17

Input Uncertainty in Stochastic Simulation

Russell R. Barton, Henry Lam, and Eunhye Song

17.1 Introduction

A stochastic simulator refers to a computer model that takes random inputs in and generates outputs by following a set of deterministic system rules. The simulation outputs are collected and used to estimate a performance measure of interest. For instance, a simple queueing simulator may prescribe the system rules on how jobs are processed by servers, where the goal is to estimate the expected waiting time of jobs in the queue. The inputs to the simulator consist of interarrival times of jobs and service times of the servers. These inputs are typically generated from probability distributions referred to as *input models*. According to the system rules, the simulator calculates each job's waiting time in the queue from the inputs and returns it as an output. Simulating many such jobs, the expected waiting time can be estimated

R. R. Barton (✉) · E. Song
Penn State University, University Park, PA, USA
e-mail: rrb2@psu.edu

E. Song
e-mail: eus358@psu.edu

H. Lam
Columbia University, New York, NY, USA
e-mail: khl2114@columbia.edu

S. Salhi and J. Boylan (eds.), *The Palgrave Handbook of Operations Research*,
https://doi.org/10.1007/978-3-030-96935-6_17

by averaging the outputs. Typically, the estimated performance measure is subject to stochastic error since one can only generate finitely many simulation outputs. As the simulation sample size increases, the hope is that the estimated performance measure converges to the true performance measure in some probabilistic sense.

When the simulator is built to mimic a real-world system, the input models may be estimated from *input data* (e.g., interarrival and service times) collected from the system. Then, there is statistical error in estimating the input models from the finite input data. As a result, any simulation outputs computed from the inputs generated from these models are subject to the estimation error. In turn, the performance measure estimate now has additional uncertainty caused by the input-model estimation error. The latter is referred to as *input uncertainty* and is the main focus of this chapter.

To summarize, the two sources of stochastic variability in the performance measure estimate are: (i) the finiteness of the simulation output sample, and (ii) the finiteness of the input data used to fit the input models. These two sources of variability in the simulation literature have been given a number of different names. Perhaps most intuitive are *simulation variability* and *parameter variability* by Cheng and Holland [26], with the former characterizing uncertainty or error in a (deterministic) function of the simulation output random variable due to the stochastic nature of the output, and the latter characterizing uncertainty in estimated input model's parameters. Other names for simulation variability include *simulation error*, *Monte Carlo error*, *variance error*, *stochastic uncertainty*, *sampling error*, and *intrinsic output uncertainty*. Other terms used for parameter variability (and beyond to cover other errors stemming from input fitting) include *input uncertainty*, *input-model uncertainty*, *bias error*, and *extrinsic error*. In the Bayesian setting, *structural uncertainty* captures both model uncertainty (probability model forms and system logic) and probability model parameter uncertainty. See [6, 7, 28, 29, 63, 75, 92, 120] and Chapter 7 of [92] for additional information. For this chapter we will use *Monte Carlo error* and *input uncertainty* to name the two sources of variability in simulation output.

We distinguish analyzing input uncertainty from *uncertainty quantification* (UQ) of a computer model. For UQ, the computer models typically are differential equation-based and have deterministic outputs; the uncertainty is in the values of model parameters. For instance, [121] perform UQ of a fire-spread model, which calculates the spread speed of forest fire based on a set of differential equations. Here, the source of uncertainty is the unknown values of wind speed, moisture level in the air, size of the fuel in the forest, etc. Closely related to UQ is the topic of *sensitivity analysis*,

again typically associated with differential equation models. See [107] for an overview. Examples of this work include [94, 97] and, more recently, [95]. A main distinction of input uncertainty from computer model UQ is the stochasticity of the considered model, which adds variability on top of real data noise that complicates estimation. Another distinction is the type of data used. Input uncertainty often considers the availability of observations that directly inform the input distributions while UQ can involve output-level data [11, 33, 70, 124]. The latter, which is sometimes known as calibration [64, 125, 134] or simulation-based inference [32], has nonetheless been also considered in the stochastic simulation literature in the context of model validation [4, 73, 108, 112] and more recently [58, 59, 101], though still relatively open.

Concepts closely related to Monte Carlo error and input uncertainty from the UQ setting are named *aleatory uncertainty* and *epistemic uncertainty* [114]. *Aleatory* refers to inherent randomness in the output, and this variability cannot be reduced by better modeling or using more data. On the other hand, *epistemic* refers to lack of knowledge of model parameter value and model bias, and can be reduced by modeling or more data. While these terms have erudite philosophical roots, they are not as widely known in stochastic simulation and it would benefit to have more systematic connections.

Another related topic is *robust analysis*. We review some of robust analysis methods applied to stochastic simulation. See for example, [42, 53, 66, 75, 99].

17.2 Characterizing Input Uncertainty

Consider the estimation of a performance measure ψ that depends on the input distributions $\mathbf{P} = (P_1, \ldots, P_m)$, where $P_1, \ldots, P_m$ denote the individual distributions for independent random sources. For instance, in queueing models, P_1 and P_2 can denote the interarrival time and service time distributions, respectively. Sets of dependent random sources can be considered to be captured by multivariate distribution P_j of $\mathbf{P}$. The performance measure ψ, given $\mathbf{P}$, can be evaluated with error via simulation runs, i.e., we can generate $\widehat{\psi}_r, r = 1, \ldots, R$ where R is the simulation budget, and output the average $\widehat{\psi} = (1/R) \sum_{r=1}^{R} \widehat{\psi}_r$ as a natural point estimate of ψ. We call $\widehat{\psi} - \psi$ Monte Carlo error.

Input uncertainty arises when the ground-truth $\mathbf{P}$ is not known but only observed via data. Thus, $\mathbf{P}$ needs to be estimated, and this statistical error can propagate to the output and affect the accuracy of the point estimate of ψ.

Typically, the data set to inform each P_i is assumed independent of others and comprises i.i.d. observations (though can be generalized to serial dependence with sufficient mixing). In the parametric regime, we assume P_i is from a parametric family, say P_θ with unknown true parameter vector θ. In this case, an estimate of P_i reduces to estimating θ. In the nonparametric regime, no parametric assumption is placed on P_i, and a natural estimate of P_i is the empirical distribution (or its smoothed variants).

While simulation output analysis in general may consider any characterization of the probability distribution of simulation output, $\widehat{\psi}$, there are two main, and closely related, approaches to quantify input uncertainty. First is the construction of confidence interval (CI) on ψ that accounts for both the input uncertainty and Monte Carlo error. Second is the estimation of the variance of the point estimate $\widehat{\psi}$, which involves decomposition of the variance into the input uncertainty variance and Monte Carlo error variance components. The input uncertainty variance component is typically more difficult to estimate than the Monte Carlo variance, as the latter can be quite readily estimated by using the sample variance of replicate simulation runs. The former, however, not only involves a variance of a nonlinear functional, but also can only be evaluated with added Monte Carlo error. The CI approach and the variance estimation approach are closely connected as the variance estimate provides the standard error for a variance-based (as opposed to quantile-based) CI. Note, however, that when ψ is a steady-state measure such as average queue length or average time in system, $\widehat{\psi}$ may have infinite expectation and its variance is undefined [111]. In this case CIs may still be obtained, but they cannot be variance-based.

17.3 Confidence Interval Construction and Variance Estimation

A $(1-\alpha)$-level CI for ψ is an interval $[L, U]$ that satisfies

$$\mathbb{P}(\psi \in [L, U]) \geq 1 - \alpha$$

where $\mathbb{P}$ refers to the probability with respect to both input data and Monte Carlo runs. We say that a CI is asymptotically valid if

$$\liminf_{n\to\infty} \mathbb{P}(\psi \in [L, U]) \geq 1 - \alpha$$

for some scaling parameter n. We call CI asymptotically exact if

$$\lim_{n\to\infty} \mathbb{P}(\psi \in [L, U]) = 1 - \alpha.$$

Besides coverage, the efficiency of the CI measured by the length of the interval (in expectation) is also important. Obviously, the infinite interval $L = -\infty, U = \infty$ is asymptotically valid but not useful.

As will be described in further detail in the following sections, we typically have

$$\mathrm{Var}(\widehat{\psi}_{point}) = V_{IU} + V_{MC} \tag{17.1}$$

where $\widehat{\psi}_{point}$ is a natural point estimate of ψ, by "plugging in" the point estimate of the input parameter and conducting simulation runs based on the resulting input model, and Var refers to the variance from both input uncertainty and Monte Carlo error. (17.1) implies that the variance of the natural point estimate of ψ can be decomposed into an input uncertainty variance component, V_{IU}, and a simulation Monte Carlo error variance component V_{MC}. The variance estimation approach in input uncertainty often refers to the estimation of this decomposed variance, in particular V_{IU} which is the more challenging piece as mentioned before.

CIs and variance estimation are closely connected. Under suitable regularity conditions, not only (17.1) holds, but also we have a central limit theorem (CLT) for $\widehat{\psi}_{point}$ such that it is approximately $N(\psi, \mathrm{Var}(\widehat{\psi}_{point}))$. Thus, to construct a CI for ψ, it suffices to estimate the variance of $\widehat{\psi}_{point}$ and then use

$$[L, U] = \left[\widehat{\psi}_{point} - z_{1-\alpha/2}\sqrt{\mathrm{Var}(\widehat{\psi}_{point})}, \widehat{\psi}_{point} + z_{1-\alpha/2}\sqrt{\mathrm{Var}(\widehat{\psi}_{point})}\right]$$

where $\mathrm{Var}(\widehat{\psi}_{point})$ is plugged in with the variance estimate, and $z_{1-\alpha/2}$ denotes the $(1-\alpha/2)$-level normal critical value, given by the $(1-\alpha/2)$-quantile of standard normal.

In the following, we divide our review into methods that primarily apply to parametric regimes (i.e., when the input model is specified up to a parametric family with unknown parameters), and methods that can handle

nonparametric settings (which typically are also applicable to parametric cases). Moreover, we will focus on the problem of CI construction as, in view of our discussion above, variance estimation often serves as an intermediate step in obtaining the CI.

17.3.1 Parametric Methods

When the input probability distributions are assumed to have a parametric form, the uncertainty characterization is simplified. For each P_j, rather than dealing with an unknown functional form (restricted to the class of probability distribution functions), the uncertainty is over a finite set of parameters that define a particular member of the assumed parametric family. While easier to handle, parametric assumption should be viewed with caution: In real-world systems, random phenomena rarely follow any parametric distributions (or mixtures) exactly [20], and sometimes this error ignored by the existing frequentist and Bayesian input uncertainty quantification methods described below could be substantial.

17.3.1.1 Central Limit Theorem and Delta Method

To highlight the dependence on θ, we denote $\psi = \psi(\theta)$ where $\psi(\cdot)$ is a map from the input parameter to the performance measure value. From input data, we give an estimate of the ground-truth parameter vector θ given by $\hat{\theta}$. Then, given $\hat{\theta}$, we estimate $\psi(\hat{\theta})$ by running and averaging R simulation runs, each denoted by $\widehat{\psi}_r(\theta)$, to get $\widehat{\psi}(\hat{\theta})$. This point estimate $\widehat{\psi}(\hat{\theta})$ is contaminated by both the input and Monte Carlo noises, with the two "hats" indicating the two sources of noises.

From the decomposition

$$\widehat{\psi}(\hat{\theta}) - \psi(\theta) = [\widehat{\psi}(\hat{\theta}) - \psi(\hat{\theta})] + [\psi(\hat{\theta}) - \psi(\theta)]$$

it can be inferred that

$$\widehat{\psi}(\hat{\theta}) - \psi(\theta) \approx N\left(0, \mathrm{Var}(\psi(\hat{\theta})) + \frac{\mathrm{Var}(\widehat{\psi}_r(\theta))}{R}\right), \tag{17.2}$$

where $\mathrm{Var}(\psi(\hat{\theta}))$ is the input variance, with Var on the randomness of input data, and $\mathrm{Var}(\widehat{\psi}_r(\theta))/R$ is the variance of the Monte Carlo error, with Var on the simulation noise in $\widehat{\psi}_r$ (see, e.g., [24, 57]).

Moreover, by the delta method, asymptotically as the input data size increases,

$$\mathrm{Var}(\psi(\widehat{\theta})) \approx \nabla_\theta \psi(\theta)' \frac{\Sigma}{n} \nabla_\theta \psi(\theta) \tag{17.3}$$

where $\frac{\Sigma}{n}$ is the estimation variance of $\widehat{\theta}$, typically scaling in a data size parameter n (e.g., a parameter linear in all the individual data sizes for different input models) and $\nabla_\theta \psi$ is the gradient of ψ with respect to θ, known as the sensitivity coefficient. If we use maximum likelihood for instance, then Σ would comprise the inverse Fisher information matrices.

Thus, when both the input data size and simulation replication size are large, we have

$$\widehat{\psi}(\widehat{\theta}) - \psi(\theta) \approx N\left(0, \frac{\nabla_\theta \psi(\theta)' \Sigma(\theta) \nabla_\theta \psi(\theta)}{n} + \frac{\mathrm{Var}(\widehat{\psi}_r(\theta))}{R}\right)$$

which then can be used to generate $[L, U]$ in Section 17.3. Note that $\nabla_\theta \psi(\theta)$ needs to be estimated by simulation (as ψ itself also needs to be estimated by such), via one of the following ways:

Unbiased gradient estimator: $\nabla_\theta \psi(\theta)$ estimated via unbiased methods such as the likelihood ratio or score function method [55, 103, 106]. This, however, may have high variance especially for long-horizon problems.

Finite-difference or zeroth-order gradient estimator: $\nabla_\theta \psi(\theta)$ estimated by using finite difference that requires only unbiased function evaluations [47, 140]. The rate of convergence, however, is subcanonically slow, and a precise complexity analysis for the use in (17.3) is not available in the literature.

Two-point methods: $\nabla_\theta \psi(\theta)$ estimated by using finite difference, but only using a couple of "perturbation directions" for the θ vector, judicially chosen (the "two points"). [25, 26] show some advantages in this approach.

Though the delta method is based on the normality approximation common in statistics, in the context of input uncertainty its implementation requires the estimation of a gradient or sensitivity coefficient that may not always be easy. Moreover, in finite-sample situations this method may undercover [8]. This partially motivates the alternatives that are discussed later in this section.

17.3.1.2 Bayesian Methods

The delta method and the resulting CI discussed above take the classical frequentist perspective. In the input uncertainty literature, Bayesian methods provide an alternate approach. Assume we have data D for a single input model parametrized by the unknown parameter vector θ, which is distributed according to $p(\xi|\theta)$. In the Bayesian framework, we impose a prior distribution $p_{prior}(\theta)$ on θ, and compute the posterior distribution

$$p_{post}(\theta|D) \propto p_{prior}(\theta)p(D|\theta) \tag{17.4}$$

where $p(D|\theta)$ is the likelihood of the data D, which is often a product of terms $p(\xi|\theta)$ (note that we could have multiple input models each with its own set of parameters, in which case we multiply these likelihoods together if the data are all independent). Computing the posterior $p_{post}(\theta|D)$ is a subject of long-standing interest, where in some cases (e.g., conjugate prior) it is readily computable, while in other cases more elaborate techniques such as Markov chain Monte Carlo (MCMC) are required [27]. Compared to the frequentist interpretation, a commonly viewed advantage of a Bayesian approach is the flexibility in specifying prior belief about uncertain parameters, which can be used to incorporate expert knowledge [63]. Moreover, Bayesian approaches are especially convenient in handling dynamic problems where data are sequentially assimilated, since the posterior distribution can be naturally used as an updating mechanism to reflect all the historical information.

In translating the above into inference on $\psi = \psi(\theta)$, note that, much like the frequentist case, we encounter two sources of uncertainty, one from the statistical error in estimating θ, and one from the simulation error. There are two views in handling this combination of uncertainties:

Direct Combination: This approach uses a distribution to capture the overall uncertainty that "lumps" the two sources together. More precisely, the sampling scheme repeatedly draws a sample from the posterior of θ, and given this sample that is used to calibrate the input model, a simulation run is conducted. The distribution of all these simulation outputs comprises a quantification of the combined input-simulation error. This approach is conceptually straightforward, and is used in, e.g., [27].

Variance Decomposition: The second approach resembles more closely the frequentist approach described in the last subsection. It consists of a two-layer sampling, where at each outer layer, a posterior sample of θ is drawn, and given this value that calibrates the input model, several (i.e., R) simulation runs are conducted which forms the inner sampling layer. Then the input

variance and the simulation variance can be estimated via methods much like the analysis-of-variance to be presented in the variance-based bootstrap in Section 17.3.2.1. This approach is used by [143, 144].

17.3.1.3 Metamodel-Based Methods

The direct bootstrap, whether used to construct variance-based or quantile-based CIs (to be discussed in Section 17.3.2.1), is corrupted by Monte Carlo error. When the Monte Carlo error is large relative to the input uncertainty, the result can be significant overcoverage of CIs, providing the experimenter with lower precision than what they should be.

For the parametric setting, constructing a metamodel for $\psi(\theta)$ can greatly reduce the impact of Monte Carlo error on the bootstrap distribution of $\widehat{\psi}(\theta_b)$, where the bootstrap resamples are indexed by b. This phenomenon is easiest to see by considering the case where ψ can be modeled with linear regression (to full fidelity), and then considering prediction in the CLT case. To further simplify the motivation for metamodeling, assume that ψ includes a transformation of the simulation output so that $\widehat{\psi}_r(\theta)$ has homogeneous variance with respect to θ.

That is, we have the full-fidelity metamodel:

$$\psi(\theta) = g(\theta)'\beta + \varepsilon, \varepsilon \sim N(0, \mathrm{Var}(\widehat{\psi}_r/R)) \tag{17.5}$$

where g is the vector-valued regression function. Denoting the fitted regression metamodel prediction by $\widehat{\psi}_{mm}$, the approximate variance decomposition of $\widehat{\psi}_{mm}(\widehat{\theta}) - \psi(\theta)$ can be compared with (17.2), where $\mathrm{Var}(\psi_r/R)$ in (17.2) is multiplied by

$$(g(\theta)'(\mathrm{G}'\mathrm{G})^{-1}g(\theta)), \tag{17.6}$$

and G is the design matrix of g values used to fit the metamodel. This assumes that the design matrix has one row per different design condition, which would result in R replications of each row in G. The multiplier will be smaller than one for θ values inside the design space. For simple linear regression with $\theta \in \mathbb{R}^d$, design region scaled to $(+/-1)^d$ and a 2^d factorial design, the largest value for $\theta'(\mathrm{G}'\mathrm{G})^{-1}\theta$ occurs at the corners of the hypercube, with value $(d+1)/2^d$. This provides one motivation for metamodeling: using metamodel-predicted bootstrap estimates for $\widehat{\psi}$ can have greatly reduced Monte Carlo error. The second motivation is that the bootstrap resamples of θ each requires an inexpensive evaluation of the metamodel, rather than

several replications of the expensive simulation model. If the metamodel fitting design has fewer runs than the bootstrap, the computational effort will be less.

Linear regression metamodels are essentially identical to the delta method reviewed in Chapter 17.3.1.1. Here we focus on the use of nonlinear metamodels with the bootstrap to characterize the distribution of $\psi(\widehat{\theta})$.

When the input data is limited, the potential for highly nonlinear response of the simulation as a function of θ makes low-order polynomial regression less attractive, since Taylor's Theorem is less likely to apply. Further, the assumption of homoscedastic variance (independence of $\mathrm{Var}(\widehat{\psi}_r/R)$ from θ) is harder to support.

In a series of papers [9, 137, 138], Xie, Nelson, and Barton employed stochastic kriging [1] to provide metamodel-based bootstrap CIs for input uncertainty. The work covers frequentist, Bayesian, and dependent input variable cases.

Key to the method was the experiment design strategy. Unlike linear regression, prediction error for stochastic kriging models increases when the prediction point is far from any experiment design point. In this setting, space-filling designs are preferred to traditional factorial experiment designs. Since the metamodel is used to evaluate bootstrap-resampled θ_b values, the design should focus on the bootstrap-resample space. The design proposed by the authors had two phases. In the first phase, a large number of bootstrap sample θ_b values were generated (without simulating the resulting performance), then a hyperellipse enclosing a high fraction (e.g., 99%) of the θ_b values was fitted. In the second phase, a space-filling design for a hypercube (e.g., Latin hypercube design) was transformed to cover the fitted ellipsoid.

The experiment design was executed (with replications) for specified θ values to fit a stochastic kriging metamodel. Then bootstrap-resampled θ_b values were used with the metamodel to generate approximate bootstrap $\psi(\theta_b)$ values to assess input uncertainty.

Metamodel-assisted bootstrapping has two advantages over direct bootstrapping. It makes efficient use of simulation budget by assigning simulation effort through a designed experiment that evenly covers the uncertain θ input space. The direct bootstrap over-emphasizes simulation effort in the design region of the most commonly occurring realizations of θ_b, which are unlikely to contribute much to confidence interval or quantile estimation. Second, the metamodeling approach produces the reduction in Monte Carlo uncertainty described above. This makes bootstrapping not only efficient but also robust to Monte Carlo error.

But metamodel-assisted bootstrapping may have disadvantages over the direct bootstrap when θ is high-dimensional. The number of simulation experiment runs required to fit a high-fidelity metamodel can increase rapidly with the dimension of θ. Thus, for a simulation model with many input parameters, the number of runs for the metamodel fitting experiment may exceed the number of direct bootstrap simulation model runs required for adequate input uncertainty characterization.

17.3.1.4 Green Simulation

Green simulation is an application of the likelihood ratio method, where simulation replications made to estimate $\psi(\theta)$ are reused to construct an estimator of $\psi(\theta')$ for $\theta' \neq \theta$ [44, 45].

Green simulation has been applied to reduce the computational cost of input uncertainty quantification. We focus on the Bayesian setting in this section, although the same approach can be applied in the frequentist setting as well. Suppose $\theta_1, \theta_2, \ldots, \theta_B$ are sampled from the posterior, $p_{post}(\theta|D)$. At each $\theta_b, 1 \leq b \leq B$, R simulation replications, $\widehat{\psi}_1(\theta_b), \widehat{\psi}_2(\theta_b), \ldots, \widehat{\psi}_R(\theta_b)$, are made and let $\widehat{\psi}(\theta_b)$ denote their sample mean. Moreover, let ζ_r^b be the vector of input random variables generated within the r-th replication given θ_b and $f(\cdot|\theta_b)$ denote the probability density function of ζ_r^b. Thus, $\widehat{\psi}_r(\theta_b)$ is a deterministic function of ζ_r^b. The following change of measure allows us to reuse the inputs and outputs generated from the R replications made at θ_b to estimate $\psi(\theta_{b'})$ for $\theta_{b'} \neq \theta_b$:

$$\psi(\theta_{b'}) = \mathrm{E}\left[\widehat{\psi}_r(\theta_b)\frac{f(\zeta_r^b|\theta_{b'})}{f(\zeta_r^b|\theta_b)}\right] = \int_\zeta \widehat{\psi}_r(\theta_b)\frac{f(\zeta|\theta_{b'})}{f(\zeta|\theta_b)} f(\zeta|\theta_b)d\zeta.$$

Here, an implicit assumption is that the support of the input vector does not depend on θ. Therefore, $\tilde{\psi}^b(\theta_{b'})$ below is an unbiased estimator of $\psi(\theta_{b'})$:

$$\tilde{\psi}^b(\theta_{b'}) = \frac{1}{R}\sum_{r=1}^{R} \widehat{\psi}_r(\theta_b)\frac{f(\zeta_r^b|\theta_{b'})}{f(\zeta_r^b|\theta_b)}.$$

Using this trick, [43] propose pooling all BR replications to estimate each $\psi(\theta_b)$ to improve computational efficiency. However, this approach should be taken with caution; although $\tilde{\psi}^b(\theta_{b'})$ is unbiased, its variance may be unbounded. The exact derivation of $\mathrm{Var}(\tilde{\psi}^b(\theta_{b'}))$ cannot be obtained in

general, but it can be estimated. Thus, one can choose not to pool replications made at some θ_b to estimate $\theta_{b'}$, if the estimated variance is large.

Expanding on the idea, [46] study an experiment design scheme to minimize the total number of replications so that the resulting pooled estimators have (approximately) the same variances as the original two-layer design that requires BR replications. The experiment design can be optimized prior to running any replications as long as $f(\cdot|\theta_b)$ is known. For a special case, they show that the minimized simulation budget is $O(B^{1+\varepsilon})$ for any $\varepsilon > 0$, which is a significant reduction from BR.

17.3.2 Nonparametric Methods

We now turn to methods that apply to nonparametric regimes in which the input distribution is not assumed to follow any parametric family. Following Section 17.2, we write $\psi = \psi(\mathbf{P})$, where $\mathbf{P} = (P_1, \ldots, P_m)$ is a collection of m input distributions. Suppose we have a collection of data sets $\mathbf{D} = (D_1, \ldots, D_m)$, where each $D_i = (\xi_{i1}, \ldots, \xi_{in_i})$ is the data set of size n_i distributed under P_i. Suppose the data are all independent. Then, naturally we construct empirical distributions $\widehat{\mathbf{P}} = (\widehat{P}_1, \ldots, \widehat{P}_m)$ from these data sets, where

$$\widehat{P}_i(\cdot) = \frac{1}{n_i} \sum_{j=1}^{n_i} \delta_{\xi_{ij}}(\cdot)$$

for Dirac measure $\delta_{\xi_{ij}}(\cdot)$. With these input distributions, we generate R simulation runs to obtain

$$\widehat{\psi}(\widehat{\mathbf{P}}) = \frac{1}{R} \sum_{r=1}^{R} \widehat{\psi}_r(\widehat{\mathbf{P}})$$

where $\widehat{\psi}_r$, $r = 1, \ldots, R$ are independent simulation runs.

Under regularity conditions, we have a CLT

$$\widehat{\psi}(\widehat{\mathbf{P}}) - \psi(\mathbf{P}) \approx N\left(0, \mathrm{Var}(\psi(\widehat{\mathbf{P}})) + \frac{\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))}{R}\right) \tag{17.7}$$

much like the parametric case in Section 17.3.1.1 [57].

Though conceptually similar, a main difference between the nonparametric and parametric setups is the representation of the input variance

$\text{Var}(\psi(\widehat{\mathbf{P}}))$. In particular, to specify this quantity, we would need to employ the nonparametric delta method, which involves the notion of functional derivatives. More specifically, the so-called influence function [61] $IF(\xi;\mathbf{P}) = (IF_1(\xi_1;\mathbf{P}),\ldots,IF_m(\xi_m;\mathbf{P}))$, which maps from the image of random variable $\xi = (\xi_1,\ldots,\xi_m)$ to $\mathbb{R}^m$, is a function satisfying the property

$$\psi(\mathbf{Q}) \approx \psi(\mathbf{P}) + \int IF(\xi;\mathbf{P})d(\mathbf{Q}-\mathbf{P}) + \text{remainder} \tag{17.8}$$

where $\mathbf{Q} = (Q_1,\ldots,Q_m)$ is a sequence of independent distributions Q_i each defined on the same space as P_i, $\int IF(\xi;\mathbf{P})d(\mathbf{Q}-\mathbf{P})$ is defined as

$$\int IF(\xi;\mathbf{P})d\mathbf{Q} = \sum_{i=1}^{m}\int IF_i(\xi_i;\mathbf{P})d(Q_i - P_i),$$

and the remainder in (17.8) goes to zero at a higher order than $\mathbf{Q}-\mathbf{P}$ (which can be rigorized). Note that we can replace $\int IF(\xi;\mathbf{P})d(\mathbf{Q}-\mathbf{P})$ by $\int IF(\xi;\mathbf{P})d\mathbf{Q}$, by redefining $IF(\xi;\mathbf{P})$ as $IF(\xi;\mathbf{P}) - \text{E}_{\mathbf{P}}[(\xi;\mathbf{P})]$. Thus, without loss of generality, we can use a canonical version of IF that satisfies (17.8) and also the mean-zero property (under $\mathbf{P}$). From (17.8), we see that the influence function dictates the linearization of $\psi(\mathbf{P})$ as $\mathbf{P}$ perturbs to $\mathbf{Q}$, and plays a distributional analog of the derivative in Euclidean space, which leads to the notion of Gateaux, Frechet or most relevantly Hadamard derivatives [128].

With the influence function IF, it turns out that the input variance $\text{Var}(\psi(\widehat{\mathbf{P}}))$ is given by

$$\text{Var}(\psi(\widehat{\mathbf{P}})) = \sum_{i=1}^{m}\frac{\text{Var}(IF(\xi_i;\mathbf{P}))}{n_i} \tag{17.9}$$

where the variance in the RHS is on the random variable ξ_i generated from P_i. This formula is a nonparametric analog to (17.3).

A major challenge in the nonparametric case that distinguishes from parametric is that the influence function generally requires more effort to estimate than the sensitivity coefficient for parametric input models. Efficient estimation of the variance of influence function is potentially doable [84], but quite open in the literature, and the input uncertainty literature has focused on resampling, cancellation methods or nonparametric Bayesian approaches, as

we describe below. We also note that many of these approaches, by design, also apply naturally for parametric settings.

17.3.2.1 Bootstrap: Elementary Schemes

The bootstrap method in input uncertainty can be roughly categorized into two frameworks, quantile-based and variance-based.

Quantile-based bootstrap: The bootstrap approach is principled on the observation that the variability of a statistic, under the data distribution, can be approximated by the counterpart of a resampled statistic, conditioned on the realized data [34, 40]. To be more precise, suppose we have a point estimate of $\psi(\mathbf{P})$ given by $\psi(\widehat{\mathbf{P}})$, and we want to construct a $(1-\alpha)$-level CI. This problem can be recast as the search of $[\underline{q}, \overline{q}]$ such that $P(\underline{q} \leq \psi(\widehat{\mathbf{P}}) - \psi(\mathbf{P}) \leq \overline{q}) = 1-\alpha$ which then gives $[\psi(\widehat{\mathbf{P}}) - \overline{q}, \psi(\widehat{\mathbf{P}}) - \underline{q}]$ as the CI. The bootstrap stipulates that

$$P_*(\underline{q} \leq \psi(\mathbf{P}^*) - \psi(\widehat{\mathbf{P}}) \leq \overline{q}) \approx P(\underline{q} \leq \psi(\widehat{\mathbf{P}}) - \psi(\mathbf{P}) \leq \overline{q}) \qquad (17.10)$$

where $\mathbf{P}^*$ is a resampled distribution, namely the empirical distribution formed by sampling with replacement from the data with the same size (or, in the case of m independent input distributions, the resampling is done independently from each input data set), and P_* denotes the probability conditional on the data. Thanks to (17.10), we can use Monte Carlo to approximate $\underline{q}$ and $\overline{q}$, say $\underline{q}^*$ and $\overline{q}^*$, which then gives $[\psi(\widehat{\mathbf{P}}) - \overline{q}^*, \psi(\widehat{\mathbf{P}}) - \underline{q}^*]$ as our CI.

The above principle has been used in constructing CIs for $\psi(\theta)$ under input uncertainty. A main distinction in this setting compared to conventional usage of the bootstrap is that the performance function ψ itself needs to be estimated from running many simulation runs. Thus, applying the bootstrap in the input uncertainty setting typically requires a *nested simulation*, where in the first layer, we resample the input distributions B times, and then given each resampled input distribution, we draw in the second layer a number, say R, of simulation runs driven by the resampled input distribution. The overall simulation complexity is BR.

The *basic bootstrap* method gives precisely the interval $[\widehat{\psi}(\widehat{\mathbf{P}}) - \overline{q}^*, \widehat{\psi}(\widehat{\mathbf{P}}) - \underline{q}^*]$, where $\widehat{\psi}(\widehat{\mathbf{P}})$ is a point estimate that uses the empirical distribution and enough simulation runs, and $\underline{q}^*$ and $\overline{q}^*$ are obtained as described above, using R large enough to approximate ψ sufficiently accurately in each resampled performance measure $\widehat{\psi}(\mathbf{P}^*)$.

On the other hand, (Efron's) *percentile bootstrap* uses the interval $[\underline{q}^*, \overline{q}^*]$ directly, where $\underline{q}^*$ and $\overline{q}^*$ are defined as in the basic bootstrap above. This approach does not require computing the point estimate, and is justified with the additional assumption that the limiting distribution of the statistic (centered at the true quantity of interest) is symmetric, which is typically true because this limiting distribution in many cases is a mean-zero normal distribution. Barton [10, 5] studies this approach.

Variance-based bootstrap: Instead of using quantiles $\underline{q}$ and $\overline{q}$ to construct CI, we can also estimate $\mathrm{Var}(\psi(\widehat{\mathbf{P}}))$ directly and then use the CLT (17.7) to deduce the interval

$$\left[\widehat{\psi}(\widehat{\mathbf{P}}) \pm z_{1-\alpha/2}\sqrt{\mathrm{Var}(\psi(\widehat{\mathbf{P}})) + \frac{\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))}{R}}\right] \tag{17.11}$$

where $z_{1-\alpha/2}$ is the $1-\alpha/2$-level normal critical value. Note that in (17.11) the simulation variance $\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))$ is typically easy to estimate by simply taking the sample variance of all simulation runs in the experiment, and the difficulty, as noted in the introduction, is the input variance.

To estimate $\mathrm{Var}(\psi(\widehat{\mathbf{P}}))$ using the bootstrap, we once again invoke the approximation principle of resampling distribution for the original distribution of a statistic, that

$$\mathrm{Var}_*(\psi(\mathbf{P}^*)) \approx \mathrm{Var}(\psi(\widehat{\mathbf{P}})) \tag{17.12}$$

where Var_* is the variance of the resampling randomness conditional on the data. Note that in the simulation setting, ψ has to be estimated from an enough number, say R, of simulation runs for each resampled input distribution $\mathbf{P}^*$. Thus, once again, this approach typically requires a nested simulation like in the quantile-based method.

Note that the accuracy of estimating $\mathrm{Var}_*(\psi(\mathbf{P}^*))$ can be improved by using analysis-of-variance (ANOVA) to debias the effect coming from finite simulation runs. To explain, note that a naive approach to estimate $\mathrm{Var}_*(\psi(\mathbf{P}^*))$, with B first-layer resampling and R second-layer simulation is

$$\frac{1}{B-1}\sum_{b=1}^{B}(\widehat{\psi}(\mathbf{P}^{*b}) - \overline{\psi(\mathbf{P}^*)})^2 \tag{17.13}$$

where $\mathbf{P}^{*b}$ denotes the b-th resample distribution, $\widehat{\psi}$ denotes the sample mean from R runs, and $\overline{\psi(\mathbf{P}^*)}$ denotes the sample mean of B resample

performance measures. Viewing the resample simulation experiment as a random-effect model, where each resample corresponds to a group and the simulation per resample corresponds to the sample within a group, we can readily see that the mean of (17.13) is actually

$$\mathrm{Var}_*(\psi(\mathbf{P}^*)) + \frac{\mathrm{E}_*[\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}^*)|\mathbf{P}^*)]}{R} \tag{17.14}$$

where $\widehat{\psi}_r$ refers to the r-th inner simulation run, and E_* refers to the expectation on the resampling randomness of $\mathbf{P}^*$ conditional on the data. Thus, since we are interested in estimating the between-group variance in (17.14), we can use

$$\frac{1}{B-1}\sum_{b=1}^{B}(\widehat{\psi}(\mathbf{P}^{*b}) - \overline{\psi(\mathbf{P}^*)})^2 - \frac{1}{BR(R-1)}\sum_{b=1}^{B}\sum_{r=1}^{R}(\widehat{\psi}_r(\mathbf{P}^{*b}) - \widehat{\psi}(\mathbf{P}^{*b}))^2 \tag{17.15}$$

to remove the within-group variance. The formula (17.15) is used in, e.g., [117].

Note that both quantile-based and variance-based frameworks can be applied to the case when the parametric bootstrap is adopted. In parametric bootstrapping, an input model and its parameter vector $\widehat{\theta}$ are first fitted from the data, then bootstrap samples are generated by sampling from the input model.

17.3.2.2 Bootstrap: Computational Enhancements

The bootstrap methods discussed above, though natural and straightforward to understand, unfortunately require high-computational load in general. This computational load arises from the need of running nested simulation (resampling at the outer layer, and simulation runs per resampled input model at the inner layer), which requires a multiplicative amount of simulation runs where, in order to control the overall error that is convoluted by both the input and simulation noises, the sampling effort in each layer has to be sufficiently large. To explain more concretely, consider the variance-based bootstrap where we use B outer samples and R inner samples. Under suitable assumptions and using [123], we obtain that the variance (conditional on the

data as we are using the bootstrap) of (17.15) is

$$O\left(\frac{1}{Bn^2}+\frac{1}{BR^2}\right) \tag{17.16}$$

where n is a scaling for the data size (i.e., the input data n_i for each input-model i is assumed to scale proportionally with n for some proportional constant). Now, note that the target quantity that (17.15) estimates is the input variance given by (17.9), which is of order $1/n$. Since this input variance shrinks to 0 as n increases, if we want to get a good input variance estimator, a basic requirement is relative consistency, which means the ratio of (17.16) and $1/n^2$ needs to go to 0. This in turn means, from the first term in (17.16), that B needs to go to ∞ and, from the second term, that R needs to be at least order n, which then gives a total required effort of strictly larger order than the data size n. [82] calls this a *simulation complexity barrier* for using naive variance-based bootstrap.

Some methods that improve the computational complexity motivated from the barrier above include subsampling, which has been used in the variance-based bootstrap framework, and shrinkage, which has been used in the quantile-based framework.

Subsampling: On a high level, the difficulty in using nested simulation to accurately estimate $\mathrm{Var}_*(\psi(\mathbf{P}^*))$, or subsequently $\mathrm{Var}(\psi(\widehat{\mathbf{P}}))$, is due to the small magnitude of these quantities that are in the order of $1/n$. This small magnitude requires one to wash away the noise coming from the inner sampling and necessitates the use of a large inner sample size. This issue manifests explicitly when we analyze the variance of the variance estimator in (17.16).

[82, 83] proposes to use subsampling to reduce sampling effort. Their main insight is the following. Suppose we had a smaller data size, say with a scale s, to begin with. Then, from our discussion above, this becomes an *easier* problem and in particular we would only need a larger order than s (instead of n) total simulation effort to ensure relative consistency. Of course, s is not the original scale of the sample size. However, we can utilize the reciprocal form of the input variance in terms of data size shown in (17.9) to rescale our estimate in the s-scale back to the n-scale.

To make the argument above more precise, denote the bootstrap input variance as

$$\mathrm{Var}_*(\psi(\mathbf{P}_n^*))$$

where we introduce a subscript n in $\mathbf{P}_n^*$ to explicitly denote the data size in the resample. From (17.9) and (17.12), we know

$$\text{Var}(\psi(\widehat{\mathbf{P}})) \approx \text{Var}_*(\psi(\mathbf{P}_n^*)) \approx \sum_{i=1}^{m} \frac{\text{Var}(IF(\xi_i; \mathbf{P}))}{n_i} \tag{17.17}$$

Now, if we use a resample size of scale s, or more precisely we use $s_i = \rho n_i$ for some factor $0 < \rho < 1$, then the bootstrap input variance becomes

$$\text{Var}_*(\psi(\mathbf{P}_s^*)) \approx \sum_{i=1}^{m} \frac{\text{Var}(IF(\xi_i; \mathbf{P}))}{s_i} \tag{17.18}$$

which now requires order larger than s effort instead of n effort due to (17.16). Now, comparing (17.17) and (17.18), we see that we have

$$\begin{aligned} \rho \text{Var}_*(\psi(\mathbf{P}_s^*)) &\approx \sum_{i=1}^{m} \frac{\rho \text{Var}(IF(\xi_i; \mathbf{P}))}{s_i} \\ &= \sum_{i=1}^{m} \frac{\text{Var}(IF(\xi_i; \mathbf{P}))}{n_i} \approx \text{Var}_*(\psi(\mathbf{P}_n^*)) \end{aligned} \tag{17.19}$$

So, by subsampling input distributions using a size scale s, running the nested simulation to estimate the bootstrap input variance, and then multiplying back by a factor of ρ gives rise to a valid estimate of the input variance, now with a total computational effort controlled by s instead of n. We could choose s to be substantially smaller than n, in principle independent of the data size.

Shrinkage: The principle of quantile-based bootstrap relies on the closeness between the distribution of a resampled estimate $\psi(\mathbf{P}^*)$ (conditional on data) and the original estimate $\psi(\widehat{\mathbf{P}})$, when suitably scaled and centered. In other words,

$$\psi(\mathbf{P}^*) - \psi(\widehat{\mathbf{P}}) \approx \psi(\widehat{\mathbf{P}}) - \psi(\mathbf{P})$$

where $\approx$ denotes approximation in distribution, conditional on data for the LHS, which then gives rise to (17.10) as a way to generate CI for $\psi(\mathbf{P})$. When $\psi(\cdot)$ needs to be computed via simulation, then the point estimate becomes $\widehat{\psi}(\widehat{\mathbf{P}})$, and we would use

$$\widehat{\psi}(\mathbf{P}^*) - \psi(\widehat{\mathbf{P}}) \approx \widehat{\psi}(\widehat{\mathbf{P}}) - \psi(\mathbf{P})$$

where each $\widehat{\psi}(\cdot)$ is estimated from a number of simulation runs. To use the basic bootstrap, we would use the quantiles of the LHS above to approximate the quantiles of the RHS. Unfortunately, this means we also need to estimate the "center" quantity $\psi(\widehat{\mathbf{P}})$ in the LHS via simulation. Moreover, we need to use enough simulation to wash away this noise so that

$$\widehat{\psi}(\mathbf{P}^*) - \widehat{\psi}(\widehat{\mathbf{P}}) \approx \widehat{\psi}(\widehat{\mathbf{P}}) - \psi(\mathbf{P}) \tag{17.20}$$

In other words, we need a large simulation size, say R_0, to get a point estimate $\widehat{\psi}(\widehat{\mathbf{P}})$ that has negligible simulation error. And if we do so, then the bootstrap-resample estimate $\widehat{\psi}(\mathbf{P}^*)$ in (17.20) would each require a matching simulation size R_0, and at the end the computation load is $R_0 B$ where B is the bootstrap size, which could be very demanding.

The shrinkage method proposed by [8] is an approach to reduce the simulation size in each bootstrap-resample estimate, while retaining the approximation (17.20). The approach is inspired from a similar concept to adjust for variances in statistical linear models [34]. Suppose each bootstrap-resample estimate $\widehat{\psi}(\mathbf{P}^*)$ uses $R < R_0$ runs (while the point estimate $\widehat{\psi}(\widehat{\mathbf{P}})$ uses R_0 runs), then the quantity

$$\widehat{\psi}(\mathbf{P}^*) - \widehat{\psi}(\widehat{\mathbf{P}}) \tag{17.21}$$

has a larger variance than

$$\widehat{\psi}(\widehat{\mathbf{P}}) - \psi(\mathbf{P}).$$

To compensate for this, we scale down the variability of the outcomes of (17.21) by a *shrinkage factor*

$$S = \sqrt{\frac{\mathrm{Var}(\psi(\widehat{\mathbf{P}}))}{\mathrm{Var}(\psi(\widehat{\mathbf{P}})) + \frac{\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))}{R}}}$$

which comes from the ratio between the standard deviation of (17.21) when $\widehat{\psi}(\mathbf{P}^*)$ is estimated using R_0 simulation runs (which is assumed so large that the simulation noise becomes negligible) and R simulation runs. To execute this shrinkage, we can either scale the resample estimate, i.e., multiply each $\widehat{\psi}(\mathbf{P}^*)$ by S before applying the basic bootstrap, or scale the quantile obtained from the basic bootstrap directly. The shrinkage factor itself is

estimated using ANOVA described in Section 17.3.2.1. Moreover, a similar shrinkage approach can be applied to the percentile bootstrap.

17.3.2.3 Batching, Sectioning, and Sectioned Jackknife

Recall the CLT in (17.7) that, when combined with (17.9), gives

$$\widehat{\psi}(\widehat{\mathbf{P}}) - \psi(\mathbf{P}) \approx N\left(0, \sum_{i=1}^{m} \frac{\mathrm{Var}(IF(\xi_i; \mathbf{P}))}{n_i} + \frac{\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))}{R}\right) \tag{17.22}$$

The batching method studied by [57] utilizes a pivotal t-statistic constructed from asymptotic normal variables in (17.22) to efficiently generate a CI. Divide the input data for each input-model i into say K batches, each of size m_i (so that $Km_i = n_i$, ignoring integrality). For each batch (which includes the data corresponding to all input models), we construct the empirical distribution $\mathbf{F}^k$ and run R simulation runs to obtain the k-th batched estimate $\widehat{\psi}(\mathbf{F}^k)$. When K is a fixed, small number (e.g., $K = 5$), then as $n_i \to \infty$ we have the CLT

$$\begin{aligned}&\left(\widehat{\psi}(\widehat{\mathbf{P}}^k) - \psi(\mathbf{P})\right)_{k=1,\ldots,K} \\ &\quad \approx \left(N\left(0, \sum_{i=1}^{m} \frac{\mathrm{Var}(IF(\xi_i; \mathbf{P}))}{m_i} + \frac{\mathrm{Var}(\widehat{\psi}_r(\mathbf{P}))}{R}\right)\right)_{k=1,\ldots,K}\end{aligned} \tag{17.23}$$

where the normal variables are all independent. Thus, we can form a t-statistic

$$\frac{\bar{\psi} - \psi(\mathbf{P})}{S/\sqrt{K}}$$

where

$$\bar{\psi} := \frac{1}{K}\sum_{k=1}^{K} \widehat{\psi}(\widehat{\mathbf{P}}^k), \quad S^2 = \frac{1}{K-1}\sum_{k=1}^{K} (\widehat{\psi}(\widehat{\mathbf{P}}^k) - \bar{\psi})^2$$

which is distributed as t_{K-1}, the t-distribution with degree of freedom $K-1$. This gives a CI

$$\left[\bar{\psi} - t_{K-1,1-\alpha/2}\frac{S}{\sqrt{K}}, \bar{\psi} + t_{K-1,1-\alpha/2}\frac{S}{\sqrt{K}}\right]$$

where $t_{K-1,1-\alpha/2}$ is the $(1-\alpha/2)$-quantile of t_{K-1}. This idea resembles the batch means method commonly used in steady-state analysis [56, 109, 110], but here as a means to generate a CI capturing input uncertainty. The main strength of this approach is its light computation effort. To use batching with K number of batches, we need a simulation effort KR, and K in principle can be as small as 2. The caution here is that a small K would give a long interval (note the critical value $t_{K-1,1-\alpha/2}$ is large when K is small). Nonetheless, as K increases from 2, the decrease in interval width is steep and then stabilizes quickly [109]. In general, K equal to 5 would already reasonably approach the limiting critical value, i.e., the normal critical value $z_{1-\alpha/2}$.

If we have a point estimate $\widehat{\psi}(\widehat{\mathbf{P}})$ constructed from using all the input data, then we can also use the interval

$$\left[\widehat{\psi}(\widehat{\mathbf{P}}) - t_{K-1,1-\alpha/2}\frac{S}{\sqrt{K}}, \widehat{\psi}(\widehat{\mathbf{P}}) + t_{K-1,1-\alpha/2}\frac{S}{\sqrt{K}}\right].$$

where now the simulation effort for each sectioned estimate needs to be $1/K$ of the effort used for the point estimate $\widehat{\psi}(\widehat{\mathbf{P}})$ to elicit a proper self-normalization. This corresponds to the sectioning method.

The above can also be generalized to the jackknife, resulting in a sectioned jackknife method [2] for constructing CI under input uncertainty. The roadmap for deriving such a CI is similar in that a pivotal statistic is proposed, the difference being that due to the leave-one-section-out estimates in jackknife the cancellation needed in the pivotal statistic becomes more delicate to analyze. The benefit of sectioned jackknife, however, is that its resulting point estimate has a lower-order bias [2, 88], and that it is less sensitive, or more robust, against the adverse effect of a small batch size, because it uses all data *except* the batch.

17.3.2.4 Mixture-Based and Nonparametric Bayesian

When the sample size of the input data is relatively small, selecting a single parametric input model may be difficult. Instead of taking a purely nonparametric approach, Zouaoui and Wilson [144] propose to apply the Bayesian model averaging (BMA) scheme to construct a mixture of candidate input distributions and account for parametric as well as model uncertainties in their input uncertainty quantification framework.

Recall the Bayesian framework for modeling uncertainty about θ discussed in Section 17.3.1.2. The posterior update in (17.4) implicitly assumes that the distribution family is known. In BMA, in addition to imposing a prior

distribution for each candidate distribution's parameter vector, a prior is assumed for the weights that determine the mixture. Given the real-world data, both priors are updated to their posteriors. Using both posteriors, Zouaoui and Wilson [144] propose variance decomposition method that accounts for Monte Carlo error and estimation error in θ as well as uncertainty about the parametric distribution family, which they refer to as model uncertainty.

Although BMA provides flexibility in choosing the parametric family, one must come up with a set of candidate distributions. Bayesian bootstrap [105], on the other hand, is a Bayesian analog to the frequentist's bootstrap method. For the (nonparametric) bootstrap scheme discussed in Section 17.3.2.1, recall that $\mathbf{P}^*$ is an empirical distribution of resampled observations from the original data set, say, $\mathbf{D} = \{\xi_1, \xi_2, \ldots, \xi_n\}$, with replacement. Therefore, $\mathbf{P}^*$ can be written as a n-dimensional probability simplex assigning a probability mass to each ξ_j in $\mathbf{D}$. For $\mathbf{P}^*$, each ξ_j is assigned with a multiple of $1/n$, e.g., $0, 1/n, 2/n, \cdots$. In Bayesian bootstrap, the probability simplex is modeled as a realization of a Dirichlet distribution whose density function is proportional to

$$\prod_{j=1}^{n} p_j^{\delta_j - 1} \mathbf{1}\left\{\sum_j p_j = 1\right\},$$

where p_j is the probability mass assigned to ξ_j and $\{\delta_1, \delta_2, \ldots, \delta_n > 0\}$ are the concentration parameters. Therefore, the Bayesian bootstrap allows more flexibility in modeling $\mathbf{P}^*$ than the frequentist's bootstrap. The Dirichlet distribution is a conjugate prior for the multinomial distribution with probabilities $\{p_j\}$; the resulting posterior distribution of $\{p_j\}$ is still Dirichlet.

In the input uncertainty context, [10] show their uniform resampling method to be a kind of Bayesian bootstrap, [116] and [130] include nonparametric input models in their stochastic kriging metamodels by sampling the probability simplex from the Dirichlet posterior given data, and [136] study the use of Dirichlet process mixture to construct credible intervals for simulation outputs.

17.3.2.5 Robust Simulation

In recent years, an approach based on (distributionally) robust optimization has been studied to quantify input uncertainty. Robust optimization [15, 16] is a framework that originated from optimization under uncertainty, in which

the decision-maker uses a worst-case perspective, i.e., makes a decision that optimizes the worst-case performance over the unknown or uncertain parameter in the optimization problem. This uncertain parameter is postulated to lie in a so-called *uncertainty set* or *ambiguity set* that reflects the belief of the modeler which, roughly speaking, is a set where the truth is likely to lie in. The approach thus typically results in a minimax optimization problem, where the outer optimization is over the decision and the inner optimization computes the worst-case scenario constrained within the uncertainty set.

Distributionally robust optimization (DRO) [35, 60, 133] can be viewed as a branch of robust optimization when the uncertain parameter is the underlying probability distribution in a stochastic optimization. The uncertainty set can take a variety of forms, but mainly falls into two categories. The first consists of neighborhood balls surrounding a baseline distribution, where the ball size is measured by statistical distance such as φ-divergence [12, 14, 68], which includes for instance Kullback-Leibler (KL) and χ^2-distance, and Wasserstein distance [19, 23, 41, 48]. The second class consists of distributional summary constraints including moments and support [50, 62], marginal information [22, 36, 37], and distribution shape such as unimodality [79, 89, 102, 127].

The DRO approach, when applied to input uncertainty, can be viewed as a nonparametric approach, since the uncertainty sets can be created nonparametrically. The goal of this approach, much like the methods described above, is to construct intervals that cover the truth. This can be attained by imposing a worst-case optimization problem over the uncertainty set. Here, we use the term DRO broadly to refer to worst-case optimization, not necessarily having a decision to determine but only standard output analysis. More concretely,

$$\max / \min \psi(\mathbf{Q}) \quad \text{subject to} \quad \mathbf{Q} \in \mathcal{U} \tag{17.24}$$

where $\max / \min$ refers to a pair of maximization and minimization, and $\mathcal{U}$ is the uncertainty set in the probability space, and the decision variable is the unknown input distribution $\mathbf{Q}$. When the uncertainty set $\mathcal{U}$ is a confidence region on the unknown input distribution, then the worst-case optimization pair above would output an interval covering the true target quantity with at least the same confidence level. This implication can be readily seen and forms the basis of data-driven DRO [14, 17].

Regarding the construction of CIs, a main benefit of DRO is the flexibility to capture certain types of uncertainties beyond traditional statistical methods. For instance, in some problems, the modeler might be concerned about the misspecification of, say, i.i.d. assumptions, but is confident about the marginal distribution specification of the input process. In this case, the

modeler can constrain the marginal information in the uncertainty set, but leave the dependence structure open to some extent. Then the resulting values of the optimization pair (17.24) would give the worst-case interval subject to this level of uncertainty or information. As another example, in other problems with little knowledge or very few data on the input distribution, one cannot fit a distribution and needs to rely on expert knowledge or crude a priori information. In this situation, the modeler can impose an uncertainty set on the first-order moments only.

In general, the DRO problems (17.24) need to be solved via simulation optimization, since the objective function is the target performance measure that can be only be approximated via the simulation model. (17.24) is thus a constrained stochastic optimization where the constraints are built from the uncertainty set, and moreover the decision variable is probability distribution, i.e., constrained by probability simplex constraints. When $\psi(\cdot)$ is a linear function in $\mathbf{Q}$, i.e., an expectation, the problem can be solved by sample average approximation [53, 54, 66, 67]. In more general cases such as discrete-event systems where $\psi(\cdot)$ is nonlinear in the input distributions, [51, 52, 85] devise approaches using stochastic approximation to iteratively solve these problems, which involve stochastic Frank-Wolfe or mirror descent that specializes to a variety of uncertainty sets.

Another perspective that has been taken to utilize (17.24) is to conduct local sensitivity analysis. In this context, the modeler imposes an uncertainty set whose size signifies the deviation of some model parameters away from a baseline value, with auxiliary constraints in the uncertainty set that capture the model structure or quantity that is kept unchanged. When the size shrinks, the values of the worst-case optimization (17.24) are expressible as a Taylor-type expansion around the baseline value in terms of the ball size, with the coefficients representing the worst-case sensitivity of the performance measure due to these input model changes subject to the auxiliary constraints. [76] develops these expansions when the balls are measured in KL, and [77] further develops expansions under auxiliary constraints on p-lag serial dependency and when the ball size is measured by Pearson's φ^2-coefficient.

17.3.3 Empirical Likelihood

Relating to the last subsection, when $\mathcal{U}$ is set as the neighborhood ball measured by a statistical distance surrounding the empirical distribution, the optimization (17.24) has a close connection to the empirical likelihood (EL) method [96]. The latter is a nonparametric analog to the celebrated

maximum likelihood estimator (MLE) in parametric inference, and operates with the nonparametric MLE that turns out to equate to the (empirical, reverse) KL distance. EL uses a so-called profile likelihood that is the optimal value of an optimization problem with objective being this reverse KL distance and constraint being the quantity to be estimated, and conducts inference based on an analog to the classical Wilks' theorem.

It can be readily seen that when $\mathcal{U}$ uses the reverse KL distance, then (17.24) becomes the dual of the EL in the sense that the roles of objective and constraint in the profile likelihood are now reversed. This implies that the optimal value of (17.24) gives rise to CI that matches those generated from EL, when we choose the ball size of $\mathcal{U}$ to be a suitable χ^2-quantile [39, 87]. In other words, (17.24) provides an alternate approach to construct asymptotically exact CIs that is different from the delta method and the bootstrap. Notably, this interpretation goes beyond the rationale of data-driven DRO presented in Section 17.3.2.5 [78]. Lastly, it is notable that the statistical distance in $\mathcal{U}$ does not have to be reverse KL, but can also be any of a wide class of φ-divergence, and more recently Wasserstein distance [18].

Lam and Qian [80, 81] use the EL, in the form of the DRO (17.24), to construct CI under input uncertainty. More precisely, [81] use a tractable approximation of (17.24), via linearization, to obtain solution of the worst-case distributions encoded in terms of probability weights on the data points. They then run simulation using input distributions that are weighted by these worst-case probability weights to obtain upper and lower bounds. Compared to the delta method, this approach demonstrates better finite-sample performance because its bound construction does not rely on linearization directly, which can give poor coverage especially for performance measures that are close to some "natural boundaries" (e.g., small probability estimation). Compared to the bootstrap, it does not involve nested simulation whose configuration can be difficult to optimize, but instead replace the resampling with solving a pair of optimization problems derived from (17.24).

17.4 Other Aspects

In addition to the construction of CIs and variance estimation, there are a few other aspects regarding the handling of input uncertainty.

17.4.1 Bias Estimation

Input uncertainty affects not only the variance of $\widehat{\psi}$, but also its bias relative to ψ as well. With small input data samples and highly nonlinear simulation response, this bias can be substantial. Morgan [91] employs a quadratic linear regression metamodel to compute a bias estimate:

$$\hat{b} = \frac{1}{2}\text{tr}(\widehat{\Omega}\widehat{\text{H}}(\hat{\theta})), \tag{17.25}$$

where $\widehat{\Omega}$ is the estimated covariance matrix of the MLE $\hat{\theta}$ and $\widehat{\text{H}}$ approximates the Hessian of $\psi(\theta)$, computed via a quadratic regression metamodel. In addition to providing a bias estimate, the authors construct a hypothesis test to identify statistically significant bias.

17.4.2 Online Data

Zhou and Liu [141] first study an online input uncertainty quantification problem, where additional real-world observations are sequentially made available and the posterior distribution on θ, $p_{post}(\theta|D)$, is updated at each stage. They apply green simulation techniques (see Section 17.3.1.4) to reuse replications made at the parameters sampled from a previous stage in the current stage.

17.4.3 Data Collection vs Simulation Expense

In some applications, additional data collection is feasible at a cost. A natural question in this setting is how to allocate the finite resource for data collection among a number of data sources so that input uncertainty can be minimized. Ng and Chick [93] study this problem for a parametric Bayesian setting where all input distributions are independent. They also consider a joint resource allocation problem among simulation and data collection when the cost of a simulation replication is relatively expensive. Xu et al. [139] expand this framework to the case with correlated inputs.

Relatedly, Song and Nelson [117] focus on decomposing the total input uncertainty into each input distribution's contribution when there are m independent input distributions; note a nonparametric version of such decomposition is shown in (17.9). They also propose a sample-size sensitivity measure that indicates how much input uncertainty can be reduced by collecting an extra observation from each data source.

17.4.4 Model Calibration and Inverse Problems

While the input uncertainty literature has focused on situations where the input models are observable from direct data, in some applications the *output*-level instead of input-level data are observed. In these cases, calibrating the input model becomes an inverse problem where the unknown parameters or input distributions can only be observed through the input-output relation described by the simulation model, which is usually analytically intractable but evaluatable only via noisy simulation runs. This problem resembles the UQ literature in computer experiments [64, 70, 125, 134], but now with additional stochastic noise in the simulator (recall our introduction). In discrete-event simulation, the problem of informing input parameter values through output data falls traditionally under the umbrella of model validation and calibration [4, 73, 108, 112], in which a modeler would compare the simulation model output with real output data using statistical tests or Turing tests, re-calibrate or enhance the model if the test concludes a mismatch, and iterate the process.

Recently, some approaches have been suggested to directly calibrate the input unknowns by setting up (simulation) optimization problems. In particular, [59] use entropy maximization, and [3, 58] use DRO with an uncertainty set constructed from a statistical distance between simulated and real output data. Furthermore, [86, 101] study the correction of output discrepancies between the simulated and real data at the distribution level.

17.5 Simulation Optimization under Input Uncertainty

Thus far, the focus was on quantifying input uncertainty of a simulation model. In this section, we discuss how one can formulate and solve a simulation optimization problem in the presence of input uncertainty. We first define the generic simulation optimization problem with the following form:

$$x^* = \arg\max_{x \in \mathcal{X}} \psi(x, \mathbf{P}), \tag{17.26}$$

where $\mathcal{X}$ is a feasible solution set, and $\mathbf{P}$ and ψ are as before. The parametric form is:

$$x^* = \arg\max_{x \in \mathcal{X}} \psi(x, \theta). \tag{17.27}$$

The performance measure is parameterized by both solution x and input parameter vector θ. What makes (17.26) and (17.27) "simulation" optimization problems is that $\psi(x, \theta)$ must be estimated by running simulations. When $\psi(x, \theta)$ is replaced with its estimate, $\widehat{\psi}(x, \theta)$, Monte Carlo error is introduced. Therefore, as long as the simulation budget is finite, we cannot find x^* with certainty in general. Instead, a simulation optimization algorithm aims to provide an estimate $\hat{x}^*$ of x^* with some statistical guarantee on closeness of $\hat{x}^*$ to x^*. One such guarantee may be

$$\mathbb{P}\{|\psi(\hat{x}^*, \theta) - \psi(x^*, \theta)| \leq \varepsilon\} \geq 1 - \alpha$$

for some $\varepsilon > 0$ and $0 < \alpha < 1$, where the probability is taken with respect to the Monte Carlo error in simulation. This implies that the probability that the optimality gap between $\hat{x}^*$ and x^* is within a tolerable level ($< \varepsilon$) is at least $1 - \alpha$.

In the traditional simulation optimization setting, $\mathbf{P}$ or θ is assumed to be given. Thus, the only source of stochasticity is Monte Carlo error in estimating $\widehat{\psi}(x, \theta)$, which can be reduced by running more simulation replications. The problem becomes more complex once input uncertainty is considered in conjunction with Monte Carlo error.

One may be tempted to solve the "plug-in" version by replacing the input distributions with their "best estimates" given the data. For instance, in the parametric case, θ may be replaced with its point estimate $\widehat{\theta}$ in (17.27) to find

$$x^*(\widehat{\theta}) = \arg\max_{x \in \mathcal{X}} \psi(x, \widehat{\theta}),$$

which is the conditional optimum given $\widehat{\theta}$. In general, $x^*(\widehat{\theta}) \neq x^*$ as $x^*(\widehat{\theta})$ depends on the random vector, $\widehat{\theta}$. However, when a generic simulation optimization algorithm is applied to the plug-in problem, it provides a statistical guarantee for finding $x^*(\widehat{\theta})$, not x^*. Therefore, to properly account for the effect of input uncertainty in simulation optimization, one must explicitly consider dependence of $\psi(x, \theta)$ on θ when designing the simulation optimization algorithm to provide a statistical guarantee for finding x^*.

Once again, consider the delta method:

$$\psi(x, \widehat{\theta}) \approx \psi(x, \theta) + \nabla_\theta \psi(x, \theta)'(\widehat{\theta} - \theta).$$

If this linear model is exact for all x and the gradient, $\nabla_\theta \psi(x, \theta)$, does not depend on x, then $x^*(\widehat{\theta}) = x^*$. However, this is an unrealistic assumption

for many practical problems as $\psi(x, \widehat{\theta})$ tends to have an interaction effect between x and $\widehat{\theta}$.

Suppose we compare performance measures at x and given $x^*\widehat{\theta}$. We have from the delta method,

$$\psi(x^*, \widehat{\theta}) - \psi(x, \widehat{\theta}) - \{\psi(x^*, \theta) - \psi(x, \theta)\} \approx \left[\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)\right]'(\widehat{\theta} - \theta). \quad (17.28)$$

The distribution of (17.28) lets us infer the value of $\psi(x^*, \theta) - \psi(x, \theta)$ from that of $\psi(x^*, \widehat{\theta}) - \psi(x, \widehat{\theta})$. Here, $\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)$ quantifies how differently the performance measures at x^* and x are affected by a small change in the parameter vector. [118] refer to the right-hand side of (17.28) as the common-input-data (CID) effect since it captures the impact of input uncertainty caused by the same set of real-world data in comparing x^* and x. Notice that the CID effect is random due to the uncertainty in $\widehat{\theta}$. If $\widehat{\theta}$ is a maximum likelihood estimator, then from the asymptotic distribution of $\widehat{\theta}$, (17.28) is approximately distributed as

$$N\left(0, \left[\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)\right]' \frac{\Sigma(\theta)}{n} \left[\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)\right]\right). \quad (17.29)$$

Observe that uncertainty about $\widehat{\theta}$ measured by $\Sigma(\theta)/n$ is amplified or reduced by the gradient difference. For instance, if the gradient difference is near 0 along a dimension of $\widehat{\theta}$, then even if its marginal variance is large, it may have very little impact on the variance of the CID effect. On the other hand, if the gradient difference is large, then the variance of (17.29) becomes large, which makes it difficult to infer that $\psi(x^*, \theta) - \psi(x, \theta) > 0$.

In fact, we do not observe $\psi(x^*, \widehat{\theta}) - \psi(x, \widehat{\theta})$, either. Suppose $\psi(x, \widehat{\theta})$ is estimated by a sample average of noisy simulation outputs, i.e., $\widehat{\psi}(x, \widehat{\theta}) = \frac{1}{R(x)} \sum_{r=1}^{R(x)} \psi_r(x, \widehat{\theta})$, where $R(x)$ is the number of replications run at x. Assuming that all simulations are run independently, the variance of $\widehat{\psi}(x^*, \widehat{\theta}) - \widehat{\psi}(x, \widehat{\theta}) - \{\psi(x^*, \theta) - \psi(x, \theta)\}$ is approximately

$$\left[\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)\right]' \frac{\Sigma(\theta)}{n} \left[\nabla_\theta \psi(x^*, \theta) - \nabla_\theta \psi(x, \theta)\right] + \frac{\mathrm{Var}(\psi_r(x^*, \widehat{\theta}))}{R(x^*)} + \frac{\mathrm{Var}(\psi_r(x, \widehat{\theta}))}{R(x)}. \quad (17.30)$$

Clearly, the latter two terms can be reduced by increasing $R(x^*)$ and $R(x)$, whereas the first term may be reduced only by collecting more real-world data.

In the following subsections, we first discuss the case when the data size, n, is fixed (17.5.1 and 17.5.2) and when streaming data are available (17.5.3).

17.5.1 Selection of the Best under Input Uncertainty

In this section, we consider the case when the set of feasible solutions, $\mathcal{X}$, contains a finite number of solutions. In particular, when $|\mathcal{X}|$ is relatively small, say k, we are able to simulate them all and compare their estimated performance measures to select the best (ranking and selection) or return a set of solutions that contains the best (subset selection). We refer the readers to Kim and Nelson [72] for foundations of ranking and selection, subset selection, and related work. Our main focus in this section is integration of input uncertainty into these methodologies.

17.5.1.1 Ranking and Selection

A classical problem formulation for ranking and selection (R&S) is

$$k^* = \arg\max_{1 \le k \le K} \psi(k, \theta) \tag{17.31}$$

where θ is assumed known. Notice that we replaced x with its index k given that the total number of alternatives in comparison is K. Typically, $\psi(k, \theta)$ is assumed to be the expectation of a stochastic simulation output, $\psi_r(k, \theta)$. A R&S procedure controls the numbers of replications assigned to each alternative in comparison given the total budget, N, so that it achieves the statistical guarantee that it is designed to provide upon termination. Depending on how the allocation is made, R&S procedures can be categorized into single-stage [13], two-stage [104], or sequential [71] procedures. Upon termination, an estimate of k^*, $\hat{k}^*$, is returned. Typically, $\hat{k}^*$ is the alternative that has the best sample mean given the simulation replications made throughout the procedure, namely,

$$\hat{k}^* = \arg\max_{1 \le k \le K} \widehat{\psi}(k, \theta),$$

where $\widehat{\psi}(k, \theta) = \frac{1}{R(k)} \sum_{r=1}^{R(k)} \psi_r(k, \theta)$ and $R(k)$ is the number of replications allocated to the k-th alternative.

To provide a statistical guarantee for selecting k^*, R&S procedures typically control the probability of correct selection (PCS)

$$\text{PCS} = \mathbb{P}\{\widehat{k}^* = k^*\}$$

to be at least $1/K < 1-\alpha < 1$. Equivalently, one may control the probability of false selection (PFS)

$$\text{PFS} = \mathbb{P}\{\widehat{k}^* \neq k^*\}$$

to be lower than $0 < \alpha < 1 - 1/K$. There are two main approaches to provide the PCS guarantee: one is to control the exact PCS given finite simulation budget N and the other is to control the asymptotic convergence rate of PFS assuming N is sufficiently large.

For the former, most procedures assume the simulation outputs are normally distributed to get a handle on the distribution of $\widehat{\psi}(k, \theta)$ and its variance estimator is given finite $R(k)$. Moreover, the procedures adopt the indifference zone (IZ) assumption, which states that all suboptimal alternatives have performance measures that are at least δ less than the best for some known $\delta > 0$. Mathematically, this can be written as

$$\psi(k^*, \theta) - \psi(k, \theta) \geq \delta, \forall k \neq k^*. \tag{17.32}$$

First introduced by Bechhofer [13], the IZ assumption turns out to be crucial for providing the finite-sample PFS guarantee; without the assumption, any suboptimal alternative's performance measure can be arbitrarily close to the best solution so that they may not be distinguished given finite N. The PCS under the IZ assumption is denoted as

$$\text{PCS}_\delta = \mathbb{P}\{\widehat{k}^* = k^* | \psi(k^*, \theta) - \psi(k, \theta) \geq \delta, \forall k \neq k^*\}$$

to differentiate it from the PFS. Under the normality assumption, several procedures have been designed to guarantee $\text{PCS}_\delta \geq 1 - \alpha$ after spending a finite amount of simulation budget, N, for any chosen $\delta > 0$.

When input uncertainty is considered, Problem (17.31) must be reformulated as θ is unknown. As discussed for the generic simulation optimization problem, one can construct the "plug-in" version of (17.31) by replacing θ with its point estimate $\widehat{\theta}$. The corresponding optimum of the plug-in problem, $k^*(\widehat{\theta})$, is then conditional on $\widehat{\theta}$. Song et al. [119] propose to

consider the following average PCS

$$\overline{\text{PCS}}_\delta = \text{E}_{\widehat{\theta}}[\mathbb{P}\{\widehat{k}^*(\widehat{\theta}) = k^* | \widehat{\theta}\} | \psi(k^*, \theta) - \psi(k, \theta) \geq \delta, \forall k \neq k^*],$$

where the outer expectation is with respect to the distribution of $\widehat{\theta}$. In words, $\overline{\text{PCS}}_\delta$ evaluates the probability that $k^*(\widehat{\theta})$ is indeed k^*. Of course, we have a single realization of $\widehat{\theta}$ computed from the set of real-world observations. Nevertheless, if $\overline{\text{PCS}}_\delta \geq 1 - \alpha$ can be guaranteed, then in expectation (over both Monte Carlo error and input uncertainty), the PCS guarantee is achieved.

From the definition of $\widehat{k}^*$, $\overline{\text{PCS}}_\delta$ can be rewritten as

$$\overline{\text{PCS}}_\delta = \text{E}_{\widehat{\theta}}[\mathbb{P}\{\widehat{\psi}(k^*, \widehat{\theta}) > \widehat{\psi}(k, \widehat{\theta}), \forall k \neq k^* | \widehat{\theta}\} | \psi(k^*, \theta) - \psi(k, \theta) \geq \delta, \forall k \neq k^*].$$

Thus, computing $\overline{\text{PCS}}_\delta$ requires characterizing the joint distribution of

$$\left\{\widehat{\psi}(k^*, \widehat{\theta}) - \widehat{\psi}(k, \widehat{\theta}) - [\psi(k^*, \theta) - \psi(k, \theta)]\right\}_{\forall k \neq k^*}. \tag{17.33}$$

Applying the delta method as in the beginning of Chapter 17.5, the joint distribution of (17.33) can be approximated with a multivariate normal distribution whose mean is the $(K - 1)$-dimensional zero vector and the elements of its variance-covariance matrix can be computed similarly as in (17.30). Under some additional assumptions on the variance-covariance matrix, Song et al. [119] derive the expression for $\overline{\text{PCS}}_\delta$ for a single-stage R&S procedure.

However, unlike any $\delta > 0$ is allowed for the generic R&S problem, the values of α and δ may not be chosen as desired when there is input uncertainty. To see this, consider the minimum indifference zone parameter, $\delta^\alpha_{\text{min}}$, given α defined as

$$\delta^\alpha_{\text{min}} = \inf\left\{\delta : \lim_{N \to \infty} \overline{\text{PCS}}_\delta \geq 1 - \alpha\right\}.$$

Loosely speaking, $\delta^\alpha_{\text{min}}$ is the smallest performance measure difference one can detect with desired precision $(1 - \alpha)$ in the presence of input uncertainty captured by estimation error of $\widehat{\theta}$. Note that $\delta^\alpha_{\text{min}}$ is an increasing function of α and may be strictly positive when the input data sample size, n, and α are small. For any δ smaller than $\delta^\alpha_{\text{min}}$, we cannot guarantee $\text{PCS}_\delta \geq 1 - \alpha$ even with an infinite simulation budget. Such a positive lower bound on the

IZ parameter is a result of input uncertainty. Derivation of $\delta^{\alpha}_{\min}$ depends on the specific R&S procedure; Song et al. [119] derive the expression for a single-stage R&S procedure.

This challenge motivates one to consider formulations other than the plug-in version of (17.31). One popular variant studied by Fan et al. [42] is the distributionally robust R&S problem:

$$\underline{k} = \arg\max_{1\le k\le K} \min_{\theta\in\mathcal{U}} \psi(k,\theta), \tag{17.34}$$

where $\mathcal{U}$ is the uncertainty set that contains the possible values of θ. Specifically, they consider when $\mathcal{U}$ has a finite number of candidate θ values; namely, $\mathcal{U} = \{\theta_1, \theta_2, \ldots, \theta_B\}$. The inner problem of (17.34) finds the worst-case input parameter θ_b in $\mathcal{U}$ for each alternative k, whereas the outer problem selects the alternative with the best worst-case performance. Similar to the generic R&S problem, $\underline{k}$ is estimated by

$$\hat{\underline{k}} = \arg\max_{1\le k\le K} \min_{\theta_b\in\mathcal{U}} \widehat{\psi}(k,\theta_b),$$

where the number of replications allocated to each (k,θ_b) is determined by the procedure. Under this formulation, the probability of correct selection is modified to

$$\text{PCS} = \mathbb{P}\{\hat{\underline{k}} = \underline{k}\}. \tag{17.35}$$

A benefit of Formulation (17.34) is that the input uncertainty is completely characterized by solving the inner minimization problem. By limiting θ to be among a finite number of candidates in $\mathcal{U}$ and simulating all alternative-parameter pairs, it eliminates the need to model the effect of $\widehat{\theta}$ to $\psi(k,\widehat{\theta})$ for each k. Thus, one only needs to control the Monte Carlo error in $\widehat{\psi}(k,\theta_b)$ for each (k,θ_b) to achieve correct selection.

To provide a finite-sample probability guarantee for solving (17.34), Fan et al. [42] extend the IZ formulation in classical R&S procedures. First, they relabeled the performance measures at solution-parameter pairs $\{\psi(k,\theta_b)\}_{1\le k\le K 1\le b\le B}$ such that $\psi_{k,1} \le \psi_{k,2} \le \cdots \le \psi_{k,B}$ for all $1 \le k \le K$ and $\psi_{1,1} > \psi_{2,1} \ge \cdots \ge \psi_{K,1}$. Therefore, $\underline{k} = 1$. The IZ formulation is modified to

$$\psi_{\underline{k},1} - \psi_{2,1} > \delta \tag{17.36}$$

for given $\delta > 0$. That is, the worst-case performance measure of $\underline{k}$ is at least δ better than those of other $k-1$ alternatives. Instead of providing the PCS

guarantee under the IZ assumption, they guarantee the following probability of good selection (PGS):

$$\mathbb{P}\{\psi_{\underline{k},1} - \psi_{\widehat{\underline{k}},1} \leq \delta\} \geq 1 - \alpha. \tag{17.37}$$

In words, (17.37) grants that the selected solution's worst-case performance is within δ from that of $\underline{k}$. If (17.36) holds, then (17.37) is equivalent to (17.35) $\geq 1 - \alpha$. Therefore, δ here can be interpreted as the allowable error tolerance.

Fan et al. [42] further split δ to $\delta_I = \delta_O = \delta/2$, where δ_I is the allowable error in solving the inner-level minimization problem of (17.34). Specifically, they aim to achieve

$$\psi_{k,b^k} - \psi_{k,1} \leq \delta_I, \quad \forall 1 \leq k \leq K,$$

where $b^k = \min_{\theta_b \in \mathcal{U}} \widehat{\psi}(k, \theta_b)$. Assuming the IZ assumption holds, this implies that to make a correct selection at the outer level, ψ_{k,b^k} and ψ_{1,b^1} must be at least $\delta - \delta_I = \delta_O$ apart for all $1 \leq k \leq K$. Similarly, they also split the error level α for inner-level and outer-level comparisons so that the overall probability error is no larger than α. Based on these parameters, Fan et al. [42] propose two-stage and sequential R&S procedures that provide PGS guarantee (17.37) after spending a finite number of replications.

We close this subsection by mentioning that Gao et al. [49] also study (17.34). Instead of creating a R&S procedure with a finite-sample guarantee, they develop an optimal computing budget allocation scheme for (17.34) aiming to maximize the convergence rate of $1 - PCS$ as the simulation budget increases. Shi et al. [115] further extend Gao et al. [49] to solve stochastically constrained version of (17.34).

17.5.1.2 Subset Selection and Multiple Comparisons with the Best

The objective of a subset selection procedure is to return a set of alternatives $\mathcal{I}$ that contains k^* defined in (17.31) with probability $1 - \alpha$:

$$\mathbb{P}\{k^* \in \mathcal{I}\} \geq 1 - \alpha. \tag{17.38}$$

A subset selection procedure does not necessarily guarantee $|\mathcal{I}| = 1$, but when $|\mathcal{I}| = 1$, then the element of $\mathcal{I}$ is indeed k^* with probability of at least $1 - \alpha$.

Corlu and Biller [30] first consider accounting for input uncertainty in a subset selection procedure aiming to guarantee (17.38) under the IZ assumption (17.32). Similar to Song et al. [119], they also find that there is a positive lower bound to δ to guarantee (17.38) when there is input uncertainty. Corlu and Biller [31] take an approach to average out both input uncertainty and Monte Carlo error when they define the optimum. That is, their subset $\mathcal{I}$ contains

$$\bar{k} = \arg\max_{1 \le k \le K} \mathrm{E}_\theta[\psi(k, \theta)] \tag{17.39}$$

with probability no less than $1-\alpha$, where the expectation is taken with respect to the posterior distribution of θ conditional on the input data. A benefit of this approach is that one is guaranteed to have $|\mathcal{I}| = 1$ for sufficiently large simulation budget. However, this formulation may be misleading when the size of the input data is small as no warning is given regarding the risk caused by uncertainty about θ.

The multiple comparisons with the best (MCB) procedure provides simultaneous confidence intervals $[L_k, U_k]$ for all $1 \le k \le K$ such that

$$\mathbb{P}\{\psi(k, \theta) - \psi(k^*, \theta) \in [L_k, U_k], 1 \le k \le K\} \ge 1 - \alpha, \tag{17.40}$$

where $\{L_k, U_k\}_{1 \le k \le K}$ can be constructed from the confidence intervals of the pairwise difference between performance measures [65]:

$$\mathbb{P}\{\psi(k, \theta) - \psi(\ell, \theta) \in [\widehat{\psi}(k, \theta) - \widehat{\psi}(\ell, \theta) \pm w_{k\ell}], \forall k \ne \ell\} \ge 1 - \alpha. \tag{17.41}$$

In words, the intervals in (17.40) cover the difference between each alternative's performance and the optimum's. By design, either L_k or U_k is equal to 0 for each k and if we define $\mathcal{S} = \{k : U_k > 0\}$, then we have $\mathbb{P}\{k^* \in \mathcal{S}\} \ge 1 - \alpha$.

With input uncertainty, θ is unknown. Thus, $\widehat{\psi}(k, \theta)$ in (17.41) is replaced with $\widehat{\psi}(k, \widehat{\theta})$ for each k. As a result, $w_{k\ell}, \forall k \ne \ell$ that satisfy (17.41) comes down to estimating (17.30) under a normality assumption on the simulation outputs for each alternative and regularity conditions on $\widehat{\theta}$. Focusing on the case that all K alternatives' simulators share the same input models, Song and Nelson [118] propose to split $w_{k\ell}$ into two parts, where one covers the difference in the CID effects for solutions k and ℓ, $\{\nabla_\theta \psi(k, \theta) - \nabla_\theta \psi(\ell, \theta)\}'(\widehat{\theta} - \theta)$, and the other covers the stochastic error difference in $\widehat{\psi}(k, \widehat{\theta}) - \widehat{\psi}(\ell, \widehat{\theta})$ conditional on $\widehat{\theta}$. Upon choosing the coverage error appropriately for each interval, they show that the resulting

$\{w_{k\ell}\}_{1\le k\ne\ell\le K}$ provide MCB intervals with asymptotically correct coverage probability.

17.5.2 Global Optimization under Input Uncertainty

Continuous optimization in the stochastic simulation setting is also affected by uncertainty in input distributions used to drive the simulation model. Both searching for the optimal solution and characterizing its error should take input uncertainty into account.

Consider the continuous optimization problem (17.26) where $\mathcal{X}$ is a nonempty subset of $\mathbb{R}^n$. Zhou and Xie [142] provided one of the first approaches for this setting by defining a risk measure, ρ (e.g., expected value, mean-variance, Value-At-Risk). For the parametric case, this can be written $\rho_{\mathbf{P}_\theta|\xi}$ where $\mathbf{P}_\theta|\xi$ is the posterior distribution for $\mathbf{P}$ (i.e., for θ) given the observed input data ξ. Their approach replaces the optimization in (17.27) with $H^\rho(x) = \rho_{\mathbf{P}_\theta|\xi}(\psi(x|\mathbf{P}))$. Although the development was for the parametric case, the authors suggested that the approach could be applied in a nonparametric setting using a Dirichlet process prior. Under general conditions, as the input data sample size goes to infinity, they show that the risk-based objective using posterior $\psi_{\mathbf{P}_\theta|\xi}$ converges in probability to the risk-based objective using $\mathbf{P}$. A stochastic approximation method for this approach and associated stochastic gradient estimators are presented in [21].

When the performance measure ψ can be modeled as a Gaussian process (GP) over x, θ space, Bayesian methods can be employed to include input uncertainty in the GP model optimization process given a prior distribution $\mathbf{G}$ for the unknown θ. Then the optimization is of $\mathrm{E}_{\mathbf{G}(\theta)}(\psi(x,\theta))$. In [98] efficient global optimization (EGO—see [69]) and knowledge gradient (KG—see [113]) sequential optimization methods were modified to include input uncertainty. Also in the GP setting, [74] proposed a robust optimization method based on Kullback-Leibler distance with a modified EGO criterion.

Wang et al. [131] modified the method of Pearce and Branke [98] to determine the next x, θ pair sequentially (x then θ) rather than simultaneously. The θ value is selected by minimizing IMSE or IMSE weighted by the posterior distribution for θ. They showed somewhat better results for KG search with small data samples and similar results in EGO and other KG settings while reducing computational time. In addition to EGO and KG, they also incorporated the two-stage search with Informational Approach for Global Optimization (IAGO—[129]) and Expected Excursion

Volume (EEV—[100]) metrics for selecting search points in the GP-based optimization.

Bayesian optimization has also been employed for handling the nonparametric optimization case posed in (17.26). In two recent papers, Wang et al. [130, 132] approached the simulation optimization problem similarly to [98] but employed Bayesian updating of the unknown input distribution, beginning with a (presumably diffuse) Dirichlet process prior.

When there is the option to collect more input-model data or continue an optimization, Ungredda et al. [126] suggest an extension to the approach in Pearce et al. [98] that they call Bayesian Information Collection and Optimisation (BICO). The decision is based on a Value of Information (VoI) measure for an additional simulation evaluation (at cost c_f) vs. the VoI for an additional input data sample (at cost c_s).

17.5.3 Online Optimization with Streaming Input Data

Thus far, our discussion in Section 17.5 has focused on the case when a batch of finite input data is available, but no additional data can be collected. In many practical problems, however, a stream of input data may be collected continuously. Assuming that the input data-generating process is stationary, Wu and Zhou [135] propose a R&S framework to solve Problem (17.39) by continuously updating the posterior distribution of θ from the sequence of incoming data. They also consider the case when there is a trade-off between real-world data collection versus simulation replication and study a computational budget allocation scheme.

Song and Shanbhag [122] study a simulation optimization problem with continuous variables when a sequence of streaming data is made available at each period. Their propose to update $\widehat{\theta}$ at each period using the cumulative data and solve the plug-in version of (17.27) given $\widehat{\theta}$ using stochastic approximation (SA). Under a strong convexity assumption, they derive an upper bound on the expected optimality gap of the solution returned from SA for the original problem (17.27) as a function of number of SA steps taken as well as the sample size of accumulated data in the period. From the upper bound, they propose a stopping criterion for SA at each period, i.e., they take SA steps until the error rate of SA matches that from the finite-sample size.

Both work mentioned here assume that the streaming data are independent and identically distributed. A framework that can account for a nonstationary data-generating process will broaden the applicability of the online simulation optimization schemes.

17.6 Future Research Directions

As computation resources increase in power with decreasing cost, simulation analysis has evolved from accepting the point estimate of the input distribution as *the truth* to incorporating the estimation error in uncertainty quantification and in simulation optimization. However, many open questions remain to be addressed.

First, the majority of work introduced in this chapter assumes the collected data are i.i.d. observations from real-world distributions that do not change over time. In many applications, such assumptions fail to hold due to dependence among the sequence of observations or nonstationarity of the data-generating process. Also, even if the real-world distribution can be characterized as generating i.i.d. input vectors, only marginal observations may be available so that the dependence structure cannot be estimated in a straightforward way. Among the tools that have been investigated in the input uncertainty literature, robust optimization and robust simulation come closest to handling such issues, but there remains much work to be done including: (i) making the methodology computationally efficient, (ii) quantifying the reliability when facing nonstationarity in a rigorous statistical sense, and (iii) exploring alternative methods to tackle these challenges.

In some cases, the input data themselves are unobservable and we can only access the "output data" from the real-world system. For instance, in a queueing system, we may observe the departure times of the jobs, but not their arrival times nor service times. In this case, finding the "right" input model for the simulator can be viewed as a calibration problem. However, unlike a typical calibration problem in the computer experiment literature that calibrates the model parameter so that the model output matches a given benchmark, here, we need to choose the input model so that the sequence of outputs generated from the simulator matches the real-world output data. This can be a high-dimensional inference problem that faces difficult statistical challenges, including the unavailability of the likelihood function, non-identifiability issues for over-parametrized models (i.e., more than one set of parameter values give the simulation-real output match), and model bias (i.e., the best fitting model in the class has parameter values bearing a discrepancy with reality). Both traditional model validation tools and more recent approaches on this problem require further developments. Moreover, there are several other open questions, including what metric should be adopted to measure the discrepancy between real and simulation outputs, and how to incorporate uncertainty from both input and output data.

Lastly, creating more application-specific methods and addressing challenges in these areas will enrich the literature. Although input uncertainty appears ubiquitous in simulation applications, there has been little work in applying input uncertainty methods to support decision-making. For instance, a company may run a market simulation study where the key input being the utility parameters that customers use to decide which product to buy (if any) among the competing offers. One way to estimate the utility parameters is to survey (often a very small fraction of) the customer basis. Thus, the estimation error of the utility parameters may cause significant uncertainty in the sales prediction obtained from the simulation study. Without quantifying input uncertainty, the company may face a significant risk. How to quantify and mitigate the uncertainty, both statistically and in a computationally efficient way, is an important question to resolve.

References

1. Ankenman, B., Nelson, B. L., and Staum, J. (2010). Stochastic kriging for simulation metamodeling. *Operations Research*, 58(2):371–382.
2. Asmussen, S. and Glynn, P. W. (2007). *Stochastic Simulation: Algorithms and Analysis*, volume 57. Springer Science & Business Media.
3. Bai, Y., Balch, T., Chen, H., Dervovic, D., Lam, H., and Vyetrenko, S. (2021). Calibrating over-parametrized simulation models: A framework via eligibility set. *arXiv preprint* arXiv:2105.12893.
4. Balci, O. and Sargent, R. G. (1982). Some examples of simulation model validation using hypothesis testing. In *Proceedings of the 14th Winter Simulation Conference*, volume 2, pages 621–629.
5. Barton, R. R. and Schruben, L. W. (2001). Resampling methods for input modeling. In *Proceedings of the 2001 Winter Simulation Conference*, pages 372–378. IEEE.
6. Barton, R. R. (2012). Tutorial: Input uncertainty in outout analysis. In *Proceedings of the 2012 Winter Simulation Conference*, pages 67–78. IEEE.
7. Barton, R. R., Chick, S. E., Cheng, R. C., Henderson, S. G., Law, A. M., Schmeiser, B. W., Leemis, L. M., Schruben, L. W., and Wilson, J. R. (2002). Panel discussion on current issues in input modeling. In *Proceedings of the 2002 Winter Simulation Conference*, pages 353–369. IEEE.
8. Barton, R. R., Lam, H., and Song, E. (2018). Revisiting direct bootstrap resampling for input model uncertainty. In *Proceedings of the 2018 Winter Simulation Conference*, pages 1635–1645. IEEE.
9. Barton, R. R., Nelson, B. L., and Xie, W. (2014). Quantifying input uncertainty via simulation confidence intervals. *INFORMS Journal on Computing*, 26(1):74–87.

10. Barton, R. R. and Schruben, L. W. (1993). Uniform and bootstrap resampling of empirical distributions. In *Proceedings of the 1993 Winter Simulation Conference*, pages 503–508. IEEE.
11. Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H., and Tu, J. (2007). A framework for validation of computer models. *Technometrics*, 49(2):138–154.
12. Bayraksan, G. and Love, D. K. (2015). Data-driven stochastic programming using phi-divergences. In *Tutorials in Operations Research*, pages 1–19. INFORMS.
13. Bechhofer, R. E. (1954). A single-sample multiple decision procedure for ranking means of normal populations with known variances. *The Annals of Mathematical Statistics*, 25(1):16 – 39.
14. Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., and Rennen, G. (2013). Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357.
15. Ben-Tal, A., El Ghaoui, L., and Nemirovski, A. (2009). *Robust Optimization*. Princeton University Press.
16. Bertsimas, D., Brown, D. B., and Caramanis, C. (2011). Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501.
17. Bertsimas, D., Gupta, V., and Kallus, N. (2018). Robust sample average approximation. *Mathematical Programming*, 171(1-2):217–282.
18. Blanchet, J., Kang, Y., and Murthy, K. (2019). Robust Wasserstein profile inference and applications to machine learning. *Journal of Applied Probability*, 56(3):830-857.
19. Blanchet, J. and Murthy, K. (2019). Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600.
20. Box, G. E. P. (1976). Science and statistics. *Journal of the American Statistical Association*, 71(356):791–799.
21. Cakmak, S., Wu, D., and Zhou, E. (2020). Solving Bayesian risk optimization via nested stochastic gradient estimation. *arXiv:2007.05860*.
22. Chen, L., Ma, W., Natarajan, K., Simchi-Levi, D., and Yan, Z. (2018). Distributionally robust linear and discrete optimization with marginals. *Available at SSRN 3159473*.
23. Chen, R. and Paschalidis, I. C. (2018). A robust learning approach for regression models based on distributionally robust optimization. *Journal of Machine Learning Research*, 19(13):1–48.
24. Cheng, R. C. and Holland, W. (1997). Sensitivity of computer simulation experiments to errors in input data. *Journal of Statistical Computation and Simulation*, 57(1–4):219–241.
25. Cheng, R. C. and Holland, W. (1998). Two-point methods for assessing variability in simulation output. *Journal of Statistical Computation and Simulation*, 60(3):183–205.

26. Cheng, R. C. H. and Holland, W. (2004). Calculation of confidence intervals for simulation output. *ACM Transactions on Modeling and Computer Simulation*, 14(4).
27. Chick, S. E. (2001). Input distribution selection for simulation experiments: Accounting for input uncertainty. *Operations Research*, 49(5):744–758.
28. Chick, S. E. (2006). Bayesian ideas and discrete event simulation: Why, what and how. In Perrone, L. F., Wieland, F. P., Liu, J., Lawson, B. G., Nicol, D. M., and Fujimoto, R. M., editors, *Proceedings of the 2006 Winter Simulation Conference*, pages 96–106. IEEE.
29. Corlu, C. G., Akcay, A., and Xie, W. (2020). Stochastic simulation under input uncertainty: A Review. *Operations Research Perspectives*, 7:100162.
30. Corlu, C. and Biller, B. (2013). A subset selection procedure under input parameter uncertainty. In *Proceedings of the 2013 Winter Simulation Conference*, pages 463–473. IEEE.
31. Corlu, C. G. and Biller, B. (2015). Subset selection for simulations accounting for input uncertainty. In *Proceedings of the 2015 Winter Simulation Conference*, pages 437–446. IEEE.
32. Cranmer, K., Brehmer, J., and Louppe, G. (2020). The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062.
33. Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D. (1991). Bayesian prediction of deterministic functions, with applications to the design and analysis of computer experiments. *Journal of the American Statistical Association*, 86(416):953–963.
34. Davison, A. C. and Hinkley, D. V. (1997). *Bootstrap Methods and their Application*. Number 1. Cambridge University Press.
35. Delage, E. and Ye, Y. (2010). Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612.
36. Dhara, A., Das, B., and Natarajan, K. (2021). Worst-case expected shortfall with univariate and bivariate marginals. *INFORMS Journal on Computing*, 33(1):370–389.
37. Doan, X. V., Li, X., and Natarajan, K. (2015). Robustness to dependency in portfolio optimization using overlapping marginals. *Operations Research*, 63(6):1468–1488.
38. Duchi, J., Glynn, P., and Namkoong, H. (2016). Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint* arXiv:1610.03425.
39. Duchi, J. C., Glynn, P. W., and Namkoong, H. (2021). Statistics of robust optimization: A generalized empirical likelihood approach. *Mathematics of Operations Research*, 46(3):946–969.
40. Efron, B. and Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. CRC press.

41. Esfahani, P. M. and Kuhn, D. (2018). Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1-2):115–166.
42. Fan, W., Hong, L. J., and Zhang, X. (2020). Distributionally robust selection of the best. *Management Science*, 66(1):190–208.
43. Feng, M. and Song, E. (2019). Efficient input uncertainty quantification via green simulation using sample path likelihood ratios. In *Proceedings of the 2019 Winter Simulation Conference*, pages 3693–3704. IEEE.
44. Feng, M. and Staum, J. (2015). Green simulation designs for repeated experiments. In *Proceedings of the 2015 Winter Simulation Conference*, pages 403–413. IEEE.
45. Feng, M. and Staum, J. (2017). Green simulation: Reusing the output of repeated experiments. *ACM Transactions on Modeling and Computer Simulation*, 27(4):1–28.
46. Feng, M. B. and Song, E. (2021). Optimal nested simulation experiment design via likelihood ratio method. *arXiv preprint* arXiv:2008.13087v2.
47. Fox, B. L. and Glynn, P. W. (1989). Replication schemes for limiting expectations. *Probability in the Engineering and Informational Sciences*, 3(3):299–318.
48. Gao, R. and Kleywegt, A. J. (2016). Distributionally robust stochastic optimization with Wasserstein distance. *arXiv preprint*arXiv:1604.02199.
49. Gao, S., Xiao, H., Zhou, E., and Chen, W. (2017). Robust ranking and selection with optimal computing budget allocation. *Automatica*, 81:30–36.
50. Ghaoui, L. E., Oks, M., and Oustry, F. (2003). Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Operations Research*, 51(4):543–556.
51. Ghosh, S. and Lam, H. (2015). Mirror descent stochastic approximation for computing worst-case stochastic input models. In *Proceedings of the 2015 Winter Simulation Conference*, pages 425–436. IEEE.
52. Ghosh, S. and Lam, H. (2019). Robust analysis in stochastic simulation: Computation and performance guarantees. *Operations Research*, 67(1):232–249.
53. Glasserman, P. and Xu, X. (2014). Robust risk measurement and model risk. *Quantitative Finance*, 14(1):29–58.
54. Glasserman, P. and Yang, L. (2018). Bounding wrong-way risk in CVA calculation. *Mathematical Finance*, 28(1):268–305.
55. Glynn, P. W. (1990). Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84.
56. Glynn, P. W. and Iglehart, D. L. (1990). Simulation output analysis using standardized time series. *Mathematics of Operations Research*, 15(1):1–16.
57. Glynn, P. W. and Lam, H. (2018). Constructing simulation output intervals under input uncertainty via data sectioning. In *Proceedings of the 2018 Winter Simulation Conference*, pages 1551–1562. IEEE.
58. Goeva, A., Lam, H., Qian, H., and Zhang, B. (2019). Optimization-based calibration of simulation input models. *Operations Research*, 67(5):1362–1382.

59. Goeva, A., Lam, H., and Zhang, B. (2014). Reconstructing input models via simulation optimization. In *Proceedings of the 2014 Winter Simulation Conference*, pages 698–709. IEEE.
60. Goh, J. and Sim, M. (2010). Distributionally robust optimization and its tractable approximations. *Operations Research*, 58(4):902–917.
61. Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393.
62. Hanasusanto, G. A., Roitch, V., Kuhn, D., and Wiesemann, W. (2015). A distributionally robust perspective on uncertainty quantification and chance constrained programming. *Mathematical Programming*, 151(1):35–62.
63. Henderson, S. G. (2003). Input modeling: Input model uncertainty: Why do we care and what should we do about it? In Chick, S., Sánchez, P. J., Ferrin, D., and Morrice, D. J., editors, *Proceedings of the 2003 Winter Simulation Conference*, pages 90–100. IEEE.
64. Higdon, D., Gattiker, J., Williams, B., and Rightley, M. (2008). Computer model calibration using high-dimensional output. *Journal of the American Statistical Association*, 103(482):570–583.
65. Hsu, J. C. (1984). Constrained simultaneous confidence intervals for multiple comparisons with the best. *Annals of Statististics*, 12(3):1136–1144.
66. Hu, Z., Cao, J., and Hong, L. J. (2012). Robust simulation of global warming policies using the DICE model. *Management Science*, 58(12):2190–2206.
67. Hu, Z. and Hong, L. J. (2015). Robust simulation of stochastic systems with input uncertainties modeled by statistical divergences. In *2015 Winter Simulation Conference*, pages 643–654. IEEE.
68. Jiang, R. and Guan, Y. (2016). Data-driven chance constrained stochastic program. *Mathematical Programming*, 158(1):291–327.
69. Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492.
70. Kennedy, M. C. and O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 63(3):425–464.
71. Kim, S.-H. and Nelson, B. L. (2001). A fully sequential procedure for indifference-zone selection in simulation. *ACM Transactions on Modeling and Computer Simulation*, 11(3):251–273.
72. Kim, S.-H. and Nelson, B. L. (2006). Chapter 17 selecting the best system. In Henderson, S. G. and Nelson, B. L., editors, *Simulation*, volume 13 of *Handbooks in Operations Research and Management Science*, pages 501–534. Elsevier.
73. Kleijnen, J. P. (1995). Verification and validation of simulation models. *European Journal of Operational Research*, 82(1):145–162.
74. Lakshmanan, S. and Venkateswaran, J. (2017). Robust simulation based optimization with input uncertainty. In *Proceedings of the 2017 Winter Simulation Conference*, pages 2257–2267. IEEE.

75. Lam, H. (2016a). Advanced tutorial: Input uncertainty and robust analysis in stochastic simulation. In *Proceedings of the 2016 Winter Simulation Conference*, pages 178–192. IEEE.
76. Lam, H. (2016b). Robust sensitivity analysis for stochastic systems. *Mathematics of Operations Research*, 41(4):1248–1275.
77. Lam, H. (2018). Sensitivity to serial dependency of input processes: A robust approach. *Management Science*, 64(3):1311–1327.
78. Lam, H. (2019). Recovering best statistical guarantees via the empirical divergence-based distributionally robust optimization. *Operations Research*, 67(4):1090–1105.
79. Lam, H. and Mottet, C. (2017). Tail analysis without parametric models: A worst-case perspective. *Operations Research*, 65(6):1696–1711.
80. Lam, H. and Qian, H. (2016). The empirical likelihood approach to simulation input uncertainty. In *Proceedings of the 2016 Winter Simulation Conference*, pages 791-802. IEEE.
81. Lam, H. and Qian, H. (2017). Optimization-based quantification of simulation input uncertainty via empirical likelihood. *arXiv preprint*arXiv:1707.05917.
82. Lam, H. and Qian, H. (2018a). Subsampling to enhance efficiency in input uncertainty quantification. *Operations Research*, published online in *Articles in Advance*, 03 Dec 2021.
83. Lam, H. and Qian, H. (2018b). Subsampling variance for input uncertainty quantification. In *2018 Winter Simulation Conference*, pages 1611–1622. IEEE.
84. Lam, H. and Qian, H. (2019). Random perturbation and bagging to quantify input uncertainty. In *2019 Winter Simulation Conference*, pages 320–331. IEEE.
85. Lam, H. and Zhang, J. (2020). Distributionally constrained stochastic gradient estimation using noisy function evaluations. In *Proceedings of the 2020 Winter Simulation Conference*, pages 445–456. IEEE.
86. Lam, H., Zhang, X., and Plumlee, M. (2017). Improving prediction from stochastic simulation via model discrepancy learning. In *Proceedings of the 2017 Winter Simulation Conference*, pages 1808–1819. IEEE.
87. Lam, H. and Zhou, E. (2017). The empirical likelihood approach to quantifying uncertainty in sample average approximation. *Operations Research Letters*, 45(4):301–307.
88. Lewis, P. A. and Orav, E. J. (2017). *Simulation Methodology for Statisticians, Operations Analysts, and Engineers*. Chapman and Hall/CRC.
89. Li, B., Jiang, R., and Mathieu, J. L. (2017). Ambiguous risk constraints with moment and unimodality information. *Mathematical Programming*, 173:151–192.
90. Miller, B. L. and Wagner, H. M. (1965). Chance constrained programming with joint constraints. *Operations Research*, 13(6):930–945.

91. Morgan, L. E., Nelson, B. L., Titman, A. C., and Worthington, D. J. (2019). Detecting bias due to input modelling in computer simulation. *European Journal of Operational Research*, 279(3):869–881.
92. Nelson, B. (2013). *Foundations and Methods of Stochastic Simulation: A First Course*. Springer Science & Business Media.
93. Ng, S. H. and Chick, S. E. (2006). Reducing parameter uncertainty for stochastic systems. *ACM Transactions on Modeling and Computer Simulation*, 16(1):26–51.
94. Oakley, J. E. and O'Hagan, A. (2004). Probabilistic sensitivity analysis of complex models: a Bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3):751–769.
95. Oakley, J. E. and Youngman, B. D. (2017). Calibration of stochastic computer simulators using likelihood emulation. *Technometrics*, 59(1):80–92.
96. Owen, A. B. (2001). *Empirical Likelihood*. CRC press.
97. O'Hagan, A., Kennedy, M. C., and Oakley, J. E. (1999). Uncertainty analysis and other inference tools for complex computer codes. In Bernardo, J., Berger, J., Dawid, A., and Smith, A., editors, *Bayesian Statistics 6: Proceedings of the Sixth Valencia International Meeting*, pages 503–524. Oxford Science Publications.
98. Pearce, M. and Branke, J. (2017). Bayesian simulation optimization with input uncertainty. In *Proceedings of the 2017 Winter Simulation Conference*, pages 2268–2278. IEEE.
99. Phuong Le, H. and Branke, J. (2020). Bayesian optimization searching for robust solutions. In *Proceedings of the 2020 Winter Simulation Conference*, pages 2844–2855. IEEE.
100. Picheny, V. (2015). Multiobjective optimization using Gaussian process emulators via stepwise uncertainty reduction. *Statistics and Computing*, 25(6):1265–1280.
101. Plumlee, M. and Lam, H. (2016). Learning stochastic model discrepancy. In *Proceedings of the 2016 Winter Simulation Conference*, pages 413–424. IEEE.
102. Popescu, I. (2005). A semidefinite programming approach to optimal-moment bounds for convex classes of distributions. *Mathematics of Operations Research*, 30(3):632–657.
103. Reiman, M. I. and Weiss, A. (1989). Sensitivity analysis for simulations via likelihood ratios. *Operations Research*, 37(5):830–844.
104. Rinott, Y. (1978). On two-stage selection procedures and related probability-inequalities. *Communications in Statistics - Theory and Methods*, 7(8):799–811.
105. Rubin, D. B. (1981). The Bayesian bootstrap. *The Annals of Statistics*, 9(1):130–134.
106. Rubinstein, R. Y. (1986). The score function approach for sensitivity analysis of computer simulation models. *Mathematics and Computers in Simulation*, 28(5):351–379.
107. Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. John Wiley & Sons.

108. Sargent, R. G. (2005). Verification and validation of simulation models. In *Proceedings of the 2005 Winter Simulation Conference*, pages 130–143. IEEE.
109. Schmeiser, B. (1982). Batch size effects in the analysis of simulation output. *Operations Research*, 30(3):556–568.
110. Schruben, L. (1983). Confidence interval estimation using standardized time series. *Operations Research*, 31(6):1090–1108.
111. Schruben, L. and Kulkarni, R. (1982). Some consequences of estimating parameters for the M/M/1 queue. *Operations Research Letters*, 1(2):75–78.
112. Schruben, L. W. (1980). Establishing the credibility of simulations. *Simulation*, 34(3):101–105.
113. Scott, W., Frazier, P., and Powell, W. (2011). The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM Journal on Optimization*, 21(3):996–1026.
114. Shafer, G. (1976). Statistical evidence. In *A Mathematical Theory of Evidence*, pages 237–273. Princeton University Press.
115. Shi, Z., Gao, S., Xiao, H., and Chen, W. (2019). A worst-case formulation for constrained ranking and selection with input uncertainty. *Naval Research Logistics*, 66(8):648–662.
116. Song, E. (2021). Sequential bayesian risk set inference for robust discrete optimization via simulation. *arXiv preprint*arXiv:2101.07466.
117. Song, E. and Nelson, B. L. (2015). Quickly assessing contributions to input uncertainty. *IIE Transactions*, 47(9):893–909.
118. Song, E. and Nelson, B. L. (2019). Input–output uncertainty comparisons for discrete optimization via simulation. *Operations Research*, 67(2):562–576.
119. Song, E., Nelson, B. L., and Hong, L. J. (2015). Input uncertainty and indifference-zone ranking & selection. In *Proceedings of the 2015 Winter Simulation Conference*, pages 414–424. IEEE.
120. Song, E., Nelson, B. L., and Pegden, C. D. (2014). Advanced tutorial: Input uncertainty quantification. In *Proceedings of the 2014 Winter Simulation Conference*, pages 162–176. IEEE.
121. Song, E., Nelson, B. L., and Staum, J. (2016). Shapley effects for global sensitivity analysis: Theory and computation. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):1060–1083.
122. Song, E. and Shanbhag, U. V. (2019). Stochastic approximation for simulation optimization under input uncertainty with streaming data. In *Proceedings of the 2019 Winter Simulation Conference*, pages 3597–3608. IEEE.
123. Sun, Y., Apley, D. W., and Staum, J. (2011). Efficient nested simulation for estimating the variance of a conditional expectation. *Operations Research*, 59(4):998–1007.
124. Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM.
125. Tuo, R., Wu, C. J., et al. (2015). Efficient calibration for imperfect computer models. *The Annals of Statistics*, 43(6):2331–2352.

126. Ungredda, J., Pearce, M., and Branke, J. (2020). Bayesian optimisation vs. input uncertainty reduction. *arXiv:2006.00643*.
127. Van Parys, B. P., Goulart, P. J., and Kuhn, D. (2016). Generalized Gauss inequalities via semidefinite programming. *Mathematical Programming*, 156(1-2):271–302.
128. Van der Vaart, A. W. (2000). *Asymptotic Statistics*, volume 3. Cambridge University Press.
129. Villemonteix, J., Vazquez, E., and Walter, E. (2008). An informational approach to the global optimization of expensive-to-evaluate functions. *Journal of Global Optimization*, 44(4):509.
130. Wang, H., Ng, S. H., and Zhang, X. (2020a). A Gaussian process based algorithm for stochastic simulation optimization with input distribution uncertainty. In *Proceedings of the 2020 Winter Simulation Conference*, pages 2899–2910. IEEE.
131. Wang, H., Yuan, J., and Ng, S. H. (2020b). Gaussian process based optimization algorithms with input uncertainty. *IISE Transactions*, 52(4):377–393.
132. Wang, H., Zhang, X., and Ng, S. H. (2021). A nonparametric Bayesian approach for simulation optimization with input uncertainty. *arXiv:2008.02154*.
133. Wiesemann, W., Kuhn, D., and Sim, M. (2014). Distributionally robust convex optimization. *Operations Research*, 62(6):1358–1376.
134. Wong, R. K. W., Storlie, C. B., and Lee, T. C. M. (2017). A frequentist approach to computer model calibration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(2):635–648.
135. Wu, D. and Zhou, E. (2017). Ranking and selection under input uncertainty: fixed confidence and fixed budget. *arXiv preprint*arXiv:1708.08526.
136. Xie, W., Li, C., Wu, Y., and Zhang, P. (2021). A Bayesian nonparametric framework for uncertainty quantification in simulation. *SIAM Journal on Uncertainty Quantification*, 9(4):1527–1552.
137. Xie, W., Nelson, B. L., and Barton, R. R. (2014). A Bayesian framework for quantifying uncertainty in stochastic simulation. *Operations Research*, 62(6):1439–1452.
138. Xie, W., Nelson, B. L., and Barton, R. R. (2016). Multivariate input uncertainty in output analysis for stochastic simulation. *ACM Transactions on Modeling and Computer Simulation*, 27(1):5:1–5:22.
139. Xu, J., Zheng, Z., and Glynn, P. W. (2020). Joint resource allocation for input data collection and simulation. In *Proceedings of the 2020 Winter Simulation Conference*, pages 2126–2137. IEEE.
140. Zazanis, M. A. and Suri, R. (1993). Convergence rates of finite-difference sensitivity estimates for stochastic systems. *Operations Research*, 41(4):694–703.
141. Zhou, E. and Liu, T. (2018). Online quantification of input uncertainty for parametric models. In *Proceedings of the 2018 Winter Simulation Conference*, pages 1587–1598. IEEE.

142. Zhou, E. and Xie, W. (2015). Simulation optimization when facing input uncertainty. In *Proceedings of the 2015 Winter Simulation Conference*, pages 3714–3724. IEEE.
143. Zouaoui, F. and Wilson, J. R. (2003). Accounting for parameter uncertainty in simulation input modeling. *IIE Transactions*, 35(9):781–792.
144. Zouaoui, F. and Wilson, J. R. (2004). Accounting for input-model and input-parameter uncertainties in simulation. *IIE Transactions*, 36(11):1135–1151.