Data-Driven Tuning for Chance-Constrained Optimization: Two Steps Towards Probabilistic Performance Guarantees

Ashley M. Hou and Line A. Roald, Member, IEEE

Abstract—Parameters involved in the formulation of optimization problems are often partially unknown or random. A popular way to mitigate the effect of uncertainty is using joint chance constraints, which guarantee constraint satisfaction with high probability but are challenging to solve. In this paper, we analyze an approach for joint chance constrained problems that involves iteratively tuning problem parameters. We first show that existing naive approaches to tuning can lead to solutions without feasibility guarantees. We then introduce a two-step approach, where a tuning-based solution generation step is followed by an a posteriori solution verification step. A main challenge of the two-step approach is to guarantee that the solution generated in the first step has a high probability of being verified as feasible in the second step. We therefore analyze how the relationship between the feasibility criteria used in each step impacts the probability of obtaining a feasible solution. We demonstrate our results in a numerical case study of the optimal power flow problem.

I. INTRODUCTION

Chance constraints, which require constraints hold probabilistically rather than deterministically, are an intuitive way to account for uncertainties present in optimization problems. We focus on joint chance constrained problems (JCCPs), where all constraints must be simultaneously satisfied with probability greater than $1 - \epsilon$ and $\epsilon$ is a pre-specified acceptable violation probability. Joint chance constraints provide a guarantee on the overall system risk and are, in general, challenging to solve. Approaches such as the scenario approach [1], [2], reformulations replacing the joint constraint using individual chance constraints [3]–[5], moment-based distributionally robust formulations [6], or conservative convex chance-constraint approximations [7] may lead to unnecessarily high cost or infeasibility for problems where there would exist a feasible solution. Achieving less conservative solutions might require strong assumptions such as Gaussian uncertainty [8], which may not be realistic, or the consideration of a large number of data samples within the optimization problem formulation [9], [10], which reduces computational tractability. Some methods utilize post-hoc evaluations to obtain a posteriori guarantees [11]–[14], which often provide tighter bounds on violation probability.

Another approach is to use tuning to adapt chance-constrained or stochastic formulations. Tuning-based approaches can broadly be defined as solution methods that combine approximate optimization models with a posteriori evaluations to iteratively refine the formulation [15]–[20]. Most methods identify a tuning parameter, also referred to as a safety parameter [21], which is modified to obtain solutions with higher violation probability but lower cost (or vice versa). Examples include adjusting the size of an uncertainty set [21], directly tightening constraints [16], [22], or tuning a smoothing parameter [17]. While such methods have computational and empirical performance advantages, most approaches utilize tuning in an ad hoc fashion, without considering how iterative tuning may impact the theoretical feasibility guarantees of the resulting solution. This paper aims at addressing this gap.

In this work, we propose a method to obtain feasibility guarantees for tuning-based algorithms by using a two-step approach, which includes a solution generation step, where we generate a solution by tuning the problem parameters, and a solution verification step, where we use a posteriori evaluations to obtain probabilistic guarantees on the resulting solution. Our main contribution is to analyze the relationships between the empirical feasibility criterion applied in the tuning step and the rigorous feasibility criterion applied in the verification step, and discuss how to design the tuning step such that the algorithm is likely to return a feasible solution. Our results are not tailored to a specific tuning method, but can be used with any tuning-based solution generation method. Furthermore, we make no distributional assumptions on the uncertainty, but assume that we have access to a limited set of representative scenarios.

The paper is organized as follows: Section II formulates the JCCP and reviews existing tuning-based solution methods. Section III discusses the challenges with obtaining feasibility guarantees. Section IV addresses these challenges by proposing our two-step tuning method and theoretical results. Section V presents numerical results via a case study applied to the optimal power flow (OPF) problem. Section VI concludes.

II. PRELIMINARIES

In this section, we present the problem formulation and review previous work on tuning-based solution algorithms.

A. Problem Formulation

The JCCP is defined as a minimization problem subject to a joint chance constraint,

$$\min_{x \in X} f(x) \quad \text{(1a)}$$

s.t. $\mathbb{P}_\xi(g_k(x, \xi) \leq 0, \forall k = 1, \ldots, K) \geq 1 - \epsilon$, \quad \text{(1b)}$

where $x \in \mathbb{R}^n$ represents the decision variables, $X \subseteq \mathbb{R}^n$ is a set defined by deterministic constraints, $\xi \in \mathbb{R}^n$ is a random...
vector, and \( \epsilon \in [0, 1] \) is a pre-specified level of acceptable constraint violation. The objective function is \( f : \mathbb{R}^n \to \mathbb{R} \) and the constraints are \( g_k : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R} \). The joint chance constraint \( \{g_k(x, \xi) \leq 0 \} \) requires that all individual constraints must be simultaneously satisfied with probability greater than \( 1 - \epsilon \). This is equivalent to requiring that

\[
G(x, \xi) := \max_{k=1, \ldots, K} g_k(x, \xi) \leq 0
\]

holds with probability at least \( 1 - \epsilon \). This problem formulation captures a very general class of problems, and does not make any assumptions about the distribution of \( \xi \), or regarding how \( x \) and \( \xi \) enter in \( g_k(x, \xi) \) and \( f(x) \). We assume access to data and ability to reformulate the problem such that it is numerically tractable and amenable to tuning (see Section II-B).

1) **Problem Feasibility**: We next characterize the violation probability of a solution \( \tilde{x} \). Following [9], we define the random variable \( Y_G(\tilde{x}, \xi) \) to be an indicator of whether the constraint \( G(\tilde{x}, \xi) \leq 0 \) is violated

\[
Y_G(\tilde{x}, \xi) := \begin{cases} 
0 & \text{if } G(\tilde{x}, \xi) \leq 0 \\
1 & \text{otherwise} 
\end{cases}
\]

Thus, we can write the violation probability of solution \( \tilde{x} \) in terms of the expected value of \( Y_G(\tilde{x}, \xi) \), i.e.,

\[
E_G(\tilde{x}) := \mathbb{E}_\xi[Y_G(\tilde{x}, \xi)] = \text{Pr}(G(\tilde{x}, \xi) > 0).
\]

A solution \( \tilde{x} \) is **feasible** for the chance-constrained problem \( \mathcal{P} \) if its violation probability is less than or equal to the acceptable violation probability \( \epsilon \), i.e., \( E_G(\tilde{x}) \leq \epsilon \).

2) **Empirical Feasibility**: The violation probability \( E_G(\tilde{x}) \) is typically difficult to evaluate, and is often approximated using available uncertainty data. We assume the availability of a set of \( N \) i.i.d. samples, \( \Xi = \{\xi^{(1)}, \ldots, \xi^{(N)}\} \) and evaluate \( Y_G(\tilde{x}, \xi^{(i)}) \) for each sample \( \xi^{(i)} \). For fixed \( \tilde{x} \) and i.i.d. samples \( \xi^{(i)} \), the realizations of \( Y_G(\tilde{x}, \xi^{(i)}) \) are i.i.d. as well. We define the solution \( \tilde{x} \) to be **empirically feasible** if the empirical violation probability \( \hat{E}_G(\tilde{x}, \Xi) \) satisfies

\[
\hat{E}_G(\tilde{x}, \Xi) := \frac{1}{N} \sum_{i=1}^{N} Y_G(\tilde{x}, \xi^{(i)}) \leq \epsilon.
\]

### B. Tuning-Based Solution Algorithms

We next review iterative tuning algorithms to obtain high quality solutions to the JCCP \( \mathcal{P} \). The key idea is to formulate a numerically tractable approximation to the JCCP and identify suitable **tuning parameters** which can be (i) tightened to obtain a more robust, more costly solution, or (ii) relaxed to obtain a less robust, less costly solution. The tuning method then **adapts the tuning parameters based on feedback from data**, with the goal of determining the tuning parameter values that provide the lowest cost, feasible solution to the JCCP. The algorithm proceeds as follows:

1) **Initialization** to define the approximate problem, tuning parameters, and appropriate initial values.
2) **Solve approximate problem** to find a candidate solution.
3) **Evaluate solution** to determine whether the solution has a too high or too low level of conservativeness.
4) **Update tuning parameters** based on the evaluation.

5) **Terminate** if the tuning parameters have converged.

There are many ways of defining the approximate problem and the tuning parameters, and many possible strategies to adapt the parameters in each iteration [15]–[21]. While our discussion and analysis in subsequent sections applies broadly to many tuning-based methods, we review the tuning algorithm proposed in [20] as a concrete example.

### C. Example: Tuning of Moment-Based Reformulation

The moment-based reformulation of \( \mathcal{P} \) replaces the JCCP with the following deterministic approximation

\[
\tilde{x} = \arg \min_{x \in \mathcal{X}} f(x)
\]

\[
s.t. \ g_k(x, \tilde{\xi}) + s \sigma_k(x, \xi) \leq 0 \quad \forall k = 1, \ldots, K.
\]

Here, \( \tilde{\xi} := \mathbb{E}_{\xi}[^{\ast}\xi] \) is the expected value of \( \xi \) and \( g_k(x, \tilde{\xi}) \) is the nominal constraint. The term \( s \sigma_k(x, \xi) \) is a constraint tightening, where \( \sigma_k(x, \xi) \) represents (an approximation of) the standard deviation of \( g_k(x, \xi) \) and \( s \in \mathbb{R}^+ \) is a tuning parameter that reflects the choice of violation probability \( \epsilon \). Choosing an appropriate parameter \( s \) is challenging in general. However, \( s \) can typically be increased to obtain solutions that are more robust to uncertainties, but also more costly (or decreased to reduce robustness and cost), and is thus a good candidate for tuning. We aim to find the \( s \) that gives the least conservative, feasible solution to the JCCP. We denote this value by \( s^* \) and the corresponding solution by \( x^* \). As an example tuning method, we review the bisection search algorithm from [20]:

1) **Initialization**: Initialize the tuning algorithm with iteration count \( j = 0 \), a set of samples \( \Xi = \{\xi^{(1)}, \ldots, \xi^{(N)}\} \) and upper and lower bounds for the tuning parameter \( s \), chosen such that \( s_{\text{min}} \leq s^* \leq s_{\text{max}} \).
2) **Solve approximate problem**: Set \( s^k = (s_{\text{max}} - s_{\text{min}})/2 \) and solve \( \mathcal{P} \) with \( s^k \) to obtain a candidate solution \( \tilde{x}^k \).
3) **Evaluate solution**: Evaluate \( \hat{E}_G(\tilde{x}^k, \Xi) \) and determine whether \( \tilde{x}^k \) satisfies the empirical feasibility criterion \( \hat{E}_G(\tilde{x}^k, \Xi) \leq \epsilon \).
4) **Update tuning parameters**: If \( \tilde{x}^k \) does not satisfy \( \hat{E}_G(\tilde{x}^k, \Xi) \), tighten the formulation by setting \( s_{\text{min}} = s^k \). If \( \tilde{x}^k \) is feasible, relax the formulation by setting \( s_{\text{max}} = s^k \).
5) **Termination**: If \( |s_{\text{max}} - s_{\text{min}}| \leq \gamma \), terminate and return the lowest cost solution \( x^* \) which satisfies \( \hat{E}_G(\tilde{x}^k, \Xi) \leq \epsilon \). Otherwise, update the iteration count \( j = j + 1 \) and go to Step 2.

### III. CHALLENGES OF EXISTING APPROACHES

While tuning-based approaches have been empirically observed to perform well [15]–[21], they often do not provide feasibility guarantees for the final solution. This is primarily due to two inaccuracies associated with empirically approximating the violation probability in an iterative manner.

### A. Challenge 1: Finite Number of Samples

The first challenge arises because we use a finite number of samples, and the empirical violation probability \( \hat{E}_G(\tilde{x}, \Xi) \) may differ from the true violation probability \( E_G(\tilde{x}) \). Therefore, even if a solution satisfies \( \hat{E}_G(\tilde{x}, \Xi) \leq \epsilon \), we cannot conclude \( E_G(\tilde{x}) \leq \epsilon \). To combat this challenge, we can use Hoeffding’s inequality to characterize the relationship
between the true and empirical violation probability of a solution [9].

Theorem III.1. (Hoeffding’s inequality [23]) For i.i.d. random variables $Y_1, \ldots, Y_N \in [0, 1]$ and $t \geq 0$,
\[ \Pr\left( |E[Y] - \frac{1}{N} \sum_{i=1}^{N} Y_i | > t \right) > 1 - \exp(-2Nt^2). \] (7)

Let $\delta$ denote the desired confidence level. By choosing
\[ t \geq \frac{\ln \frac{1}{\delta}}{\frac{2}{N}}, \] (8)
we guarantee with confidence $1 - \delta$ that the expectation is upper bounded by the sample mean plus margin $t$. We can apply this to candidate solutions to the JCCP. For a solution $\tilde{x}$ and set of i.i.d. samples $\Xi_1$, let $Y_i = Y_G(\tilde{x}, \xi^{(i)})$ and $E[Y] = E \xi[Y_G(\tilde{x}, \xi)]$. (Note that $Y_G(\tilde{x}, \xi^{(i)}) \in [0, 1]$ and $E[Y | \xi] = E_G(\tilde{x}, \xi)$.) Thus, given a desired confidence level $1 - \delta$ and margin $t$ chosen according to (8), we evaluate $\hat{E}_G(\tilde{x}, \Xi_1)$ and check if
\[ \hat{E}_G(\tilde{x}, \Xi_1) + t \leq \epsilon \] (9)
holds. If satisfied, by Hoeffding’s inequality, $\tilde{x}$ is feasible to the chance-constrained problem with confidence $1 - \delta$.

B. Challenge 2: Non-i.i.d. Realizations of $Y_G(\tilde{x}, \xi)$

The second less obvious, but more complex challenge is related to the fact that we evaluate $\hat{E}_G(\tilde{x}, \Xi)$ multiple times on the same set of samples. After obtaining candidate solution $\tilde{x}_1$ in the first iteration, the indicator random variables $Y_G(\tilde{x}_1, \xi)$ are i.i.d. due to the i.i.d. uncertainty samples $\xi$. However, in each subsequent iteration $j > 1$, because $s$ is tuned based on the empirical violation probability of the previous solution, $\tilde{x}_j$ is a function of the previous solution $\tilde{x}_{j-1}$ as well as the samples. Consequently, the random variables $Y_G(\tilde{x}_j, \xi^{(i)})$ are no longer i.i.d., so $\hat{E}_G(\tilde{x}_j, \Xi)$ is no longer an unbiased estimator of $E_G(\tilde{x})$ As a result, we can no longer apply Hoeffding’s inequality to assess feasibility.

IV. TUNING WITH PROBABILISTIC GUARANTEES: A TWO-STEP APPROACH

To address the issue of non-i.i.d. sample data in the tuning process, we propose a two-step method to recover feasibility guarantees for tuning-based methods. This two-step method consists of (1) a solution generation step that can utilize any existing (or new) tuning method to obtain a high quality solution, and (2) a solution verification step to guarantee solution feasibility.

A. Two-Step Approach

We split our sample set into $\Xi_1 = \{\xi^{(1)}, \ldots, \xi^{(N_1)}\}$ and $\Xi_2 = \{\xi^{(1)}, \ldots, \xi^{(N_2)}\}$. We use $\Xi_1$ to generate a solution $x^*$ with a tuning algorithm. We use $\Xi_2$ for an a posteriori solution verification to assess the solution feasibility.

\[ \text{Note that the i.i.d. issue could be resolved this by drawing a new sample set in every iteration. However, this would require much more data and make it harder for the algorithm to converge. Further, we would still be applying the Hoeffding inequality in each iteration, meaning that each iteration would have probability \( \delta \) of misclassifying an unsafe solution as safe.} \]

In the solution generation step, we run a tuning algorithm [16]–[20], but instead of using the empirical feasibility criterion [5], we define a new empirical feasibility criterion inspired by [9] that includes a tuning margin $t_{\text{tune}}$.
\[ \hat{E}_G(\tilde{x}, \Xi_1) + t_{\text{tune}} \leq \epsilon. \] (10)

We run the tuning algorithm to termination using this criterion and return the solution $x^*$ (note that this solution is not guaranteed to be feasible due to the issue of non-i.i.d. $Y_G(\tilde{x}_j, \xi^{(i)})$, even if we chose $t_{\text{tune}}$ according to (8)).

In the solution verification step, we use the separate sample set $\Xi_2$ to perform an a posteriori verification of the feasibility criterion for the obtained solution $x^*$. We define a verification margin $t$ according to (8) (using a desired confidence level $\delta$ and sample size $N_2$), and evaluate
\[ \hat{E}_G(x^*, \Xi_2) + t \leq \epsilon. \] (11)

If the solution $x^*$ satisfies this criterion, it is feasible to the original joint chance constraint with confidence $1 - \delta$.

Since any solution $x^*$ that satisfies (11) is chance-constraint feasible, the main remaining question is how to choose the tuning margin $t_{\text{tune}}$ such that the solution $x^*$ from the solution generation step will satisfy (11) with a high probability. We analyze this question below.

B. Choosing the Tuning Margin

To support our subsequent analysis, we first provide a few basic results and assumptions. We analyze the case where $N_1 = N_2 = N/2$, i.e., the samples are split equally between the solution generation and solution verification step.

Claim 1. Given a solution $x^*$ from the tuning process, which satisfies (10) by design, the following inequality holds
\[ \Pr(\hat{E}_G(x^*, \Xi_2) + t \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{\text{tune}} \leq \epsilon) \geq \Pr(\hat{E}_G(x^*, \Xi_2) + t \leq \hat{E}_G(x^*, \Xi_1) + t_{\text{tune}}). \] (12)

Proof:
\[ \Pr(\hat{E}_G(x^*, \Xi_2) + t \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{\text{tune}} \leq \epsilon) = \Pr(\hat{E}_G(x^*, \Xi_2) + t \leq \hat{E}_G(x^*, \Xi_1) + t_{\text{tune}}) \]
\[ + \Pr(\hat{E}_G(x^*, \Xi_1) + t_{\text{tune}} \leq \hat{E}_G(x^*, \Xi_2) + t \leq \epsilon) \]
\[ \geq \Pr(\hat{E}_G(x^*, \Xi_2) + t \leq \hat{E}_G(x^*, \Xi_1) + t_{\text{tune}}). \]
The equality results from equivalently partitioning the event and the inequality from non-negativity of probabilities.

Next, we make the following assumption for our analysis.

Assumption 1. Approximately unbiased estimator. For the remaining proofs in this section, we assume that it is a good approximation to treat $\hat{E}_G(x^*, \Xi_1)$ as an unbiased estimator of the violation probability $\epsilon$ of $x^*$.

Recall that $\hat{E}_G(x^*, \Xi_1)$ is not a truly unbiased estimator due to the dependence of $x^*$ on tuning samples $\Xi_1$. Thus, we cannot provide any feasibility guarantees based on $\hat{E}_G(x^*, \Xi_1)$. However, we conjecture that the bias of $\hat{E}_G(x^*, \Xi_1)$ is typically small, and use Assumption 1 in our derivations to determine an appropriate choice of $t_{\text{tune}}$. 
Note that if Assumption 1 is inaccurate, we may have a lower than desired probability that a solution $x^*$ from the tuning step will pass the solution verification (11). This prevents us from providing rigorous a priori feasibility guarantees for our algorithm. However, Assumption 1 does not affect the probabilistic guarantees provided in the solution verification step, as (11) only depends on the truly unbiased estimator $E_G(x^*, \Xi_2)$. Thus, even if Assumption 1 does not hold, any solution $x^*$ which passes the solution verification (11) will still have the desired probabilistic performance guarantees.

To assess the accuracy of Assumption 1 and the level of bias in $E_G(x^*, \Xi_1)$, we define the random variable $Z(x^*, \xi^{(i)}, \xi^{(j)}) := Y_G(x^*, \xi^{(i)}) - Y_G(x^*, \xi^{(j)})$ to represent whether two i.i.d. samples $\xi^{(i)}$ and $\xi^{(j)}$ both lead to violations or satisfaction of $G(x^*, \xi) \leq 0$. By Assumption 1

$$E[Z(x^*, \xi^{(i)}, \xi^{(j)})] = E[Y_G(x^*, \xi^{(i)})] - E[Y_G(x^*, \xi^{(j)})] = 0.$$ 

In actuality, $E[Z] \neq 0$ and can be interpreted as the expected bias of $E_G(x^*, \Xi_1)$ as an estimator of $E_G(x^*)$.

1) Two-Step Algorithm with Equal Margins $t_{tune} = t$: We next analyze the probability of passing the verification test (11) using equal margins $t_{tune} = t$.

Claim 2. Under Assumption 1 if we take $t_{tune} = t$ and tune the solution to have a violation probability of $\epsilon$, the probability that we satisfy the a posteriori feasibility criterion given that the tuning feasibility criterion holds is

$$P_{\xi}(\hat{E}_G(x^*, \Xi_2) + t > \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} > \epsilon) \geq 0.5.$$ (13)

Proof:

$$P_{\xi}(\hat{E}_G(x^*, \Xi_2) + t > \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} > \epsilon) \geq P_{\xi}(\hat{E}_G(x^*, \Xi_2) + t > \hat{E}_G(x^*, \Xi_1) + t_{tune}) \geq P_{\xi}(\hat{E}_G(x^*, \Xi_2) > \hat{E}_G(x^*, \Xi_1)) \geq 0.5,$$

where the first inequality results from Claim 1. The equalities hold due to $t = t_{tune}$ and Assumption 1.

Note that if the true probability of (13) differs from 0.5, Assumption 1 is inaccurate and $E_G(x^*, \Xi_1)$ is biased.

2) Two-Step Algorithm with Tuning Margin $t_{tune} = t + k$: In many cases, we want a high probability $> 0.5$ of passing the a posteriori check. We achieve this by introducing an additional margin $k$ in the tuning step.

Claim 3. Under Assumption 1 if we choose $t_{tune} = t + k$ with $k \geq \frac{\sqrt{2}}{N} \ln \frac{1}{\beta}$, where $\beta \in [0, 1]$, a solution $x^*$ from the tuning step will satisfy the a posteriori feasibility criterion with $1 - \beta$ confidence, i.e.,

$$P_{\xi}(\hat{E}_G(x^*, \Xi_2) + t > \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq 1 - \beta.$$ (14)

Proof: We can bound the probability as follows

$$P_{\xi}(\hat{E}_G(x^*, \Xi_2) + t > \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq P_{\xi}(\hat{E}_G(x^*, \Xi_1) - \hat{E}_G(x^*, \Xi_2) \leq k) = P_{\xi}\left(\frac{1}{N} \sum_{i=1,j=1}^{N} Z(x^*, \xi^{(i)}, \xi^{(j)}) \leq k\right) = P_{\xi}\left(\frac{1}{N} \sum_{i=1,j=1}^{N} Z(x^*, \xi^{(i)}, \xi^{(j)}) \leq k\right) = P_{\xi}\left(\frac{1}{N} \sum_{i=1,j=1}^{N} Z(x^*, \xi^{(i)}, \xi^{(j)}) \leq k\right) \geq 1 - \exp(Nk^2).$$

The first inequality follows from Claim 1 and the definition of $t_{tune} = t + k$. The last inequality results from Hoeffding’s inequality [23], since $k > 0$ and random variables $Z(x^*, \xi^{(i)}, \xi^{(j)})$ are i.i.d. and bounded on $[-1, 1]$. By choosing $k = \sqrt{(1/N) \ln(1/\beta)}$, we obtain $1 - \beta$ confidence.

Remark. With $k = 0$, we obtain $\beta = 1$, which indicates that we have no confidence that our solution $x^*$ will satisfy the a posteriori feasibility check. However, Claim 2 shows that $k = 0$ actually provides a feasible solution with probability 0.5. This shows that the bound in Claim 3 is very conservative.

3) A Priori Guarantees on Solution Feasibility: Under Assumption 1 we provide a heuristic guarantee that the solution $x^*$ obtained with our two-step method will be feasible both in the verification step and to the original JCCP.

Claim 4. Under Assumption 1 and for $t_{tune} = t + k$, $t \geq \frac{\sqrt{2}}{N\epsilon} \ln \frac{1}{\beta}$, and $k \geq \frac{\sqrt{2}}{N\epsilon} \ln \frac{1}{\beta}$, the probability that a solution $x^*$ obtained from our two-step process is feasible to the JCCP is

$$P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq (1 - \delta_k)(1 - \beta).$$

Proof:

$$P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) = P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \cdot P_{\xi}(\hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \cdot P_{\xi}(\hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon).$$

Here, we first apply Bayes’ rule to obtain the first equality and then lower bound the expression by dropping the second term. The second term in the last equation can be bounded by Claim 3. The first term can be bounded as follows

$$P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) = P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq 1 - \delta_k.$$}

Here, the equality holds by design of the tuning algorithm, where $x^*$ must satisfy (10). The second inequality is obtained from application of Hoeffding’s inequality.

Remark. The above proof relies on Claim 3 which uses Assumption 1. However, making this assumption allows us to apply Hoeffding’s inequality to $E_G(x^*, \Xi_1)$ and obtain a bound on the probability that $x^*$ is feasible to the JCCP: $P_{\xi}(E_G(x^*) \leq \epsilon) = P_{\xi}(E_G(x^*) \leq \epsilon | \hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \cdot P_{\xi}(\hat{E}_G(x^*, \Xi_1) + t_{tune} \leq \epsilon) \geq (1 - \delta_k)(1 - \beta)(1 - \delta_{tune}).$
V. CASE STUDY: OPTIMAL POWER FLOW

We now apply our two-step algorithm to the joint chance-constrained optimal power flow problem (CC-OPF) [20]. Here, $\xi$ represents uncertainty from renewable generation and load variability, while the decision variable $x$ represents the generation dispatch. We formulate the power flow using the linearized DC approximation with power transfer distribution factors. Our objective is to minimize the total cost of conventional generation subject to physical system constraints. The resulting formulation is a convex quadratic program with linear constraints, where uncertainty only occurs in the right-hand-side (refer to [20] for details). We generate solutions by tuning the moment-based reformulation using the bisection algorithm in [11] with tuning feasibility criterion (10), and then test solution feasibility using the test criterion (11) in the verification step.

We perform our experiments on a modified version of the IEEE RTS96 24-bus system [24]. To represent variations in load and renewable energy, we add uncertainty sources derived from a mixture of uniform and Gaussian distributions on buses 8 and 15. More details on the system modifications can be found in [20]. For the bisection search, we use $s^0_{\text{min}} = 0$, $s^0_{\text{max}} = \sqrt{(1-\epsilon)/\epsilon}$ and $\gamma = 0.005$.

A. Choosing the Tuning Margin

Here, we explore how our algorithm performs under various choices of $t_{\text{tune}}$. We use $\epsilon = 0.05$ and $\delta = 0.01$, and assume availability of $N_1 = N_2 = 0$, 000 samples.

1) Equal Tuning Margins $t_{\text{tune}} = t$: We first examine the case with equal tuning and verification margins $t_{\text{tune}} = t = 0.0215$. This case allows us to assess the bias of $E_G(x^*, \Xi_1)$, and if this bias is small, Assumption [1] is valid. We run 1,000 replications of the algorithm. Our algorithm results in solutions that satisfy the a posteriori verification criterion for 497/1000 replications. This is consistent with our theoretical result [13], indicating that the bias of $E_G(x^*, \Xi_1)$ is small.

To further assess the bias of $E_G(x^*, \Xi_1)$, Fig. 1 shows the histogram of $E[Z] = E_G(x^*, \Xi_1) - E_G(x^*, \Xi_2)$ based on the data obtained from the 1,000 replications of the algorithm (with a new set of 10,000 samples used for each replication). Given that $E_G(x^*, \Xi_2)$ is an unbiased, we would expect this solution to be centered around 0 if $E_G(x^*, \Xi_1)$ is also unbiased. However, we can notice that the median (plotted in red) is slightly lower than the mean (plotted in black). We see that the 10% quantile (dashed line) is further from the median than the 90% quantile (dotted line), further demonstrating evidence of the negative bias of $E_G(x^*, \Xi_1)$.

2) Tuning Margin $t_{\text{tune}} = t + k$: We next investigate how using an extra tuning margin increases the probability of obtaining solutions that satisfy the test criterion (11) in the verification step. Here, we again define the verification margin $t = 0.0215$ according to [8] but choose the tuning margin to be $t_{\text{tune}} = t + k$ with $k = \sqrt{(1/N_2)} \ln 1/\beta$, where $1 - \beta$ is the confidence level that a solution generated from tuning satisfies the a posteriori check. We use confidence values $1 - \beta = \{0.05, 0.25, 0.5, 0.9, 0.95\}$ and perform 400 replications of the algorithm (assuming a new set of 10,000 samples in each replication). The results are reported in Table 1. For each replication, we also perform an out-of-sample evaluation based on 100,000 samples to more accurately assess whether $x^*$ is feasible to original CC-OPF problem.

In Table 1 for $1 - \beta > 0.5$, all replications result in solutions that satisfy the a posteriori check. For confidence values $1 - \beta = \{0.05, 0.25\}$, the actual proportion of solutions passing the a posteriori check is 0.85 and 0.995, respectively.

The portion of replications satisfying the a posteriori criterion (11) is thus significantly higher than $1 - \beta$, indicating that our bound is conservative. The cost does varies only slightly with different levels of confidence, with a maximum difference of less than 1%. This indicates that we can obtain solutions with higher confidence guarantees but little increase in total cost. Finally, in the out-of-sample evaluation, for all confidence values and all runs, the algorithm results in solutions satisfying $E_G(x^*, \Xi_{\text{oos}}) \leq \epsilon$, indicating that our method does result in conservative solutions.

B. Comparison to Scenario Approach

We compare the performance of our two-step algorithm with the scenario approach, a widely used method for solving joint chance constrained problems [1]. The scenario approach replaces the original problem with a reformulation that enforces feasibility of all constraints for $N_s$ i.i.d. samples of $\xi$. Given $n = |x|$ decision variables, we choose

$$N_s \geq \frac{1}{\epsilon} \left( n + \ln(\frac{1}{\beta}) + \sqrt{2n \ln \frac{1}{\beta}} \right),$$

(15)

![Fig. 1: The distribution of $\hat{E}_G(x^*, \Xi_1) - \hat{E}_G(x^*, \Xi_2)$. The distribution is skewed towards negative values, demonstrating the bias in $E[Z] = E_G(x^*, \Xi_2) - E_G(x^*, \Xi_1)$.](image)

TABLE 1: Solutions obtained with $t_{\text{tune}} = t + k$ and different confidence levels $1 - \beta$.

<table>
<thead>
<tr>
<th>Confidence</th>
<th>1 - \beta</th>
<th>0.05</th>
<th>0.25</th>
<th>0.5</th>
<th>0.9</th>
<th>0.95</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t$</td>
<td>0.0215</td>
<td>0.0215</td>
<td>0.0215</td>
<td>0.0215</td>
<td>0.0215</td>
<td></td>
</tr>
<tr>
<td>$k$</td>
<td>0.0032</td>
<td>0.0076</td>
<td>0.0118</td>
<td>0.0213</td>
<td>0.0245</td>
<td></td>
</tr>
<tr>
<td>$t_{\text{tune}}$</td>
<td>0.0247</td>
<td>0.0291</td>
<td>0.0332</td>
<td>0.0429</td>
<td>0.0459</td>
<td></td>
</tr>
<tr>
<td>Feasible solutions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(algorithm)</td>
<td>340/400</td>
<td>398/400</td>
<td>400/400</td>
<td>400/400</td>
<td>400/400</td>
<td></td>
</tr>
<tr>
<td>(out-sample)</td>
<td>400/400</td>
<td>400/400</td>
<td>400/400</td>
<td>400/400</td>
<td>400/400</td>
<td></td>
</tr>
<tr>
<td>Parameters</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(avg)</td>
<td>42,763</td>
<td>42,798</td>
<td>42,835</td>
<td>42,972</td>
<td>43,057</td>
<td></td>
</tr>
<tr>
<td>(avg, feas.)</td>
<td>42,766</td>
<td>42,796</td>
<td>42,835</td>
<td>42,973</td>
<td>43,056</td>
<td></td>
</tr>
<tr>
<td>(min)</td>
<td>42,726</td>
<td>42,737</td>
<td>42,775</td>
<td>42,890</td>
<td>42,955</td>
<td></td>
</tr>
<tr>
<td>(max)</td>
<td>42,810</td>
<td>42,849</td>
<td>42,894</td>
<td>43,053</td>
<td>43,184</td>
<td></td>
</tr>
</tbody>
</table>
TABLE II: Results of comparison between two-step tuning algorithm and the scenario approach.

<table>
<thead>
<tr>
<th>Cost Violation (Out-sample)</th>
<th>Tuning</th>
<th>Scenario approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>(avg)</td>
<td>42,731</td>
<td>43,144</td>
</tr>
<tr>
<td>(std dev)</td>
<td>5.2</td>
<td>103.5</td>
</tr>
<tr>
<td>Cost Violation</td>
<td>0.0297</td>
<td>0.0039</td>
</tr>
<tr>
<td>(std dev)</td>
<td>0.0012</td>
<td>0.0020</td>
</tr>
</tbody>
</table>

which provides an a priori guarantee that the solution $x^*_s$ satisfies the joint chance constraint (10) with confidence $1 - \delta_s$.

In our tuning algorithm, we use $\epsilon = 0.05$, $\delta_t = 0.001$, $\beta = 0.001$, and $\delta_{\text{tune}} = 0.001$. To create a comparison that is as close as possible, we use a confidence level of $1 - \delta_s = (1 - \delta_t)(1 - \beta)(1 - \delta_{\text{tune}}) = 0.002$ for the scenario approach. With $n = 33$, we require $N_s = 1,168$ samples to achieve confidence $1 - \delta_s$. For our two-step tuning approach, we use $N_1 = N_2 = 25,000$, resulting in $t_{\text{tune}} = 0.0201$, $t = 0.0083$, and $k = 0.0118$. In Table II, we compare the average and standard deviation of the solution costs across 100 replications, as well as the empirical violation evaluated on an out-of-sample set with 100,000 samples for each replication. With these parameters, all solutions generated by the two-step tuning method satisfy the a posteriori feasibility test (11), and are thus feasible with probability $1 - \delta = 0.999$.

We observe that the two-step tuning algorithm achieves a 1% lower average cost than the scenario approach, and a much smaller standard deviation in solution costs across replications. The average out-of-sample violation probabilities are 0.0297 and 0.0039 for solutions obtained from the two-step tuning method and scenario approach, respectively, and again our tuning method results in lower variance than the scenario approach. From this, we see that solutions obtained from tuning are significantly less conservative, with lower cost and higher out-of-sample empirical violation. The two-step tuning approach also provides more consistent solutions by leveraging the information available in the larger set of samples.

VI. CONCLUSION AND FUTURE WORK

This paper proposes a two-step method for chance constraint tuning, generating solutions via a tuning step and providing probabilistic feasibility guarantees via an a posteriori verification step. We provide theoretical results that give guidance on how to adapt the feasibility criteria used in the solution generation and verification steps to increase the probability that the resulting solutions be feasible in the verification step. While these results rely on a limiting assumption which, the analysis takes us a step towards providing a priori probabilistic feasibility guarantees for tuning based methods. We apply our two-step tuning method to the optimal power flow problem, obtaining numerical results supporting our theoretical conclusions. We further demonstrate that tuning is able to obtain less conservative solutions compared to the scenario approach. Future work includes obtaining tighter probabilistic bounds, stronger feasibility guarantees, considering more sophisticated tuning approaches (e.g., multi-dimensional tuning) and application to other problem formulations.

REFERENCES