Sequential Learning under Probabilistic Constraints

Amirhossein Meisami
Adobe, Inc.
San Jose, CA 95110

Henry Lam
Columbia University
New York, NY 10027

Chen Dong
Adobe, Inc.
San Jose, CA 95110

Abhishek Pani
Adobe, Inc.
San Jose, CA 95110

Abstract

We provide the first study on online learning problems under stochastic constraints that are "soft", i.e., need to be satisfied with high probability. These constraints are imposed on all or some stages of the time horizon so that the stage decisions probabilistically satisfy some given safety conditions. The distributions that govern these conditions are learned through the collected observations. Under a Bayesian framework, we introduce a scheme that provides statistical feasibility guarantees through the time horizon, by using posterior Monte Carlo samples to form sampled constraints which leverage the scenario generation approach in chance-constrained programming. We demonstrate how our scheme can be integrated into Thompson sampling and illustrate it with an application in online advertisement.

1 INTRODUCTION

Most of the literature in stochastic online learning focuses on performances measured by optimality achievement. Common examples include the minimization of cumulative regret in the multi-arm bandit setting (e.g., Auer et al. (2002); Lai and Robbins (1985)), best arm selection (e.g., Audibert and Bubeck (2010)) and the closely related ranking and selection (e.g., Boesel et al. (2003)) in the simulation literature. In many situations, however, the uncertainty or the stochasticity appears not only in the objective function, but also in the constraints of the problem whose feasibility can be of utmost importance. The focus of this paper is to design sequential methodologies that maintain probabilistic feasibility requirements with rigorous statistical guarantees.

Our study is motivated from a rich set of problems where "budgets" or "resources" are limited for various operational or commercial reasons, and these constraints are in a sense "soft", i.e., the capacities placed on these constraints, while preferred to be satisfied, are allowed to be violated with a small probability. Such consideration is common among applications. For example, in online advertisement problems encountered by our coauthors when optimizing spending for large advertisers, the task involves sequentially picking items (e.g., keywords, targets) to maximize revenues, while adhering to a specified marketing budget for a duration. The marketer in general expects to meet the budget goals. However, if occasionally the budget is exceeded the campaign is still acceptable as long as the revenue performance is sustained. Other similar settings include clinical trials, where the costs of competing treatments are substantial and noisy, and over-budget is undesired but sometimes allowable. Whereas past work in stochastic sequential learning has focused on rewards (with hard constraints if needed), this paper provides the first study on a class of problems that not just include the rewards but also stochastic constraints that need to be satisfied with high probability.

Our framework can be viewed as a sequential problem under so-called *probabilistic* or *chance constraints* (Prékopa, 2003), which has been widely used in stochastic programming under limited and uncertain resources (e.g., Shi et al. (2015); Lejeune and Ruszczynski (2007)). A generic representation of a chance constraint is

$$P((x,\xi) \text{ satisfies a given safety condition}) \geq 1-\alpha \quad (1)$$

where x is a decision variable and ξ denotes some randomness distributed under P. Satisfying the safety condition means that (x,ξ) lies in a desirable deterministic region, which can be represented by, e.g., a set of inequalities. The given parameter α is the tolerance level that represents the allowable probabilistic violation.

In the sequential setting, x would denote a sequence of decisions. The safety condition could include individual

requirements on all or some stages. In many applications of interest, P needs to be learned as complete distributional knowledge on ξ is not available. Along the vein of conventional online problems that focus on optimality, at each stage we may observe some components of ξ so that we can update our belief on P.

As our main methodological contribution, we analyze an online strategy that provides guarantees on (1) with a high confidence, under a statistical framework that we shall describe. On a high level, it means we can guarantee, with our proposed policy, that

$$P(P((x,\xi) \text{ satisfies a given safety condition}) \geq 1-\alpha) \geq 1-\beta \quad (2)$$

where the outer probability now refers to the randomness of x induced by the sequential observations, and $1-\beta$ is a confidence level (90% for instance). Our methodology is based on a combination of two ideas. First is a Bayesian extension of the so-called scenario generation or constraint sampling (Calafiore and Campi, 2005; De Farias and Van Roy, 2004) approach in approximating chance-constrained optimization problems. This approach replaces the unknown or difficult chance constraint with a collection of sampled constraints that come from data or from numerical simulation. Viewing such an approach in a Bayesian manner allows it to be blended naturally into popularly used online learning algorithms such as Thompson sampling (Agrawal and Goyal, 2012; Russo and Van Roy, 2016) that also operates via Bayesian updating. Second, by capitalizing results on scenario generation in the static setting, we can derive the precise number of samples required at each stage of the sequential process such that (2) holds throughout the horizon. As far as we know, our formulation and analysis of chance-constraint guarantees in an online setting is new to the literature.

After presenting our theoretical investigation on feasibility guarantees, we illustrate the integration of our scheme into a variant of Thompson sampling in an online advertisement setting. We then numerically demonstrate how this chance-constrained Thompson sampling performs competitively, in achieving feasibility but also maintaining good objective values.

2 RELATED WORK

The earliest work in chance constraints dated back to Charnes et al. (1958) and Miller and Wagner (1965). Exact solution techniques for such problems are notoriously difficult due to non-convexity, and are only available in few instances even when P is known; e.g., Lagoa et al. (2005). Several lines of approximation methodologies have been proposed. A conventional method is to use so-called safe convex approximation that replaces the chance constraint with more conservative convex constraints (Ben-Tal and Nemirovski, 2000). Rossi et al. (2011, 2015) used policy trees and confidence interval construction to obtain the so-called (α, ϑ)-solution. Scenario generation (Calafiore and Campi, 2006; Campi and Garatti, 2008), which we leverage on in this work, uses sampled constraints to populate the feasible region. This approach has several extensions, such as sampling-and-discarding (Campi and Garatti, 2011) and multi-phase schemes (Carè et al., 2014; Calafiore, 2017; Chamanbaz et al., 2016), and relates to sample average approximation (Luedtke et al., 2010). Other data-driven methods include distributionally robust optimization (Calafiore and El Ghaoui, 2006; Zymler et al., 2013) and data-driven robust optimization (Bertsimas et al., 2013).

Our work focuses on chance-constrained problem in an online fashion, under the broad umbrella of sequential decision-making. In the later part of this paper, we demonstrate our proposed strategy in a variant of the stochastic multi-arm bandit problem (Auer et al., 2002) used to address the well-known exploration-exploitation tradeoff. In budgeted bandits, Ding et al. (2013) consider the presence of random costs and an overall budget, where learning and revenue accumulation stops when the budget runs out. Xia et al. (2015) study Thompson sampling for a similar setting; in this work we integrate our strategy into Thompson sampling, especially the one considered in Ferreira et al. (2016) motivated from network revenue management. Other related work include those in the framework of "bandits with knapsacks" (Badanidiyuru et al., 2013; Tran-Thanh et al., 2012; Besbes and Zeevi, 2012) that have been applied in pricing and supply chain management (Wang et al. (2014)) and healthcare (Villar et al. (2015)). The works closest to our online advertisement example are Tran-Thanh et al. (2014) and Amin et al. (2012) that study the problem of item bidding under a budget, but they do not consider probabilistic violation of the constraints that we focus on.

3 CHANCE-CONSTRAINED ONLINE LEARNING

Consider a sequence of decision variables $x_t \in \mathbb{R}^d, t = 1, \ldots, T$, and a sequence of random variables $\xi_t \in \Xi_t, t = 1, \ldots, T$ assumed independent among the steps t in a given horizon T. For convenience, denote $\xi_{1:t} = (\xi_1, \ldots, \xi_t)$ and $x_{1:t} = (x_1, \ldots, x_t)$ as the cumulative randomness and decisions up to t. Consider a sequence of safety conditions that we write as $f_t(x_{1:t}, \xi_{1:t}) \in \mathcal{A}_t$, where each function f_t maps to some space $\mathcal{Y}_t$ such that

$\mathcal{A}_t \subset \mathcal{Y}_t$ (for example, $f_t(x_{1:t}, \xi_{1:t}) \in \mathcal{A}_t$ can be a set of inequalities so that $\mathcal{Y}_t = \mathbb{R}^m$ for some m and $\mathcal{A}_t = \{y \in \mathbb{R}^m : y \leq 0\}$.

We are interested in a sequential problem with horizon T:

$$\begin{aligned} \max_{x_1,\ldots,x_T} \quad & h(x_1, \ldots, x_T) \\ \text{subject to} \quad & P(f_t(x_{1:t}, \xi_{1:t}) \in \mathcal{A}_t | \mathcal{F}_{t-1}) \geq 1 - \alpha \; \forall t \in \mathcal{S}, \end{aligned} \tag{3}$$

where the decisions $x_1, \ldots, x_T$ are sequential, i.e., x_{t+1} depends on the past observations of $\xi_{1:t}$ and past decisions $x_{1:t}$, $\mathcal{S} \subset \{1, \ldots, T\}$ is a given set, and $h(\cdot)$ is the objective function. Note that the function f_t and the set $\mathcal{A}_t$ can depend on the time step t. For convenience, let $\mathcal{F}_t = \{\xi_{1:t}, x_{1:t}\}$ be the information up to time t. In each probability P in (3), the function $f_t(x_{1:t}, \xi_{1:t})$ can be expressed as $f_t(x_{1:(t-1)}, x_t, \xi_{1:(t-1)}, \xi_t)$ where $x_{1:(t-1)}$ and $\xi_{1:(t-1)}$ belong to the past information $\mathcal{F}_{t-1}$.

Consistent with the introduction, α is a tolerance parameter on the violation of the safety condition. This parameter is assumed constant across t for convenience, but our analysis can be easily adapted to the case where it varies. Note that $\mathcal{S}$ determines how many chance constraints need to be maintained throughout the horizon. For example, $\mathcal{S} = \{1, \ldots, T\}$ means there is a budget requirement for each step, and $\mathcal{S} = \{T\}$ means there is only one overall budget requirement across the whole horizon.

3.1 SCENARIO GENERATION FOR STATIC PROBLEMS

We first discuss a well-studied approach to approximate a static version of (3). Suppose $T = 1$. In this setting we can simplify notation and write the formulation as

$$\begin{aligned} \max_{x} \quad & h(x) \\ \text{subject to} \quad & P(f(x, \xi) \in \mathcal{A}) \geq 1 - \alpha \end{aligned} \tag{4}$$

Suppose we can simulate or collect data for ξ to obtain, say, i.i.d. $\xi^1, \ldots, \xi^N$. We consider replacing the constraint in (4) by sampled constraints, so that the optimization program becomes

$$\begin{aligned} \max_{x} \quad & h(x) \\ \text{subject to} \quad & f(x, \xi^n) \in \mathcal{A}, \; \forall \, n = 1, \ldots, N \end{aligned} \tag{5}$$

We call (5) a sampled program, which serves as a reasonable approximation to (4) when N is large. However, since ξ^n's are randomly generated, the solution obtained from (5) is subject to statistical noise and cannot be guaranteed feasible for (4). The following celebrated result from (Calafiore and Campi, 2006; Campi and Garatti, 2008) gives the sample size needed to guarantee feasibility for (4) with a certain confidence by solving (5).

Condition 1. *An optimization program is said to be in class $\mathcal{R}$ if: 1) It is feasible and the feasible region has a non-empty interior; 2) Its optimal solution exists and is unique.*

Theorem 1. *(Adopted from Theorem 2.4, Campi and Garatti (2008)) Suppose for each ξ, $f(x, \xi) \in \mathcal{A}$ is a convex set in $x \in \mathbb{R}^d$, and h is concave. Suppose also that any instance of* (5) *belongs to $\mathcal{R}$. Fix real numbers $\alpha, \beta \in [0, 1]$. Then for N chosen such that*

$$\sum_{i=0}^{d-1} \binom{N}{i} \alpha^i (1 - \alpha)^{N-i} \leq \beta$$

the optimal solution of the sampled stochastic program (5) *is feasible for* (4) *with probability no smaller than $1 - \beta$.*

It is known that this result can be improved, e.g., by using sampling-and-discarding (Campi and Garatti, 2011) and multi-stage or iterative schemes (Carè et al., 2014; Calafiore, 2017; Chamanbaz et al., 2016). In this paper we stick with the requirement in Theorem 1 to illustrate our proposed strategies; improvements can be made accordingly by modifying the use of Theorem 1 to better results available in the literature.

3.2 SCENARIO GENERATION UNDER UNKNOWN DISTRIBUTION: A BAYESIAN PERSPECTIVE

The scenario generation approach depicted in Theorem 1 requires direct observations on ξ or the capacity to obtain Monte Carlo samples for ξ. In problems with learning, the distribution of ξ is not fully known, and Theorem 1 does not apply directly. We shall adopt a Bayesian perspective that naturally integrates to many online algorithms (e.g., Thompson sampling). Suppose ξ follows a parametric distribution $G|\mu$ with unknown parameter μ. After specifying a prior distribution for μ and collecting some data historically, we have a posterior distribution for μ denoted by F. We seek to use Monte Carlo sampling to conduct an analog of scenario generation so that a posterior credibility guarantee

$$P_\mu(P_{\xi|\mu}(f(x, \xi) \in \mathcal{A}) \geq 1 - \alpha) \geq 1 - \beta \tag{6}$$

is achieved, where P_μ denotes the posterior probability distribution on μ, and $P_{\xi|\mu}$ denotes the distribution of ξ given a parameter value of μ. In other words, we want the chance constraint to hold with a posterior credibility level $1 - \beta$. Note that this is a natural Bayesian analog of the frequentist result in Theorem 1. In the setting of

Theorem 1, P is not known but data are available, so that a $1-\beta$ confidence is attained. In our current Bayesian investigation, P is not known but subject to a posterior belief summarized by the distribution of μ, and we want this posterior credibility to be $1-\beta$.

Directly sampling ξ using any particular value of μ does not sufficiently capture the posterior uncertainty. To blend the latter into a scenario generation, we can use a two-level sampling, where in the first level we generate a posterior sample for μ, and in the second level we generate ξ conditional on μ. This sampling procedure is described in Algorithm 1.

Algorithm 1 Posterior Constraint Sample Generator (PCSG)

1. Repeat N times:
   (a) Generate $\mu^n \sim F$.
   (b) Generate $\xi^n \sim G|\mu^n$.
2. Impose the constraints $f(x,\xi^n) \in \mathcal{A}$, $n = 1,\dots,N$ and solve the sampled program

$$\begin{aligned} \max_x \quad & h(x) \\ \text{subject to} \quad & f(x,\xi^n) \in \mathcal{A},\ \forall\, n = 1,\dots,N \end{aligned} \tag{7}$$

In PCSG we encounter two different sources of randomness. First is the statistical noise from the uncertainty of μ, captured by the posterior credibility level $1-\beta$. The second source is the Monte Carlo error, and we denote by $1-\delta$ the confidence level induced from this error. By choosing a suitable sample size N in terms of α,β,δ, PCSG turns out to achieve a guarantee below.

Theorem 2. *Suppose $f(x,\xi) \in \mathcal{A}$ is a convex set in $x \in \mathbb{R}^d$, and $h(x)$ is concave. Suppose also that any instance of (7) belongs to $\mathcal{R}$. Fix real numbers $\delta,\alpha,\beta \in [0,1]$ and choose*

$$\sum_{i=0}^{d-1} \binom{N}{i} (\alpha\beta)^i (1-\alpha\beta)^{N-i} \le \delta \tag{8}$$

Consider a solution x obtained from the sampled program in PCSG. Then,

1. *x satisfies (6) with a Monte Carlo confidence $1-\delta$, i.e.,*

$$P_{MC}(P_\mu(P_{\xi|\mu}(f(x,\xi)\in\mathcal{A}) \ge 1-\alpha) \ge 1-\beta) \ge 1-\delta \tag{9}$$

   where the outermost P_{MC} denotes the probability with respect to the N Monte Carlo samples.

2. *x satisfies*

$$E_{MC}[P_\mu(P_{\xi|\mu}(f(x,\xi)\in\mathcal{A}) \ge 1-\alpha)] \ge (1-\beta)(1-\delta) \tag{10}$$

   where $E_{MC}[\cdot]$ denotes the expectation with respect to the N Monte Carlo samples.

Theorem 2 Part 1 stipulates that choosing N in (8) achieves chance-constraint feasibility with a Bayesian credibility $1-\beta$, under a Monte Carlo confidence $1-\delta$. Part 2 will be useful in generalizing to the multi-stage setting presented next.

3.3 SEQUENTIAL POLICIES

We now move our analysis to the sequential problem depicted in (3). We generalize PCSG, with the posterior update occurring at every step of the horizon and the sample size required at each step modified in order to achieve a chance constraint guarantee over the whole horizon. Let N_t be the sample size used in step t, which depends on α and also the confidence-level parameters β_t and δ_t. We denote F_0 as the prior distribution of μ and F_t as the posterior distribution of μ at step t. We denote $G_t|\mu$ as the distribution of ξ_t given μ. Note that μ is a parameter shared among the ξ_t at different steps so that information can be learned over time. To distinguish the real data from the Monte Carlo samples, we use $\tilde{\xi}_{1:(t-1)}$ to denote the actual data of ξ coming from steps 1 to $t-1$.

We have the following procedure:

Algorithm 2 Dynamic PCSG

Set F_0 as the prior distribution of μ. For $t = 1,\dots,T$: While $t \in \mathcal{S}$, and given F_t and the realized $x_{1:(t-1)}$ and $\tilde{\xi}_{1:(t-1)}$:

1. Repeat N_t times:
   (a) Generate $\mu^n \sim F_t$.
   (b) Generate $\xi^n \sim G_t|\mu^n$.
2. Impose the constraints

$$f_t(x_{1:(t-1)}, x_t, \tilde{\xi}_{1:(t-1)}, \xi^n) \in \mathcal{A}_t,\ \forall\, n = 1,\dots,N_t \tag{11}$$

   at stage t.

It is understood that in the second step of Dynamic PCSG, the constraints are imposed together with an appropriate objective function (typically the cost-to-go in formulation (3)) to form a stepwise optimization with decision variable x_t. The following result gives the choice of N_t and the resulting guarantee:

Theorem 3. *Suppose the stepwise safety conditions are all convex sets, the objective function at every step is concave, and $x_t \in \mathbb{R}^d$. Suppose also that any instance of the optimization resulted from imposing* (11) *belongs to $\mathcal{R}$. Suppose $0 \leq \beta_t, \delta_t \leq 1$ are constants such that*

$$\sum_{i=0}^{d-1} \binom{N_t}{i} (\alpha\beta_t)^i (1-\alpha\beta_t)^{N_t - i} \leq \delta_t \tag{12}$$

and

$$\sum_{t \in \mathcal{S}} (\beta_t + \delta_t - \beta_t\delta_t) \leq \beta\lambda \tag{13}$$

Then the policy obtained from Dynamic PCSG is feasible for (3) *under the updated posterior distribution with probability at least $1-\beta$, with overall Monte Carlo confidence $1-\lambda$, i.e.,*

$$P_{MC}(P_{\mu_{1:T}}(P_{\xi_t|\mu_t}(f_t(x_t,\xi_t) \in \mathcal{A}_t|\mathcal{F}_{t-1}) \geq 1-\alpha \\ \forall t \in \mathcal{S}) \geq 1-\beta) \geq 1-\lambda \tag{14}$$

where $P_{\mu_{1:T}}$ denotes the probability with respect to $\mu_1, \ldots, \mu_t$, where each $\mu_t \sim F_t$, the posterior distribution of μ at step t, and $P_{\xi_t|\mu_t}$ denotes the probability with respect to ξ_t given a realized parameter of μ_t.

Theorem 3 asserts that the round-specific statistical parameters, namely the posterior credibility $1-\beta_t$ and the Monte Carlo confidence level $1-\delta_t$, which determine the constraint sample size, can be chosen to satisfy a local condition (12) and a global condition (13) to achieve an overall statistical guarantee.

For convenience we can set $\beta_t = \delta_t$ and both equal to some constant, say γ_t. This γ_t can be set to be stage-independent or dependent. The following subsection shows two choices of γ_t.

3.4 TWO EXPLICIT STRATEGIES

We demonstrate two choices of $\{N_t\}$ in terms of $\{\gamma_t\}$. The first choice is a simple one that requires knowledge of the horizon length T, by setting γ_t to be a constant. The second choice uses a decaying γ_t, consequently an increasing sample size N_t, which does not require knowledge of T a priori. For convenience, we denote $|\mathcal{S}|$ as the size of the set $\mathcal{S}$. For the first strategy, we have:

Proposition 1. *Given a time horizon T, if we let $\beta_t = \delta_t = \gamma$ for all $t \in \mathcal{S}$ such that $\gamma \leq 1 - \sqrt{1-\beta\lambda/|\mathcal{S}|}$, then* (14) *holds.*

The following describes our second strategy that is stage-dependent such that (14) holds without knowing the horizon T or $|\mathcal{S}|$ a priori:

Proposition 2. *If we let $\beta_t = \delta_t = \gamma_t$ for all $t \in \mathcal{S}$ such that $\gamma_t = (1/\zeta(t)^\rho) \wedge \eta$, where $\rho > 1$, $0 < \eta < 1$, and $\zeta(t) = \sharp\{s \in \mathcal{S} : s \leq t\}$ (i.e., $\zeta(t)$ is the "counter" of t in $\mathcal{S}$) such that*

$$2\eta^{1-1/\rho} + \frac{2}{\rho-1}\frac{1}{(1/\eta^{1/\rho}-1)^{\rho-1}} - \frac{\eta^2}{\eta^{1/\rho}+1} \\ - \frac{1}{2\rho-1}\frac{1}{(1/\eta^{1/\rho}+1)^{2\rho-1}} \leq \beta\lambda \tag{15}$$

then (14) *holds regardless of $|\mathcal{S}|$.*

For example, if ρ is set to be 2, then (15) becomes $2\sqrt{\eta} + 2\sqrt{\eta}/(1-\sqrt{\eta}) - \eta^2/(\sqrt{\eta}+1) - \eta^{3/2}/(3(1+\sqrt{\eta})^3) \leq \beta\lambda$.

4 INTEGRATION INTO THOMPSON SAMPLING

We illustrate the integration of our strategies with Thompson sampling, which also operates via Bayesian updating, by an example of revenue maximization in online advertising (Pani et al., 2017). The advertiser is interested in maximizing the expected revenue across a portfolios of keywords or biddable ad units while ensuring that the budget constraint is not violated. When the advertiser selects a bid value for a keyword it results in ad clicks, the volume of which is stochastic. The distribution of clicks and the associated revenue is not initially known to the decision-maker and needs to be learned over time. Further, the cost associated with the choice of a bid is also unknown and hence, there is uncertainty regarding how the budget will be affected.

To be more concrete, consider a set of K bid values $\{\kappa_1, ..., \kappa_K\}$, for M items labeled $\{\pi_1, ..., \pi_M\}$, over the campaign horizon T. Bidding value j on item i will induce an average revenue r_{ij} and cost c_{ij} respectively. These quantities are assumed to follow independent Poisson distributions with initially unknown parameters (the Poisson assumptions come from the click count nature). In each period $t = 1, \ldots, T$, the advertiser picks a bid value j from every item, observes the outcome, i.e., the realizations of $r_{ij}, c_{ij}, i = 1, \ldots, M$. She gains $\sum_{i=1}^M \sum_{j=1}^K r_{ij}x_{ij}$ and consumes $\sum_{i=1}^M \sum_{j=1}^K c_{ij}x_{ij}$ from the budget, where x_{ij} is the allocation portion for bid value j of item i (i.e., the fraction of time or the probability in a randomized scheme that is allocated to this particular bid value and item).

The advertiser's goal is to maximize the total revenue while maintain a budget constraint with high probability. In other words, letting $r_{ij}(t)$ and $c_{ij}(t)$ be the realized revenue and cost for a bid at time t, and $x_{ij}(t)$ be the corresponding allocation variables, she wants to maximize $E[\sum_{t=1}^T \sum_{i=1}^M \sum_{j=1}^K r_{ij}(t)x_{ij}(t)]$. A typical

budget constraint is a bound given by the remaining budget averaged over the remaining horizon. Specifically, let the overall budget be B. Denoting $B(t-1) = B - \sum_{u=1}^{t-1}\sum_{i=1}^{M}\sum_{j=1}^{K} c_{ij}(u)x_{ij}(u)$ as the remaining budget before epoch $t \in \mathcal{S}$, the advertiser wants to keep $P(\sum_{u=1}^{t}\sum_{i=1}^{M}\sum_{j=1}^{K} c_{ij}(u)x_{ij}(u) \leq B(t-1)/(T-t+1)) \geq 1-\alpha\ \forall t \in \mathcal{S}$. This type of dynamically updated per-round budgets is common in practice and is argued to be more effective than fixed per-round budgets. In the following, we will concentrate on the particular choice described above as the feasibility requirement.

4.1 A BUDGETED ALGORITHM

The setting of this problem resembles a recent study (Ferreira et al., 2016) on a network revenue management problem. They developed a Thompson sampling algorithm to sequentially assign a price vector to items under resource constraints, where each step involves a knapsack optimization problem. Here we present a variant of their algorithm to suit our setting (Algorithm 3). This initial algorithm does not take into account the possibility of constraint violation; the idea is to later illustrate how our Dynamic PCSG strategy can be integrated.

Denote by $X_{ij}(t-1)$ the allocation units on bid value j for item i cumulated in the first $t-1$ rounds, and denote by $W_{ij}^{r}(t-1)$ and $W_{ij}^{c}(t-1)$ the total revenue and cost generated by assigning bid value j to item i during these periods respectively. In Algorithm 3, the advertiser samples from the joint posterior distributions of θ_{ij}, the unknown Poisson rate of the revenue, and μ_{ij}, the rate of the cost, corresponding to bid value j for item i. We put independent Gamma prior distributions for these parameters and hence the posterior distributions are also independent. The posterior samples of these parameters are then used in a linear program to decide the allocation. This algorithm follows quite intuitively from standard Thompson sampling, in which one generates posterior samples for the unknown parameters, and use them as "plug-in" to solve stage-wise optimization problems.

4.2 CHANCE-CONSTRAINED BUDGETED THOMPSON SAMPLING

We want to ensure $P(\sum_{u=1}^{t}\sum_{i=1}^{M}\sum_{j=1}^{K} c_{ij}(u)x_{ij}(u) \leq B(t-1)/(T-t+1)) \geq 1-\alpha\ \forall t \in \mathcal{S}$ holds with posterior credibility $1-\beta$. To achieve this, we integrate our Dynamic PCSG into Step 2 of Algorithm 3 by restructuring the involved optimization program. Algorithm 4 shows Dynamic PCSG in this particular setting.

The output of this procedure is a set of constraints, which will be used in the linear program of the budgeted

Algorithm 3 Budgeted Thompson Sampling for deterministically constrained problems adopted from Ferreira et al. (2016)

Given a total budget $B(0) = B$. For $t = 1, ..., T$, do the following:

1: For each bid value j and each item i, sample θ_{ij} from $Gamma(W_{ij}^{r}(t-1)+1, X_{ij}(t-1)+1)$ and μ_{ij} from $Gamma(W_{ij}^{c}(t-1)+1, X_{ij}(t-1)+1)$.

2: Solve the following linear program:

$$\begin{aligned} \max_{x} \quad & \sum_{j=1}^{K}\sum_{i=1}^{M} \theta_{ij}x_{ij} \\ \text{subject to} \quad & \sum_{j=1}^{K}\sum_{i=1}^{M} \mu_{ij}x_{ij} \leq \frac{B(t-1)}{T-t+1} \\ & \sum_{j=1}^{K} x_{ij} \leq 1,\ \forall\, i = 1, \ldots, M \\ & x_{ij} \geq 0,\ \forall\, i = 1, \ldots, M,\ j = 1, \ldots, K \end{aligned}$$

to obtain $(x_{ij}^{*}(t))_{i=1,\ldots,M, j=1,\ldots,K}$.

3: The revenue, $r_{ij}(t)$, and the cost, $c_{ij}(t)$, generated by assigning bid value j on item i are revealed. We update $X_{ij}(t) = X_{ij}(t-1) + x_{ij}^{*}(t)$, $W_{ij}^{r}(t) = W_{ij}(t-1) + r_{ij}(t)x_{ij}^{*}(t)$, $W_{ij}^{c}(t) = W_{ij}^{c}(t-1) + c_{ij}(t)x_{ij}^{*}(t)$ and $B(t) = B(t-1) - \sum_{j=1}^{K}\sum_{i=1}^{M} c_{ij}(t)x_{ij}^{*}(t)$.

4: If $B(t) \leq 0$, the algorithm terminates.

Thompson sampling. Algorithm 5 shows how Dynamic PCSG can be integrated into Algorithm 3. The following is an immediate consequence of Theorem 3:

Corollary 4. *Suppose that any instance of the sampled program in* (16) *belongs to* $\mathcal{R}$. *Suppose* $0 \leq \beta_t, \delta_t \leq 1$ *are chosen to satisfy*

$$\sum_{i=0}^{KM-1} \binom{N_t}{i} (\alpha\beta_t)^{i}(1-\alpha\beta_t)^{N_t-i} \leq \delta_t$$

and

$$\sum_{t\in\mathcal{S}} (\beta_t + \delta_t - \beta_t\delta_t) \leq \beta\lambda$$

Consider a modification of Algorithm 5 such that at any point of time, if the total budget B is fully depicted, we refill the shortfall and add extra budget B (so that the total remaining budget in the next step returns to the full level B). Then the sequence of decisions obtained will satisfy

$$P_{\boldsymbol{\mu}}\left(P_{\mathbf{c}_t|\boldsymbol{\mu}_t}\left(\sum_{u=1}^{t}\sum_{i=1}^{M}\sum_{j=1}^{K} c_{ij}(u)x_{ij}(u) \leq \frac{B(t-1)}{T-t+1} \right) \geq 1-\alpha\ \forall t \in \mathcal{S} \right) \geq 1-\beta$$

Algorithm 4 Dynamic PCSG for the Bidding Problem

1. Set $F_{ij}(0), i = 1, \ldots, M, j = 1, \ldots, K$ as the prior distribution of μ_{ij}. For $t = 1, \ldots, T$: Given $B(t-1)$ and $F_{ij}(t-1)$, the posterior distribution of μ_{ij} given $\mathcal{F}_{t-1}$,
   (a) Repeat N_t times:
      i. Generate $\mu_{ij}^n \sim F_{ij}(t-1)$ independently for each item $i = 1, \ldots, M$ and bid value $j = 1, \ldots, K$.
      ii. Generate $\xi_{ij}^n \sim Poisson(\mu_{ij}^n)$ for each item $i = 1, \ldots, m$ and bid value $j = 1, \ldots, K$.
   (b) Form the constraints

$$\sum_{j=1}^{K}\sum_{i=1}^{M} \xi_{ij}^n x_{ij} \leq \frac{B(t-1)}{T-t+1}, \ \forall\, n = 1, \ldots, N_t$$

with Monte Carlo confidence level at least $1-\lambda$. $P_{\boldsymbol{\mu}}$ *denotes the probability of* $\{\boldsymbol{\mu}_t\}_{t=1,\ldots,T}$*, where* $\boldsymbol{\mu}_t$ *is the collection* $(\mu_{ij})_{i,j}$ *and each element is distributed independently according to the posterior distribution* $F_{ij}(t-1)$*, and* $P_{\mathbf{c}_t|\boldsymbol{\mu}_t}$ *denotes the probability with respect to the collection* $(c_{ij})_{i,j}$ *given the realization of* $(\mu_{ij})_{i,j}$.

We mention that the "modification of Algorithm 5" introduced in Corollary 4 is only a technicality that takes care of the unusual situation when the entire available budget is prematurely depleted. Since we divide the remaining budget by the remaining horizon (a common practice to set per-round budgets) to form our constraint at each step, the scenario of total budget depletion before the last step rarely happens.

4.3 NUMERICAL RESULTS

We examine the empirical performance of our proposed strategy on a synthetic dataset with two items and three bid values ($M = 2$, $K = 3$) over the time horizon $T = 100$. The cost and revenue of each item-bid value pair follow Poisson distributions with parameters taken uniformly from an interval that is calibrated from a real data set owned by a prominent tech firm (blinded for peer-review purpose). We test with five different values of the overall budget $B = (a\sum_{i=1}^{M}\bar{\rho}_i) \times T$ where $a \in [0.5, 0.75, 1, 1.25, 1.5]$ and $\bar{\rho}_i$ is the average cost of item i over the K bid values. The choice of B roughly matches the scale of the total cost over the time. The per-round budget is defined as the remainder of the overall budget divided by the remaining number of rounds.

To test our chance-constrained Thompson sampling (CCTS) in Algorithm 5, we use three different settings for $\mathcal{S}$, i.e. $\mathcal{S}_1 = \{25, 50, 75\}$, $\mathcal{S}_2 = \{20, 40, 60, 80, 100\}$ and $\mathcal{S}_3 = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. We enforce $\alpha = 0.1$, $\beta = \lambda = 0.3$, and $\beta_t = \delta_t = \gamma$ where γ is taken as the upper bound depicted in Proposition 1. With these configurations, the number of constraints in the involved linear programs are typically in the range of thousands, which gives a run-time of a few minutes in solving the decisions for the whole horizon using our sever machine. Note that, if we consider our online problem a daily problem (common in practice) then this solution time is quite acceptable as we have hours to solve the problem at each stage. Moreover, the number of constraints and hence the run-time can be further reduced by using more recent advances in the constraint sampling literature as depicted at the end of Section 3.1.

For each considered setting, we conduct 500 simulation runs. For each run, we estimate the proportion of violation of the decision using 100 inner repetitions of ξ_t, at each step $t \in \mathcal{S}$. Figure 1 depicts the box-plots showing the distribution of the proportion of violation. For all the tested budget levels and choices of $\mathcal{S}$, CCTS was able to maintain the proportion of budget violation well below the 10% tolerance at the relevant steps. This implies, moreover, that the overall violation (i.e., at least

Algorithm 5 Chance-constrained Thompson Sampling (CCTS)

Initialize $\alpha, \beta_t, \delta_t \in [0, 1]$ satisfying (13). Given a total budget $B(0) = B$. For $t = 1, ..., T$, do the following:

1: For each bid value j and each item i, sample θ_{ij} from $Gamma(W_{ij}^r(t-1)+1, X_{ij}(t-1)+1)$ and μ_{ij} from $Gamma(W_{ij}^c(t-1)+1, X_{ij}(t-1)+1)$.

2: Run Dynamic PCSG using N_t samples to get the constraints

$$\sum_{j=1}^{K}\sum_{i=1}^{M} \xi_{ij}^n x_{ij} \leq \frac{B(t-1)}{T-t+1}, \ \forall\, n = 1, \ldots, N_t$$

3: Solve the following linear program:

$$\begin{aligned} \max_{x} \quad & \sum_{j=1}^{K}\sum_{i=1}^{M} \theta_{ij} x_{ij} \\ \text{subject to} \quad & \sum_{j=1}^{K}\sum_{i=1}^{M} \xi_{ij}^n x_{ij} \leq \frac{B(t-1)}{T-t+1}, \ \forall\, n = 1, \ldots, N_t \\ & \sum_{j=1}^{K} x_{ij} \leq 1, \ \forall\, i = 1, \ldots, M \\ & x_{ij} \geq 0, \ \forall\, i = 1, \ldots, M, \ j = 1, \ldots, K \end{aligned} \tag{16}$$

to obtain $(x_{ij}^*(t))_{i=1,\ldots,M,j=1,\ldots,K}$.

4: The revenue, $r_{ij}(t)$, and the cost, $c_{ij}(t)$, generated by assigning bid value j on item i are revealed. We update $X_{ij}(t) = X_{ij}(t-1) + x_{ij}^*(t)$, $W_{ij}^r(t) = W_{ij}(t-1) + r_{ij}(t)x_{ij}^*(t)$, $W_{ij}^c(t) = W_{ij}^c(t-1) + c_{ij}(t)x_{ij}^*(t)$ and $B(t) = B(t-1) - \sum_{j=1}^{K}\sum_{i=1}^{M} c_{ij}(t)x_{ij}^*(t)$.

5: If $B(t) \leq 0$, the algorithm terminates.

one violation at a step in $\mathcal{S}$) is also below 10%.

Note that the theoretical guarantees studied in the previous sections focus on the feasibility in maintaining the chance constraints. In practice, the objective value performance is also important. To test this, we compare the performance of CCTS, both regarding budget violation and revenue attainment, against the following algorithms: 1) a hypothetical algorithm that assumes the distributions of the costs and revenues are all known and draws Monte Carlo samples from it, otherwise the same as CCTS; 2) the deterministically constrained Thompson sampling (DCTS) in Algorithm 3; 3) the algorithm in Badanidiyuru et al. (2013) that uses reward-to-cost ratios; and 4) Besbes and Zeevi (2012) that uses an initial learning phase (in our experiment we set the learning phase to 50 steps). Figure 2 shows the distributions of the proportion of violations at $t \in \mathcal{S}_2$ based on 500 simulation runs, for the five described budget levels. We see that only CCTS and the hypothetical algorithm maintains the proportion of violation to well below 10%, whereas the other three algorithms fluctuate around 20-40$\%$, as they do not account for the chance constraint on the per-round budget. On the other hand, Figure 3 shows the average cumulative revenue achieved through the horizon (the bar depicts one standard deviation around the average). DCTS appears the best in terms of cumulative rewards, and Badanidiyuru et al. (2013) and Besbes and Zeevi (2012) perform similarly. CCTS achieves less rewards (around 15%), which can be viewed as the price of maintaining the chance constraint. The hypothetical algorithm performs better than CCTS, not surprisingly given the full distributional knowledge. This behavior persists for $\mathcal{S}_1$ and $\mathcal{S}_3$ as well as for several priors we have tested (similar to plots Figures 2 and 3). Thus, in view of achieving overall performances in terms of both controlling violation proportions and attaining cumulative rewards, our CCTS appears to be superior to all the other considered methods.

The above experiments all used the budget violations calculated using the true underlying distribution of the cost. In Table 1, we compare with the calculation based on the evolving posterior distributions, in terms of the average of all the proportions of violations for $t \in \mathcal{S}_1, \mathcal{S}_2$ and $\mathcal{S}_3$ at the five described budget levels. The average proportion of violation based on the posterior distribution is consistently lower than the one based on the true distribution for all budget levels and $\mathcal{S}$. This is expected since our chance constraint is maintained under the posterior distribution (as Theorem 3 states). However, the proportion of violations is maintained below 10% under the true distribution, thanks to the relatively robust performance of our approach in abiding with the chance constraint.

Table 1: Comparison of the average proportions of violations based on the true distribution and the updated posterior distributions over 500 simulation runs

Budget		$\mathcal{S}_1$	$\mathcal{S}_2$	$\mathcal{S}_3$
$B1$	True Dist.	0.018	0.019	0.018
	Posterior Dist.	0.011	0.012	0.009
$B2$	True Dist.	0.016	0.015	0.014
	Posterior Dist.	0.01	0.01	0.008
$B3$	True Dist.	0.014	0.013	0.013
	Posterior Dist.	0.008	0.009	0.007
$B4$	True Dist.	0.013	0.011	0.01
	Posterior Dist.	0.009	0.007	0.005
$B5$	True Dist.	0.008	0.008	0.007
	Posterior Dist.	0.006	0.005	0.004

Alternatively, we also investigated the amount of budget violation at $t \in \mathcal{S}$. Figure 4 depicts the distributions of the total amounts of budget violation in $\mathcal{S}_1, \mathcal{S}_2$ and $\mathcal{S}_3$ for the five budget levels over 500 simulation runs. Similar to the proportion of budget constraint violations, the average amounts of violation for CCTS and DTS tend to be much lower than the other three algorithms, with at most 25% of those of the other three algorithms in the same setting. This suggests a strong dependence between the proportion and the amount of violation which substantiates the use of CCTS in maintaining over-spending even in the monetary scale.

5 CONCLUSION

We studied sequential learning subject to constraints that need to be satisfied with high probability. We investigated a methodology to obtain posterior statistical guarantees for the feasibility of these constraints, by generalizing the constraint sampling approach in chance-constraint programming to a two-level Monte Carlo procedure and analyzing the sample size needed in achieving overall feasibility through the learning horizon. We further incorporated our scheme into Thompson sampling using an online advertisement example, and numerically demonstrated how it led to desirable performances in both feasibility and optimality. As far as we know, this work represents the first methodological investigation of "soft" stochastic constraints in sequential learning. In subsequent work, we will investigate the tightening of the requirements in sampled constraints, via for instance analyzing the correlation among decisions at different stages, and will also study the scalability of this approach to higher-dimensional problems.

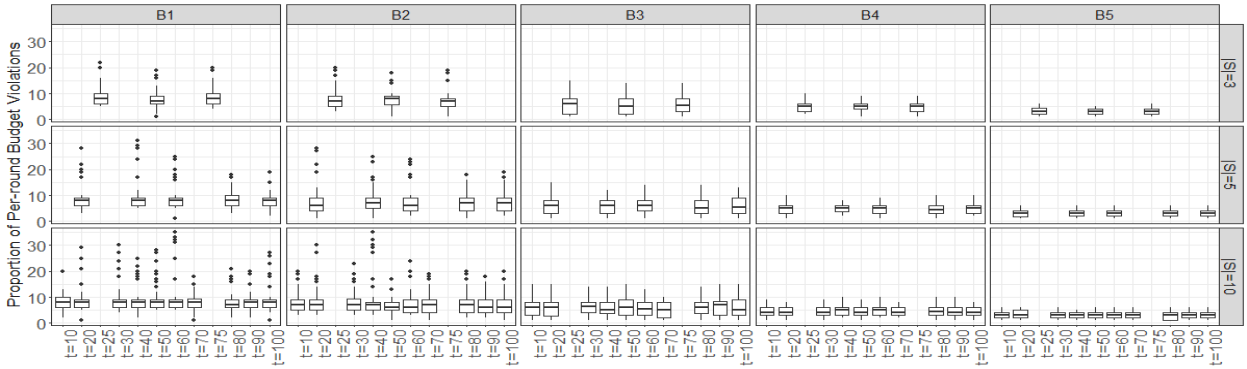

Figure 1: Estimated proportion of violation at each step in $\mathcal{S}$, for $\mathcal{S} = \mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ and five different budget levels, using 500 simulation runs under CCTS.

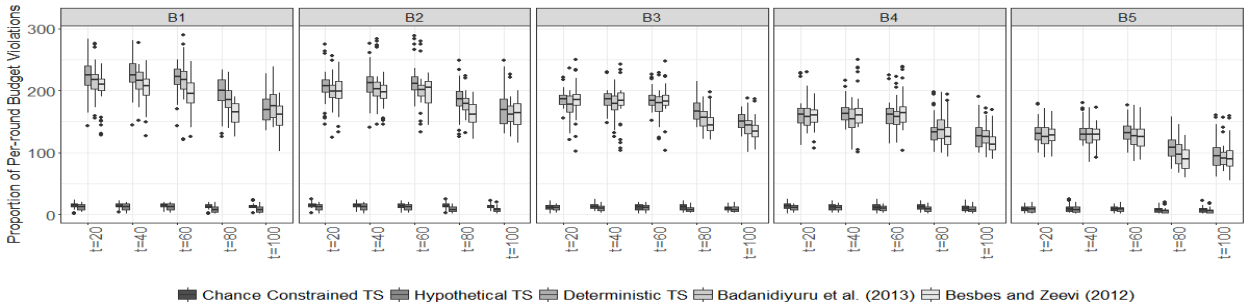

Figure 2: Estimated proportion of violation at each step in $\mathcal{S}_2$, with 500 simulation runs for different algorithms

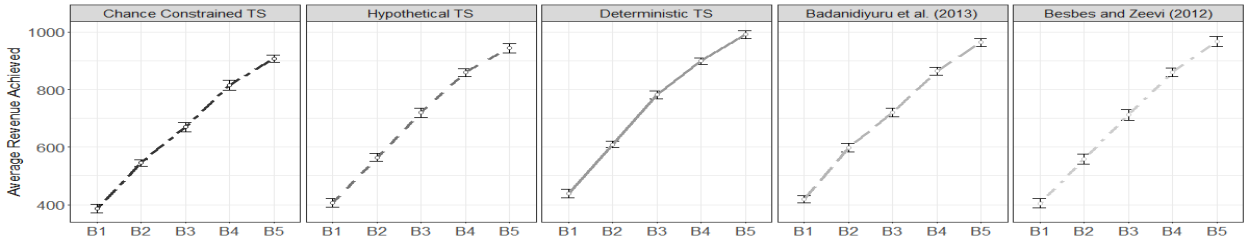

Figure 3: Average cumulative revenue over $T = 100$ achieved by different algorithms under $\mathcal{S}_2$

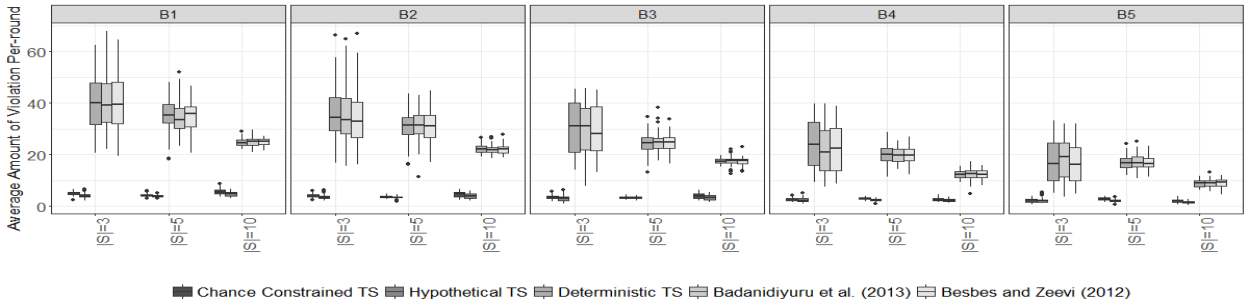

Figure 4: Total Amount of budget violations occurred over $\mathcal{S}_1, \mathcal{S}_2$ and $\mathcal{S}_3$ for different algorithms

Acknowledgements

Support from the Adobe Faculty Research Award is gratefully acknowledged.

References

Agrawal, S. and Goyal, N. (2012). Analysis of thompson sampling for the multi-armed bandit problem. In

COLT, pages 39.1–39.26.

Amin, K., Kearns, M., Key, P., and Schwaighofer, A. (2012). Budget optimization for sponsored search: Censored learning in mdps. *arXiv preprint arXiv:1210.4847*.

Audibert, J.-Y. and Bubeck, S. (2010). Best arm identification in multi-armed bandits. In *COLT*, pages 41–53.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

Badanidiyuru, A., Kleinberg, R., and Slivkins, A. (2013). Bandits with knapsacks. In *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*, pages 207–216. IEEE.

Ben-Tal, A. and Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming*, 88(3):411–424.

Bertsimas, D., Gupta, V., and Kallus, N. (2013). Data-driven robust optimization. *arXiv preprint arXiv:1401.0212*.

Besbes, O. and Zeevi, A. (2012). Blind network revenue management. *Operations Research*, 60(6):1537–1550.

Boesel, J., Nelson, B. L., and Kim, S.-H. (2003). Using ranking and selection to clean up after simulation optimization. *Operations Research*, 51(5):814–825.

Calafiore, G. and Campi, M. C. (2005). Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102(1):25–46.

Calafiore, G. C. (2017). Repetitive scenario design. *IEEE Transactions on Automatic Control*, 62(3):1125–1137.

Calafiore, G. C. and Campi, M. C. (2006). The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753.

Calafiore, G. C. and El Ghaoui, L. (2006). On distributionally robust chance-constrained linear programs. *Journal of Optimization Theory and Applications*, 130(1):1–22.

Campi, M. C. and Garatti, S. (2008). The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230.

Campi, M. C. and Garatti, S. (2011). A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *Journal of Optimization Theory and Applications*, 148(2):257–280.

Carè, A., Garatti, S., and Campi, M. C. (2014). Fast-fast algorithm for the scenario technique. *Operations Research*, 62(3):662–671.

Chamanbaz, M., Dabbene, F., Tempo, R., Venkataramanan, V., and Wang, Q.-G. (2016). Sequential randomized algorithms for convex optimization in the presence of uncertainty. *IEEE Transactions on Automatic Control*, 61(9):2565–2571.

Charnes, A., Cooper, W. W., and Symonds, G. H. (1958). Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Management Science*, 4(3):235–263.

De Farias, D. P. and Van Roy, B. (2004). On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, 29(3):462–478.

Ding, W., Qin, T., Zhang, X.-D., and Liu, T.-Y. (2013). Multi-armed bandit with budget constraint and variable costs. In *AAAI*, pages 232–238.

Ferreira, K. J., Simchi-Levi, D., and Wang, H. (2016). Online network revenue management using thompson sampling. *Working paper*.

Lagoa, C. M., Li, X., and Sznaier, M. (2005). Probabilistically constrained linear programs and risk-adjusted controller design. *SIAM Journal on Optimization*, 15(3):938–951.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22.

Lejeune, M. A. and Ruszczynski, A. (2007). An efficient trajectory method for probabilistic production-inventory-distribution problems. *Operations Research*, 55(2):378–394.

Luedtke, J., Ahmed, S., and Nemhauser, G. L. (2010). An integer programming approach for linear programs with probabilistic constraints. *Mathematical Programming*, 122(2):247–272.

Miller, B. L. and Wagner, H. M. (1965). Chance constrained programming with joint constraints. *Operations Research*, 13(6):930–945.

Pani, A., Raghavan, S., and Sahin, M. (2017). Large-scale advertising portfolio optimization in online marketing.

Prékopa, A. (2003). Probabilistic programming. *Handbooks in operations research and management science*, 10:267–351.

Rossi, R., Hnich, B., Tarim, S. A., and Prestwich, S. (2011). Finding (α,ϑ)-solutions via sampled scsp. In *IJCAI*, pages 2172–2177.

Rossi, R., Hnich, B., Tarim, S. A., and Prestwich, S. (2015). Confidence-based reasoning in stochastic constraint programming. *Artificial Intelligence*, 228:129–152.

Russo, D. and Van Roy, B. (2016). An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471.

Shi, Y., Zhang, J., and Letaief, K. B. (2015). Optimal stochastic coordinated beamforming for wireless cooperative networks with csi uncertainty. *IEEE Transactions on Signal Processing*, 63(4):960–973.

Tran-Thanh, L., Chapman, A., Rogers, A., and Jennings, N. R. (2012). Knapsack based optimal policies for budget-limited multi-armed bandits. In *AAAI*, pages 1134–1140.

Tran-Thanh, L., Stavrogiannis, L., Naroditskiy, V., Robu, V., Jennings, N. R., and Key, P. (2014). Efficient regret bounds for online bid optimisation in budget-limited sponsored search auctions. In *UAI*, pages 809–818.

Villar, S. S., Bowden, J., and Wason, J. (2015). Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2):199.

Wang, Z., Deng, S., and Ye, Y. (2014). Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research*, 62(2):318–331.

Xia, Y., Li, H., Qin, T., Yu, N., and Liu, T.-Y. (2015). Thompson sampling for budgeted multi-armed bandits. In *IJCAI*, pages 3960–3966.

Zymler, S., Kuhn, D., and Rustem, B. (2013). Distributionally robust joint chance constraints with second-order moment information. *Mathematical Programming*, pages 1–32.