essential sets is relatively easier. Section III-B reveals the connection between non-convex and convex scenario problems. Motivated by the structure of security-constrained unit commitment, Section III-C illustrates an efficient algorithm to track down essential sets for two-stage scenario problems.

#### A. Non-convex Scenario Problems

Instead of solving $2^\mathcal{N}$ non-convex problems to obtain essential sets, there are two ideas to track down invariant sets with small cardinalities (not necessarily essential): (1) removing each scenario and checking if the objective changes, this idea leads to the definition of support sets; (2) removing scenarios one by one, until the scenario set cannot be further reduced, this leads to the definition of irreducible set.

**Definition 7** (Support Scenario of $\text{SP}_A(\mathcal{N})$). Scenario $\xi^i \in \mathcal{N}$ is a support scenario for the scenario problem $\text{SP}_A(\mathcal{N})$ if its removal changes the solution $\text{opt}_A(\mathcal{N})$ of $\text{SP}_A(\mathcal{N})$. The set of support scenarios (support set in short) is denoted by $\mathcal{S}_A$. 

November 12, 2019 DRAFT
Definition 8 (Irreducible Set). A scenario set $\mathcal{R} \subseteq \mathcal{N}$ for $\text{SP}_A(\mathcal{N})$ is irreducible, if (1) it is invariant, i.e. $\text{opt}_A(\mathcal{R}) = \text{opt}_A(\mathcal{N})$; and (2) $\text{opt}_A(\mathcal{R} - s) < \text{opt}_A(\mathcal{R}) = \text{opt}_A(\mathcal{N})$ for any $s \in \mathcal{R}$.

Assumption 4 (Monotonicity). Let $A : \Xi^\mathcal{N} \rightarrow \mathbb{R}^n$ be an algorithm to obtain the optimal solution of a scenario problem $\text{SP}(\mathcal{N})$, whose optimal objective value is represented by $\text{opt}_A(\mathcal{N})$. We assume that the algorithm $A$ always satisfies $\text{opt}_A(\mathcal{M}) \leq \text{opt}_A(\mathcal{N})$ if $\mathcal{M} \subseteq \mathcal{N}$.

Assumption 4 is indeed a weak assumption. Considering two scenario problems $\text{SP}(\mathcal{N})$ and $\text{SP}(\mathcal{M})$ with $\mathcal{M} \subseteq \mathcal{N}$. Because the optimal solution to $\text{SP}(\mathcal{N})$ will be always feasible to $\text{SP}(\mathcal{M})$, algorithm $A$ could use $\text{opt}_A(\mathcal{N})$ as a starting point and obtain solution $\text{opt}_A(\mathcal{M})$ that is not worse than $\text{opt}_A(\mathcal{N})$.

Lemma 2 (Modified Lemma 2.10 of [23]). Suppose algorithm $A$ satisfies Assumption 4. Let $\mathcal{I}$ be any invariant set for a (possibly non-convex) scenario problem $\text{SP}_A(\mathcal{N})$ and $S$ stands for its support set, then $S \subseteq \mathcal{I}$. Since any essential set $\mathcal{E}$ or irreducible set $\mathcal{R}$ is also invariant, then $S \subseteq \mathcal{E}$ and $S \subseteq \mathcal{R}$.

Lemma 2 reveals the key relationship among the support set, essential and irreducible sets, and it lays the foundation of more important observations in Corollary 2. Lemma 2 is a generalized version of Lemma 2.10 in [23], which proved similar results for convex scenario problems. The importance of Lemma 2 is to show that the key assumption for such structural properties is the monotonicity of algorithm $A$, instead of convexity (Assumption 3 in [23]).

For general (non-convex) scenario problems, the support set, essential set and irreducible set are different. Under certain circumstances, these three concepts are interchangeable. Such circumstances are depicted by an extended definition of non-degeneracy for non-convex scenario problems.

Definition 9 (Non-degeneracy of $\text{SP}_A(\mathcal{N})$). For a general scenario problem $\text{SP}_A(\mathcal{N})$, let $\mathcal{N}$ stand for the set of all $N$ scenarios and $S$ denote the support (scenario) set. The scenario problem $\text{SP}_A(\mathcal{N})$ is said to be non-degenerate, if $\text{opt}_A(\mathcal{N}) = \text{opt}_A(S)$.

Corollary 2. Consider a (possibly non-convex) scenario problem $\text{SP}_A(\mathcal{N})$ and an algorithm $A$ satisfying Assumption 4. If $\text{SP}_A(\mathcal{N})$ is non-degenerate, then (1) it has a unique essential set $\mathcal{E} = S$; and (2) it has a unique irreducible set $\mathcal{R} = S$. 

November 12, 2019 DRAFT
Corollary 3. Consider a (possibly non-convex) scenario problem $SP_A(N)$ and an algorithm $A$ satisfying Assumption 4. The following three statements are equivalent with each other: (1) $SP_A(N)$ is non-degenerate; (2) $SP_A(N)$ has a unique irreducible set $R$; and (3) $SP_A(N)$ has a unique essential set $E$.

Corollaries 2 and 3 provide key insights in designing an efficient algorithm $B$. For non-convex problems, even if Assumption 1 does not always hold, $SP_A(N)$ might be non-degenerate in many instances (e.g. s-SCUC is non-degenerate in 192 out of 200 instances in Section V-D). For those non-degenerate scenario problems, Corollary 2 and 3 show that we are able to find the essential set by solving only $N$ instead of $2^N$ non-convex problems. Section III-C shows that the computational burden to obtain essential sets can be further reduced by exploiting the structure of specific problems.

**Remark 2 (Finding Essential Sets for Non-degenerate Problems).** When a scenario problem is non-degenerate, we can obtain the (unique) essential set by searching for the support set or irreducible set (Corollary 2). Algorithms of finding an irreducible set (Algorithm 3 in Appendix A) or the support set (e.g. Algorithm 1) are based on definition. More discussions on finding the support set are in Remark 3.

**B. Convex Scenario Problems**

For convex scenario problems $SP(N)$, any local minimum is a global minimum. And a broad range of algorithms to look for global optimal solutions exist. In the convex setting, we assume any algorithm $A$ returns global optimal solutions to $SP(N)$ by default. In Section II-B and III-B, we replace $\text{opx}_A(N)$ and $\text{opt}_A(N)$ by $x^*_N$ and $c^T x^*_N$, respectively. We also remove subscripts $A$ since the definition of support set, invariant set and essential set for convex problems are no longer algorithm-dependent.

**Lemma 3 (Monotonicity).** Let $x^*_N$ and $x^*_M$ stand for the global optimal solution to the (convex) scenario problems $SP(N)$ and $SP(M)$, respectively. Then $c^T x^*_M \leq c^T x^*_N$ if $M \subseteq N$.

Because $x^*_N$ is always feasible to $SP(M)$ and $x^*_M$ is globally optimal, it is obvious that $c^T x^*_M \leq c^T x^*_N$. Lemma 3 shows that any algorithm obtaining global optimal solutions will automatically satisfy Assumption 4. Therefore, all results in Section III-A hold for convex scenario problems.
It is worth mentioning that similar results for convex problems were first proved in [23]. Section III-A can be regarded as an extension of classical results in [23] towards non-convex scenario problems.

**Remark 3 (Finding Support Scenarios For Convex Problems).** The first algorithm of searching for support scenarios (for both convex and non-convex scenario problems, Algorithm 2 in Appendix A) is based on definition, i.e. checking if the removal of a scenario changes the optimal solution. Algorithm 2 requires solving $N$ scenario problems. In many cases (especially in power system applications, e.g. [31]), it is observed that the support scenarios are only a small subset of all $N$ scenarios, i.e. $|S| \ll |N|$. This observation indicates the dual solution $\mu_1, \mu_2, \ldots, \mu_N$ to $SP(N)$ is often sparse. Lemma 4 formalizes this observation and provides an approach to narrow down the range of searching for support scenarios. Built upon Lemma 4, Algorithm 1 only requires solving $\sim |S|$ scenario problems, which is much more efficient than Algorithm 2 since $|S| \ll |N|$.

**Lemma 4.** Consider a non-degenerate and convex scenario problem $SP(N)$ which has at least one strictly feasible solution. If $\xi^i$ is a support scenario ($i \in S$), then $\|\mu^{i,*}\| > 0$, where $\mu^{i,*} \in \mathbb{R}^m$ is the optimal dual solution of $SP(N)$. In other words, let $M := \{i \in N : \|\mu^{i,*}\| > 0\}$, then $S \subseteq M$.

**Algorithm 1** Finding Support Scenarios Using Dual Variables

1: Compute the primal and dual solutions $x^*_N$ and $\mu^{i,*}$ ($i = 1, 2, \ldots, N$) by solving $SP(N)$
2: Let $M = \{i \in N : \|\mu^{i,*}\| > 0\}$. Set $S \leftarrow \emptyset$.
3: for $i \in M$ do
4: Solve $SP_{M-i}$ and compute $x^*_M$.
5: if $c^T x^*_{M-i} < c^T x^*_M$ (i.e. $c^T x^*_M$) then
6: $S \leftarrow S + i$
7: end if
8: end for

In many cases, Algorithm 1 only needs to solve the dual problem of $SP(N)$, it may not be necessary to solve the primal solution $x^*_N$. We use $x^*_N$ in Algorithm 1 mainly for the purpose
of notation simplicity.

C. Two-stage Scenario Problems

Section III-A shows that searching for essential sets can be relatively easier when a scenario problem is non-degenerate. However, finding a support set or irreducible set still requires solving \( N \) non-convex problems. Motivated by SCUC, we show that more efficient algorithms are possible by exploiting the structure of specific problems. We study the following two-stage scenario problem in this subsection.

\[
\begin{align*}
\min_{y \in \mathcal{Y}} & \quad c_1^T y + \min_{x \in \mathcal{X}, (x,y) \in \mathcal{H}} c_2^T x \\
\text{s.t.} & \quad x \in \bigcap_{i=1}^N \mathcal{U}_i
\end{align*}
\]

(11a)

Constraints on the first-stage variables \( y \) and the second-stage variables \( x \) are denoted by \( y \in \mathcal{Y} \) and \( x \in \mathcal{X} \), respectively. Constraint \( (x,y) \in \mathcal{H} \) represents the constraints coupling variables \( x \) and \( y \) in both stages. Set \( \mathcal{U}_i \) stands for the constraints corresponding to the \( i \)th scenario \( \xi_i \).

Problem (11) is an abstract form of s-SCUC in Section IV. Two key features of the two-stage scenario problem are: (1) the non-convexity only comes from constraints \( y \in \mathcal{Y} \) (e.g. binary variables in SCUC), all other constraints \( \mathcal{X}, \mathcal{H}, \mathcal{U}_i \) are convex; (2) uncertainties only exist in the second stage.

Let \( (x^*, y^*) \) be a (possibly local) optimal solution that algorithm \( \mathcal{A} \) returns. Given \( y = y^* \), the second stage problem is convex by setting:

\[
\begin{align*}
\min_{x \in \mathcal{X}, (x,y^*) \in \mathcal{H}} & \quad c_2^T x \\
\text{s.t.} & \quad x \in \bigcap_{i=1}^N \mathcal{U}_i
\end{align*}
\]

(12a)

**Lemma 5.** (1) Let \( \hat{S} \) represent the set of support scenarios of (12) and \( S \) denote the support set for the two-stage problem (11), then \( \hat{S} \subseteq S \); (2) If \( \hat{S} \) is invariant for (11), i.e. \( \text{opt}_{\mathcal{A}}(\hat{S}) = \text{opt}_{\mathcal{A}}(\mathcal{N}) \), then the two-stage scenario problem \( \text{SP}_{\mathcal{A}}(\mathcal{N}) \) is non-degenerate.

Corollaries 2 and 3 demonstrate many nice properties of non-degenerate scenario problems. Lemma 5 gives a criteria of checking if the two-stage problem (e.g. s-SCUC) is non-degenerate. This lemma lays the foundation of Algorithm 4 to search for essential sets of (11). The main
idea of Algorithm 4 is to first find the support scenarios of the second-stage problem (12), then verify if SP(\mathcal{N}) is degenerate using Lemma 5. In Section V-D, it turns out that s-SCUC is non-degenerate in 96\% of cases, thus Algorithm 4 could obtain essential sets of s-SCUC (in Section V-D) in a much shorter time.

IV. SECURITY-CONSTRAINED UNIT COMMITMENT WITH PROBABILISTIC GUARANTEES

A. Nomenclature

The number of loads, generators, wind farms, transmission lines, contingencies, and snapshots are denoted by \( n_d, n_g, n_w, n_l, n_k \) and \( n_t \), respectively.

- \( k \in \{ 0, 1, \cdots, n_k \} \) contingency index
- \( t \in \{ 0, 1, \cdots, n_t \} \) time (snapshot) index
- \( \iota \in \{ t+1, \cdots, n_t \} \) additional time (snapshot) index in constraints (13j) and (13k)

Binary decision variables (at time \( t \)):

- \( z^t \in \{ 0, 1 \}^{n_g} \) generator on/off states (commitment)
- \( u^t \in \{ 0, 1 \}^{n_g} \) generator \( i \) is on if \( u^t_i = 1 \)
- \( v^t \in \{ 0, 1 \}^{n_g} \) generator \( i \) is off if \( v^t_i = 1 \)

Continuous decision variables (at time \( t, \) contingency \( k \)):

- \( g^{t,k} \in \mathbb{R}^{n_g} \) generation output
- \( r^t \in \mathbb{R}^{n_g} \) reserve

Parameters and constants:

- \( a^k \in \{ 0, 1 \}^{n_g} \) generator availability in contingency \( k \)
- \( \alpha_k \in \mathbb{R}_+ \) weight of contingency \( k \)
- \( c_g \in \mathbb{R}^{n_g} \) generation costs
- \( c_z \in \mathbb{R}^{n_g} \) no load cost
- \( c_r \in \mathbb{R}^{n_g} \) reserve costs
- \( c_u \in \mathbb{R}^{n_g} \) startup cost
- \( c_v \in \mathbb{R}^{n_g} \) shutdown cost
- \( \hat{d}^t \in \mathbb{R}^{n_d} \) load forecast (time \( t \))
- \( \tilde{d}^t \in \mathbb{R}^{n_d} \) load forecast error (time \( t \))
- \( \hat{w}^t \in \mathbb{R}^{n_w} \) wind forecast (time \( t \))
\[ \tilde{w}^t \in \mathbb{R}^{n_w} \quad \text{wind forecast error (time } t) \]
\[ \bar{g} \in \mathbb{R}^{n_g} \quad \text{generation upper bounds} \]
\[ \underline{g} \in \mathbb{R}^{n_g} \quad \text{generation lower bounds} \]
\[ \bar{\gamma} \in \mathbb{R}^{n_g} \quad \text{ramping upper bounds} \]
\[ \underline{\gamma} \in \mathbb{R}^{n_g} \quad \text{ramping lower bounds} \]
\[ u_i \in \mathbb{R}_+ \quad \text{minimum on time for generator } i \]
\[ v_i \in \mathbb{R}_+ \quad \text{minimum off time for generator } i \]

**B. Deterministic Security-constrained Unit Commitment**

Deterministic security-constrained unit commitment (d-SCUC) \([13]\) seeks optimal commitment and startup/shutdown decisions \((z^t, u^t, v^t)\), generation and reserve schedules \((g^{t,k}, r^t)\) for a horizon of time steps, typically \(24 \sim 36\) hours. The d-SCUC problem is being solved as a crucial part of the day-ahead market operation. Security constraints ensures the reliability of the power system after an unexpected event occurs.

\[
\begin{align*}
\min_{z,u,v,g,r} \quad & \sum_{t=1}^{n_t} \left( c_z^T z^t + c_u^T u^t + c_v^T v^t + c_r^T r^t + \sum_{k=0}^{n_k} \alpha_k c_g^T g^{t,k} \right) \\
\text{s.t.} \quad & 1^T g^{t,k} + 1^T \tilde{w}^t \geq 1^T d^t \\
& f \leq H_g^{t,k} g^{t,k} + H_w^{t,k} w^{t,k} - H_d^{t,k} d^{t,k} \leq \bar{f} \\
& a^k \circ \underline{\gamma} \leq g^{t,k} - g^{t-1,k} \leq a^k \circ \bar{\gamma} \\
& a^k \circ (g^{t,0} - r^t) \leq g^{t,k} \leq a^k \circ (g^{t,0} + r^t) \\
& k \in [0, n_k], t \in [1, n_t] \\
& g \circ z^t \leq g^{t,0} \leq \bar{g} \circ z^t \\
& g \circ z^t \leq g^{t,0} - r^t \leq g^{t,0} + r^t \leq \bar{g} \circ z^t \\
& z^{t-1} - z^t + u^t \geq 0 \\
& z^t - z^{t-1} + v^t \geq 0 \\
& t \in [1, n_t] \\
& z_i^t - z_{i-1}^t \leq z_i^t, \quad i \in [1, n_g] \\
& t \in [t + 1, \min\{t + u_i - 1, n_t\}], t \in [2, n_t]
\end{align*}
\]
\[ z_{i-1}^t - z_i^t \leq 1 - z_i^t, \quad i \in [1, n_g] \quad (13k) \]

\[ \iota \in [t + 1, \min\{t + v_i - 1, n_t\}], t \in [2, n_t] \]

The objective of (13) is to minimize total operation costs, including no-load costs \( c^T_z z^t \), startup costs \( c^T_u u^t \), shutdown costs \( c^T_v v^t \), generation costs \( c^T_g g^{t,k} \) and reserve costs \( c^T_r r^t \). Security constraints ensure: enough supply to meet demand (13b), transmission line flow within limits (13c), generation levels within ramping limits (13d) and capacity limits (13f) in any contingency \( k \). Constraints (13e) and (13g) are about the relationship between generation and reserve in any contingency \( k \). Constraints (13h)-(13i) are the logistic constraints about commitment status, startup and shutdown decisions. Minimum on/off time constraints for all generators are in (13j)-(13k). Constraints (13d)-(13g) also guarantee the consistency of generation levels \( g^{t,k} \) with commitment decisions \( z^t \) and generator availability \( a_k \) in contingency \( k \) [25].

The d-SCUC formulation utilizes the expected wind generation and load forecast, it does not take the uncertainties from wind and load into consideration. We propose an improved formulation of d-SCUC using chance constraints, which explicitly guarantee the system security with a tunable level of risk \( \epsilon \) with respect to uncertainties.

\[
\mathbb{P}_{\tilde{w} \times \tilde{d}} \left( 1^T g^{t,k} + 1^T (\hat{w}^t + \tilde{w}^t) \geq 1^T (\hat{d}^k + \tilde{d}^t) \right), \quad (14a) \\
\frac{f}{f} \leq H_{g^{t,k}} + H_{\tilde{w}} (\hat{w}^t + \tilde{w}^t) - \quad (14b) \\
H_{d^{t,k}} (\hat{d}^k + \tilde{d}^t) \leq \frac{f}{f}, \quad \iota \in [0, n_k], t \in [1, n_t] \right) \geq 1 - \epsilon
\]

The formulation of chance-constrained Security-constrained Unit Commitment (c-SCUC) is presented below. Instead of using expected load \( \hat{d}^t \) as in (13), we consider loads \( d^t \) as forecast \( \hat{d}^t \) plus a random forecast error \( \tilde{d}^t \) (i.e. \( d^t = \hat{d}^t + \tilde{d}^t \)).

\[
\min (13a) \quad s.t. (13b)-(13k), (14a)-(14b)
\]

Comparing with d-SCUC, the only difference of c-SCUC is the addition of the chance constraint (14a). The chance constraint guarantees there will be enough supply to meet the net demand with probability no less than \( 1 - \epsilon \).
To reveal the structure of c-SCUC, we define the sets below:

\[ B := \{ (z, u, v) : (13h), (13i), (13j), (13k) \} \]  

\[ C := \{ (g, r) : (13b), (13c), (13d), (13e) \} \]  

\[ H := \{ (z, g, r) : (13f), (13g) \} \]  

\[ U := \{ (g) : (14a), (14b) \} \]

Then c-SCUC can be succinctly represented as:

\[
\begin{align*}
\min_{z,u,v,g,r} & \quad (13a) \\
\text{s.t.} & \quad (z, u, v) \in B \\
& \quad (g, r) \in C, \ (z, g, r) \in H \\
& \quad \mathbb{P}(g \in U) \geq 1 - \epsilon
\end{align*}
\]

Sets \( B \) and \( C \) stand for the deterministic constraints for binary and continuous variables, respectively. Set \( H \) represents the hybrid constraints related with both continuous and binary variables. Set \( U \) denotes all constraints related with uncertainties. Using the scenario approach, c-SCUC is converted to the scenario-based SCUC (s-SCUC) problem below:

\[
\begin{align*}
\min_{(z,u,v) \in B} & \quad \sum_{t=1}^{n_t} \left( c_z^T z^t + c_u^T u^t + c_v^T v^t \right) + \\
& \quad \min_{(z,g,r) \in H} \sum_{t=1}^{n_t} \left( c_r^T r^t + \sum_{k=0}^{n_k} \alpha_k c_g^T g^{t,k} \right) \\
\text{s.t.} & \quad (g, r) \in C \\
& \quad g \in \cap_{i=1}^{N} U_i
\end{align*}
\]

Remark 4 (Structural Properties of SCUC). SCUC is a two-stage optimization problem by nature, it has the following nice properties. Firstly, the non-convexity only exists in the first stage, i.e. \( y \in Y \). Given a first-stage solution \( y \), the second stage is a simple linear program. Secondly, uncertainties come from renewables in the operation stage (only in the second stage). Based on the nice structural properties above, Section III-C shows that we are able to track down essential sets by solving two MILPs and \( \sim |S| \) linear programs.
C. Degeneracy of s-SCUC

This section presents an example to show that s-SCUC could be degenerate in many cases, which violates Assumption I. Therefore almost all results of the classical scenario approach are not applicable. For s-SCUC, theoretical guarantees are only possible through the non-convex scenario approach in Section II-C.

We use a 3-bus system to illustrate the degeneracy of s-SCUC. Configurations of the 3-bus system are in [26]. In order to visualize the feasible region of s-SCUC, we simplify the problem by (1) only considering one snapshot \( n_t = 1 \) and ignoring initial status (thus no \( u, v \) variables); (2) removing reserve constraints (no \( r \) variables). By doing this, there are only four decision variables left: \( z_1, z_2, g_1, g_2 \). The on/off states \( z_1, z_2 \) can be inferred from values of \( g_1 \) and \( g_2 \), therefore the feasible region of the simplified s-SCUC can be visualized on the \((g_1, g_2)\)-plane.

Using Definition 3 showing the degeneracy of s-SCUC includes three steps: (1) obtaining the optimal solution to \( \text{SP}(\mathcal{N}) \); (2) finding all support scenarios \( S \) of \( \text{SP}(\mathcal{N}) \); and (3) checking if the optimal solution of \( \text{SP}(\mathcal{N}) \) is the same as \( \text{SP}(S) \). Fig. 1a first visualizes constraints \( B_0 \sim B_3 \), which represents the region of 4 possible generator on/off status (e.g. \( B_1 : z_1 = 1, z_2 = 0 \), \( B_3 : z_1 = 1, z_2 = 1 \)). The black solid lines denote constraints \( (13b) \), \( (13c) \) and \( (13f) \) using forecast values (d-SCUC). The red, yellow and purple dotted lines are three sets \( (U_1, U_2, U_3) \) of constraints corresponding to three scenarios. Given the setting that generator 1 is much cheaper.

![Fig. 1: An illustrative example that s-SCUC is degenerate (3-bus system)](image)
than generator 2, we can easily eyeball the optimal solution with all constraints presented, marked by the red *. Next, we observe that removing scenario 1 (\(U_1\), red lines) changes the optimal solution, while removing scenario scenario 2 (\(U_2\), yellow lines) or scenario 3 (\(U_3\), purple lines) makes no difference. Thus scenario 1 is the only support scenario. Finally, we examine the scenario problem with only support scenarios presented. Fig. 1b shows that the optimal solution becomes the red \(\Diamond\) with only scenario 1, which is clearly different than the optimal solution in Fig. 1a. Hence, s-SCUC is a degenerate problem.

V. Case Study

A. Settings of the 118-bus System

Numerical simulations were conducted on a modified 118-bus, 184-line, 54-generator, 24-hour system [32]. Most settings are identical as [32], except 5 wind farms are added to the system as in [33]. The s-SCUC problems were solved using 64 GB memory on the Hera server (hera.ece.tamu.edu), provided by Texas A&M University. The mathematical models for s-SCUC was formulated using YALMIP [34] on Matlab R2019a and solved using Gurobi v8.1 [35].

After obtaining a solution \(\text{opx}_A(\mathcal{N})\) to s-SCUC, Theorem 3 provides an upper bound \(\epsilon(N, |\mathcal{I}|, \beta)\) on the actual violation probability \(\mathbb{V}(\text{opx}_A(\mathcal{N}))\). The theoretical guarantee \(\epsilon(N, |\mathcal{I}|, \beta)\) is referred as posterior \(\epsilon\) in the numerical results. The actual violation probability \(\mathbb{V}(\text{opx}_A(\mathcal{N}))\) is estimated by the out-of-sample violation probability \(\hat{\epsilon}\), using an independent set of \(10^6\) scenarios.

To quantify the randomness of the scenario approach, for each sample complexity \(N = 100, 200, \cdots, 1000\), we solve the corresponding s-SCUC problems on 10 independent sets of scenarios (i.e. 10 Monte-Carlo simulations). Results in both Fig. 2 and 3 show the average, maximum and minimum values in 10 Monte-Carlo simulations.

B. Cost vs Security: a trade-off

Fig. 2 shows the out-of-sample violation probability \(\hat{\epsilon}\) and objective value (total cost). The shadowed area shows the max-min values in 10 Monte-Carlo simulations, and the solid line is the average value of 10 independent simulations. It is shown that the system risk level (violation probability) is reduced by \(83\%\) (from \(\sim 30\%\) to \(\sim 5\%\)) by \(\sim 1.1\%\) increase in total system costs. Similar observations were found in [6], [25], [31].
C. Violation Probability

Fig. 3 presents the out-of-sample violation probability $\hat{\epsilon}$ and theoretical guarantees (posterior $\epsilon$ provided by Theorem 3). Since the cardinality of essential sets differ for each scenario problem (Fig. 4), the posterior guarantee $\epsilon$ is a band instead of a line. As illustrated in Fig. 5, the actual violation probability (approximated by $\hat{\epsilon}$) is bounded by the theoretical guarantees. This verifies the correctness of Theorem 3. The conservative ratio is $2 \sim 4$ (e.g. when out-of-sample $\hat{\epsilon}$ is $\sim 5\%$, Theorem 3 gives an upper bound $10\% \sim 20\%$).

D. Searching for Essential Sets for s-SCUC

s-SCUC was observed to be non-degenerate in 192 out of 200 simulations\(^5\). In other words, in 96% cases, we are able to find an essential set by solving $5 \sim 35$ linear programs and 2 mixed integer linear programs. It takes from 4934 seconds ($N = 100$) to 6847 seconds ($N = 1000$) to solve one MILP (s-SCUC). When searching for support scenarios for the second-stage problem (a linear program), it takes $281 \sim 388$ seconds to solve one LP. For those 8 out of 200 simulations,

\(^5\)We conducted 10 simulation for 10 different sample complexities ($100, 200, \cdots, 1000$) under two different settings: with/without $N - 1$ contingencies, both include transmission constraints.
it takes an extra 20 hours to find an irreducible set using Algorithm 3. This computation time can be greatly reduced by tricks such as choosing appropriate starting points.

VI. DISCUSSIONS

A. Cardinality of Essential Sets

Fig. 4 compares the cardinalities of essential sets for three cases: (a) c-SCUC with $N - 1$ contingencies but without transmission constraints, results of case (a) are obtained from [25]; (b) c-SCUC with transmission constraints but without $N - 1$ contingencies; and (c) c-SCUC with both transmission constraints and $N - 1$ contingencies.

Case (a) is the simplest, in [25] we show that the scenario problem for unit commitment satisfies the non-degeneracy assumption and the cardinality of essential sets is bounded by the number of snapshots $n_t$, i.e. $|S| \leq n_t = 24$ in Fig. 4. Case (b) and (c) include transmission capacity constraints. As demonstrated in Section IV-C s-SCUC could be degenerate with transmission constraints. Theoretically speaking, the cardinality of essential sets might be

\[^{6}\text{For example, when removing scenarios } s \text{ and } t \text{ consecutively in Algorithm 3, the solution } \text{opt}_s(N-s) \text{ is feasible to SP}(N-s-t) \text{ thus can serve as a good starting point.}\]
unbounded for non-convex problems. As observed in Fig. 4, the cardinality of essential sets (30 ∼ 40 in case 2, 0 ∼ 10 in case 3) is greatly smaller than the number of decision variables (e.g. about 4000 binary variables and around 75000 continuous decision variables). This observation implies that the number of scenarios $N$ required could be much smaller than expected.

Another interesting observation is that including $N - 1$ contingency constraints reduces $|\mathcal{E}|$. This observation has two implications. First, $N - 1$ contingency constraints not only protect the system from unexpected device failures, they also help reduce the impacts of uncertainties from renewables. Second, including $N - 1$ contingency constraints could help reduce sample complexity. Similar with the observations in [31], this observation indicates that the scenario approach might be of practical use.

B. From Posterior to Prior Guarantees

Theorem 3 gives posterior guarantees on the quality of solutions, namely, we calculate $\epsilon(N, k, \beta)$ after obtaining the solution $\text{opx}(\mathcal{N})$. Lemma 1 proves that the $\epsilon(N, k, \beta)$ function in (9) is monotone in $N$ and $k$. This implies that we can obtain prior guarantees. In other words, if the cardinality of essential sets is proved to be at most $h$ ($|\mathcal{E}| \leq h$), then we can find the
smallest $\tilde{N}$ such that
\[
\epsilon \geq 1 - \left( \frac{\beta}{\tilde{N} \binom{\tilde{N}}{h}} \right)^{\frac{1}{N-h}}
\] (16)
holds for given $\epsilon$ and $\beta$. Then the solution $\text{opx}_h(\mathcal{N})$ to the scenario problem using $\tilde{N}$ scenarios has the guarantee $\Pr(\mathcal{V}(\text{opx}_h(\mathcal{N})) \leq \epsilon) \geq 1 - \beta$. This prior guarantee holds before solving the scenario problem with $\tilde{N}$ scenarios. If a rigorous bound $h$ on $|\mathcal{E}|$ can be proved, then there is no need to numerically search for essential sets. This is particularly attractive compared with posterior guarantees.

VII. CONCLUDING REMARKS

This paper solves chance-constrained SCUC via the scenario approach and obtains rigorous theoretical guarantees on the solution. We demonstrate the structural properties of (possibly non-convex) general scenario problems. To obtain the tightest theoretical guarantees for chance-constrained SCUC, we design efficient algorithms to search for essential sets by exploiting the salient structures of SCUC. Numerical results on an IEEE benchmark system show that the essential scenario set is only a small subset of all scenarios. This implies that we can obtain relatively robust solutions (i.e. small $\epsilon$) using only a moderate number of scenarios. Furthermore, we observe that some power engineering practices (e.g. $N - 1$ criteria) can help us reduce the number of scenarios needed while maintaining the same level of risk.

Future work includes reducing conservativeness by improving the complexity bound in Theorem 3 and investigating the performance of the (non-convex) scenario approach on larger-scale realistic systems.
Algorithm 2 Find the Support Set $\mathcal{S}$ of $\text{SP}(\mathcal{N})$

1: Compute $x_N^*$ by solving $\text{SP}(\mathcal{N})$.
2: Set $\mathcal{S} \leftarrow \emptyset$.
3: \textbf{for} $i \in \mathcal{N}$ \textbf{do}
4: \hspace{1em} Solve the scenario problem $\text{SP}_{\mathcal{N}-i}$ and compute $x_{\mathcal{N}-i}^*$.
5: \hspace{1em} \textbf{if} $c^t x_{\mathcal{N}-i}^* < c^t x_N^*$ \textbf{then}
6: \hspace{2em} $\mathcal{S} \leftarrow \mathcal{S} + i$.
7: \hspace{1em} \textbf{end if}
8: \textbf{end for}

Algorithm 3 Find an Irreducible Set $\mathcal{I}$ of $\text{SP}_\lambda(\mathcal{N})$

1: Compute $\text{opx}_\lambda(\mathcal{N})$ by solving $\text{SP}_\lambda(\mathcal{N})$. Set $\mathcal{I} \leftarrow \mathcal{N}$.
2: \textbf{for} $i \in \mathcal{N}$ \textbf{do}
3: \hspace{1em} Compute $\text{opx}_\lambda(\mathcal{I}-i)$ by solving $\text{SP}(\mathcal{I}-i)$.
4: \hspace{1em} \textbf{if} $\text{opt}_\lambda(\mathcal{I}-i) = \text{opt}_\lambda(\mathcal{N})$ \textbf{then}
5: \hspace{2em} $\mathcal{I} \leftarrow \mathcal{I} - i$.
6: \hspace{1em} \textbf{end if}
7: \textbf{end for}

Algorithm 4 For the two-stage scenario problem (11)

1: Solve $\text{SP}_\lambda(\mathcal{N})$ and compute the solution $(x^*, y^*)$.
2: Fix $y = y^*$, find support scenarios $\mathcal{S}$ of the second-stage problem (12), e.g. using Algorithm 1.
3: \textbf{if} $\text{opt}_\lambda(\mathcal{S}) = \text{opt}_\lambda(\mathcal{N})$ \textbf{then}
4: \hspace{1em} $\text{SP}_\lambda(\mathcal{N})$ is non-degenerate and $\mathcal{S}$ is the essential set.
5: \textbf{else}
6: \hspace{1em} $\text{SP}_\lambda(\mathcal{N})$ is degenerate, the best we can find is an irreducible set, e.g. using Algorithm 3.
7: \textbf{end if}
A. Proofs

Proof of Lemma 1. Monotonicity in $\beta$ is obvious. To prove 2), we show that $\ln\left(\frac{1 - \epsilon(N,k,\beta)}{1 - \epsilon(N,k+1,\beta)}\right) \geq 0$ for fixed values of $(N, \beta)$. For simplicity, we use $\epsilon(k)$ to represent $\epsilon(N, k, \beta)$.

\[
(N - k - 1) \ln\left(\frac{1 - \epsilon(k)}{1 - \epsilon(k + 1)}\right) = \frac{1}{N - k} \ln\left(\frac{N}{\beta}\right) + \ln\left(\frac{N - k}{k}\right)
\]

Clearly, $\ln\left(\frac{1 - \epsilon(k)}{1 - \epsilon(k + 1)}\right) \geq 0$ if $N \geq 2k$. We now show that it also holds for the case of $N \leq 2k$.

\[
(N - k - 1) \ln\left(\frac{1 - \epsilon(k)}{1 - \epsilon(k + 1)}\right) = \frac{1}{N - k} \ln\left(\frac{N}{\beta}\right) + \ln\left(\frac{N - k}{k}\right)
\]

\[
\geq \frac{1}{N - k} \ln\left(\frac{N}{N - k}\right) = \ln\left(\frac{N}{k}\right) \geq 0
\]

The last line uses the well-known lower bound on binomial coefficients $\binom{N}{k} \geq (\frac{N}{k})^k$ and the fact that $\beta \in (0, 1)$ and $1 \leq k \leq N$.

Similarly, we prove 3) by showing $\ln\left(\frac{1 - \epsilon(N+1,k,\beta)}{1 - \epsilon(N,k,\beta)}\right) \geq 0$ for fixed values of $(k, \beta)$. It is easy to verify this is true for the cases $N = k$ and $N = k+1$. The remainder of the proof shows that this is also true for the case $N > k+1$. For simplicity, we show that $(N-k+1)(N-k) \ln\left(\frac{1 - \epsilon(N+1,k,\beta)}{1 - \epsilon(N,k,\beta)}\right) \geq 0$.

\[
(N - k + 1)(N - k) \ln\left(\frac{1 - \epsilon(N+1,k,\beta)}{1 - \epsilon(N,k,\beta)}\right)
\]

\[
= (N - k) \ln\left(\frac{\beta}{(N+1)\binom{N+1}{k}}\right) - (N - k + 1) \ln\left(\frac{\beta}{N\binom{N}{k}}\right)
\]

\[
= \ln\left(\frac{1}{\beta}\right) + \ln(N) + (N - k) \ln\left(\frac{N(N - k + 1)}{(N + 1)^2}\right) + \ln\left(\frac{N}{\binom{N}{k}}\right)
\]

We notice that $\ln(N)$, $\ln\left(\binom{N}{k}\right)$ and $\ln\left(\frac{N(N+1)}{(N+1)^2}\right) = \ln\left(1 - \frac{1}{N+1}\right)(1 - \frac{k}{N+1})$ are monotonically increasing with $N$, therefore $(N-k+1)(N-k) \ln\left(\frac{1 - \epsilon(N+1,k,\beta)}{1 - \epsilon(N,k,\beta)}\right) \geq \ln\left(\frac{k}{\beta}\right) > 0$, i.e. $\epsilon(N+1,k,\beta) \leq \epsilon(N,k,\beta)$.

Proof of Lemma 2 \[23]\. For the purpose of contradiction, we assume that there is a scenario $s \in S$ but $s \notin I$. According to the definition of support scenarios, $\text{opt}(N - s) < \text{opt}(N)$. However, Assumption \[4\] claims that removing scenarios will not increase the optimal objective value and $I \subseteq N - s$, we have $\text{opt}(N - s) = \text{opt}(I) = \text{opt}(N)$, which causes a contradiction.

Proof of Lemma 4. We first write out the Lagrange dual function $D(\mu, \lambda)$ of $\text{SP}(N)$:

\[
D(\mu, \lambda) = \inf_x \left( c^T x + \sum_{i=1}^N (\mu^*)^T f(x, \xi^i) + \lambda^T g(x) \right)
\]
The Lagrange dual problem is $\max_{\mu, \lambda} D(\mu, \lambda)$, s.t. $\mu \geq 0, \lambda \geq 0$. By assumption, we know that $\text{SP}(\mathcal{N})$ has a strictly feasible solution, thus Slater’s condition holds and $D(\mu^*_{\mathcal{N}}, \lambda^*_{\mathcal{N}}) = c^T x^*_{\mathcal{N}}$ by strong duality. We then consider the Lagrange dual problem of $\text{SP}(\mathcal{N} – i)$. The dual solution to $\text{SP}(\mathcal{N} – i)$ is denoted by $\lambda^*_{\mathcal{N} – i}$ and $\mu^*_{\mathcal{N} – i} = \{\mu^*_{\mathcal{N} – i}^1, \cdots, \mu^*_{\mathcal{N} – i}^{i-1}, \mu^*_{\mathcal{N} – i}^{i+1}, \cdots, \mu^*_{\mathcal{N} – i}^N\}$.

If $\xi^i$ is not a support scenario, then $c^T x^*_\mathcal{N} = c^T x^*_{\mathcal{N} – i}$, thus $D(\mu^*_{\mathcal{N}}, \lambda^*_{\mathcal{N}}) = c^T x^*_\mathcal{N} = c^T x^*_{\mathcal{N} – i} = D(\mu^*_{\mathcal{N} – i}, \lambda^*_{\mathcal{N} – i})$ by Slater’s condition and strong duality.

We could assign $\mu^*_{\mathcal{N} – i}^i = \mu^*_{\mathcal{N} – i}^N$ for $i \neq i$ and let $\mu^*_{\mathcal{N} – i}^i = 0$. Clearly this is one optimal solution to the dual problem of $\text{SP}(\mathcal{N})$. The uniqueness of this solution is due to the non-degeneracy of $\text{SP}(\mathcal{N})$ by assumption. Thus $\|\mu^*_{\mathcal{N} – i}\| = 0$.

**Proof of Lemma 5** We first prove (1). The case that $\hat{S} = \emptyset$ is obvious. For the case that $\hat{S}$ contains at least one scenario $s \in \hat{S}$. Solving the 2nd stage problem with $s$ removed gives a different optimal solution $\hat{x}$ with $c^T \hat{x} < c^T x^*$. Clearly $(\hat{x}, y^*)$ is a feasible solution to $\text{SP}(\mathcal{N} – s)$, with

$$c^T y^* + c^T \hat{x} < c^T y^* + c^T x^* \tag{18}$$

therefore $s$ is a support scenario for $\text{SP}(\mathcal{N})$ and $\hat{S} \subseteq S$.

We then prove (2). By Assumption 4 we know that $\text{opt}_A(\hat{S}) \leq \text{opt}_A(S) \leq \text{opt}_A(\mathcal{N})$ since $\hat{S} \subseteq S \subseteq \mathcal{N}$. If $\hat{S}$ is invariant, i.e. $\text{opt}_A(\hat{S}) = \text{opt}_A(\mathcal{N})$, then $\text{opt}_A(\mathcal{N}) \leq \text{opt}_A(S) \leq \text{opt}_A(\mathcal{N})$ gives $\text{opt}_A(S) = \text{opt}_A(\mathcal{N})$, therefore $\text{SP}(\mathcal{N})$ is non-degenerate.

**Proof of Corollary 2** We first prove (1), that is $\text{SP}(\mathcal{N})$ has a unique essential set if it is non-degenerate (similar with the proof of Lemma 2.11 in [23]). From Lemma 2, an essential set can be written as $\mathcal{E} = S \cup \mathcal{Y}$ where $\mathcal{Y} \subseteq (\mathcal{N} – S)$. The support set $S$ is invariant because of the non-degeneracy of $\text{SP}(\mathcal{N})$ by assumption. Since $\mathcal{E}$ is the invariant set of minimal cardinality, we can let $\mathcal{Y} = \emptyset$ and $S$ is the essential set. The support set $S$ is unique by definition, this implies the uniqueness of the essential set $\mathcal{E}$ for non-degenerate $\text{SP}(\mathcal{N})$.

We then prove (2). Lemma 2 shows that $S \subseteq \mathcal{R}$, we only need to show $\mathcal{R} \subseteq S$ when $\text{SP}_A(\mathcal{N})$ is non-degenerate. For the purpose of contradiction, we assume there exists $s \in \mathcal{R}$ but $s \notin S$. By hypothesis ($s \notin S$), we have $S \subseteq \mathcal{R} – s$ (Lemma 2). The monotonicity assumption 4 gives $\text{opt}_A(S) \leq \text{opt}_A(\mathcal{R} – s)$. Since $\mathcal{R}$ is irreducible, we have $\text{opt}_A(\mathcal{R} – s) < \text{opt}_A(\mathcal{R})$. $\text{SP}_A(\mathcal{N})$ is non-degenerate and $\mathcal{R}$ is invariant gives $\text{opt}_A(\mathcal{R}) = \text{opt}_A(\mathcal{N}) = \text{opt}_A(S)$. Combining the results
above, we have
\[
\text{opt}_A(S) \leq \text{opt}_A(R - s) < \text{opt}_A(R) = \text{opt}_A(N) = \text{opt}_A(S),
\]
which is clearly a contradiction. Therefore \( S = R \). \( \square \)

**Lemma 6.** Consider a (possibly non-convex) scenario problem \( SP_A(N) \) and an algorithm \( A \) satisfying Assumption 4. Suppose \( k \) is not a support scenario for \( SP_A(N) \), then
\[
S(N) \subseteq S(N - k)
\]

Note: \( SP(N - k) \) could have more support scenarios than \( SP(N) \).

*Proof of Lemma 6.* \( k \notin S \) and \( s \in S \) give \( \text{opt}(N - k) = \text{opt}(N) \) and \( \text{opt}(N - s) < \text{opt}(N) \), respectively. Assumption 4 shows \( \text{opt}(N - k - s) \leq \text{opt}(N - s) \). Hence, it holds that
\[
\text{opt}(N - k - s) \leq \text{opt}(N - s) < \text{opt}(N) = \text{opt}(N - k),
\]
\[
\forall s \in S(N), \tag{21}
\]
then \( s \) is a support scenario for \( SP(N - k) \). Therefore \( S(N) \subseteq S(N - k) \). \( \square \)

*Proof of Corollary 3.* \( (1) \Rightarrow (2) \) is proved in Corollary 2. And \( (2) \Rightarrow (3) \) is obvious, since the essential set \( E \) is irreducible. If there is only one irreducible set, then it is the essential set.

Lastly, we prove \( (3) \Rightarrow (1) \). We prove \( SP(N) \) being degenerate implies the essential set is not unique (equivalent with the statement that \( SP(N) \) is non-degenerate if it has a unique essential set). Suppose \( SP(N) \) is degenerate, i.e. \( \text{opt}(S) < \text{opt}(N) \). Consider an essential set \( E = S \cup T \) (Lemma 2), where \( T \) is non-empty and \( k \in T \). Consider the scenario problem \( SP(N - k) \), and \( \text{opt}(N - k) = \text{opt}(N) \) because \( k \notin S \). We also know that \( S \) is contained in any essential set of \( SP(N - k) \) by Lemma 6, i.e. \( E(N - k) = S \cup \hat{T} \). And \( \hat{T} \) has to be non-empty. Then \( \text{opt}(S \cup \hat{T}) = \text{opt}(N - k) = \text{opt}(N) \), therefore \( S \cup \hat{T} \) must contain at least one essential set that is different from \( S \cup T \) (because \( k \in T \) and \( k \notin \hat{T} \)). Therefore \( SP(N) \) has more than one essential set when it is degenerate. \( \square \)

7 Otherwise \( \text{opt}(S) = \text{opt}(E(N - k)) = \text{opt}(N - k) = \text{opt}(N) \), which contradicts with the hypothesis that \( SP(N) \) is degenerate.
### TABLE I: Settings of the 3-bus System

| Line No. | From Bus | To bus | Reactance (p.u.) | Capacity (MW) | Gen No. | Bus | Min | Max | Marginal Cost |
|----------|----------|--------|------------------|---------------|---------|-----|-----|-----|---------------|
| 1        | 1        | 2      | 1.0              | 20            | 1       | 1   | 20  | 100 | 1             |
| 2        | 1        | 3      | 1.0              | 100           | 2       | 2   | 20  | 90  | 100           |
| 3        | 2        | 3      | 1.0              | 100           |         |     |     |     |               |

| Bus | Forecast | Error 1 | Error 2 | Error 3 |
|-----|----------|---------|---------|---------|
| 3   | 110      | -30     | -35     |         |

| Bus | Forecast | Error 1 | Error 2 | Error 3 |
|-----|----------|---------|---------|---------|
| 2   | 30       | -15     | -25     |         |

### APPENDIX B

**SETTINGS OF THE 3-BUS SYSTEM**

All settings of the 3-bus system can found in Table I.
REFERENCES

[1] S. Takriti, J. R. Birge, and E. Long, “A stochastic model for the unit commitment problem,” IEEE Trans. Power Syst, 1996.
[2] L. Wu, M. Shahidehpour, and T. Li, “Stochastic Security-Constrained Unit Commitment,” IEEE Transactions on Power Systems, 2007.
[3] Q. P. Zheng, J. Wang, and A. L. Liu, “Stochastic optimization for unit commitment: A review,” IEEE Trans. Power Syst, 2015.
[4] D. Bertsimas, D. B. Brown, and C. Caramanis, “Theory and applications of robust optimization,” SIAM review, vol. 53, no. 3, pp. 464–501, 2011.
[5] D. Bertsimas, E. Litvinov, X. A. Sun, J. Zhao, and T. Zheng, “Adaptive robust optimization for the security constrained unit commitment problem,” IEEE Trans. Power Syst, 2013.
[6] X. Geng and L. Xie, “Data-driven decision making in power systems with probabilistic guarantees: Theory and applications of chance-constrained optimization,” Annual Reviews in Control, 2019.
[7] U. A. Ozturk, M. Mazumdar, and B. A. Norman, “A solution to the stochastic unit commitment problem using chance constrained programming,” IEEE Trans. Power Syst, 2004.
[8] D. Pozo and J. Contreras, “A chance-constrained unit commitment with an Sn-k$ security criterion and significant wind generation,” IEEE Trans. Power Syst, 2013.
[9] Q. Wang, Y. Guan, and J. Wang, “A chance-constrained two-stage stochastic program for unit commitment with uncertain wind power output,” IEEE Trans. Power Syst, 2012.
[10] C. Zhao, Q. Wang, J. Wang, and Y. Guan, “Expected value and chance constrained stochastic unit commitment ensuring wind power utilization,” IEEE Trans. Power Syst, 2014.
[11] W.-S. Tan and M. Shaaban, “A hybrid stochastic/deterministic unit commitment based on projected disjunctive milp reformulation,” IEEE Trans. Power Syst, 2016.
[12] A. Bagheri, C. Zhao, and Y. Guo, “Data-driven chance-constrained stochastic unit commitment under wind power uncertainty,” in Power & Energy Society General Meeting, 2017 IEE. IEEE, 2017.
[13] Y. Zhang, J. Wang, Y. Li, and X. Cao, “Chance-constrained Transmission Expansion Planning with Guaranteed Wind Power Utilization,” Chicago, IL, 2017.
[14] Q. Wang, J. Wang, and Y. Guan, “Price-based unit commitment with wind power utilization constraints,” IEEE Trans. Power Syst, 2013.
[15] H. Wu, M. Shahidehpour, Z. Li, and W. Tian, “Chance-constrained day-ahead scheduling in stochastic power system operation,” IEEE Trans. Power Syst, 2014.
[16] R. Jiang, J. Wang, and Y. Guan, “Robust unit commitment with wind power and pumped storage hydro,” IEEE Trans. Power Syst, 2012.
[17] K. Margellos, V. Rostampour, M. Vrakopoulou, M. Prandini, G. Andersson, and J. Lygeros, “Stochastic unit commitment and reserve scheduling: A tractable formulation with probabilistic certificates,” in Control Conference (ECC), 2013 European. IEEE, 2013.
[18] K. Hreinsson, M. Vrakopoulou, and G. Andersson, “Stochastic security constrained unit commitment and non-spinning reserve allocation with performance guarantees,” International Journal of Electrical Power & Energy Systems, 2015.
[19] K. Margellos, P. Goulart, and J. Lygeros, “On the road between robust optimization and the scenario approach for chance constrained optimization problems,” IEEE Transactions on Automatic Control, 2014.
[20] T. Alamo, R. Tempo, and A. Luque, “On the sample complexity of randomized approaches to the analysis and design under uncertainty,” in American Control Conference (ACC), 2010. IEEE, 2010.

[21] G. Calafiore and M. C. Campi, “Uncertain convex programs: randomized solutions and confidence levels,” Mathematical Programming, 2005.

[22] M. C. Campi and S. Garatti, “The exact feasibility of randomized solutions of uncertain convex programs,” SIAM Journal on Optimization, 2008.

[23] G. C. Calafiore, “Random convex programs,” SIAM Journal on Optimization, 2010.

[24] M. C. Campi, S. Garatti, and F. A. Ramponi, “A general scenario theory for non-convex optimization and decision making,” IEEE Transactions on Automatic Control, 2018.

[25] X. Geng and L. Xie, “Chance-constrained Unit Commitment via the Scenario Approach,” in Proceedings of the 51st North American Power Symposium, 2019.

[26] X. Geng, L. Xie, and M. S. Modarresi, “A General Scenario Theory for Security-Constrained Unit Commitment with Probabilistic Guarantees,” arXiv preprint arXiv:1910.07672, 2019.

[27] A. Charnes, W. W. Cooper, and G. H. Symonds, “Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil,” Management Science, 1958.

[28] M. C. Campi, S. Garatti, and M. Prandini, “The scenario approach for systems and control design,” Annual Reviews in Control, 2009.

[29] B. Pagnoncelli, S. Ahmed, and A. Shapiro, “Sample average approximation method for chance constrained programming: theory and applications,” Journal of optimization theory and applications, 2009.

[30] A. Nemirovski and A. Shapiro, “Convex approximations of chance constrained programs,” SIAM Journal on Optimization, 2006.

[31] M. S. Modarresi, L. Xie, M. Campi, S. Garatti, A. Car, A. Thatte, and P. Kumar, “Scenario-based Economic Dispatch with Tunable Risk Levels in High-renewable Power Systems,” IEEE Transactions on Power Systems, 2018.

[32] Illinois Institute of Technology, “IEEE 118-bus, 54-unit, 24-hour system,” 2004. [Online]. Available: http://motor.ece.iit.edu/data/

[33] I. Pea, C. B. Martinez-Anido, and B.-M. Hodge, “An extended IEEE 118-bus test system with high renewable penetration,” IEEE Trans. Power Syst, 2018.

[34] J. Lfberg, “YALMIP : A Toolbox for Modeling and Optimization in MATLAB,” in In Proceedings of the CACSD Conference, Taipei, Taiwan, 2004.

[35] I. Gurobi Optimization, Gurobi Optimizer Reference Manual, 2016. [Online]. Available: http://www.gurobi.com