# Simulation Methodology: An Overview

Peter W. Glynn Stanford University

Theory of RL Bootcamp, Simons Institute September 4, 2020

## **Outline:**

- I. Simulation: Basic Terminology
- II. Connection to Numerical Integration
- III. The Monte Carlo Method
- IV. Dimensional Insensitivity for Monte Carlo
- V. Quasi-Random Sequences
- **VI.** Output Analysis
- VII. Replication
- VIII. Sub-Sampling
  - IX. Output Analysis in Parallel Computing Context

# I. Simulation: Basic Terminology

Simulation:

Generate trajectories of a dynamical system

$$x_{n+1} = g(x_n)$$

or

$$\frac{d}{dt}x(t) = \mu(x(t))$$

Stochastic Simulation:

Generate trajectories of a (stochastic) dynamical system

$$X_{n+1} = g(X_n, \xi_{n+1})$$

or

$$\frac{d}{dt}X(t) = \mu(X(t)) + \xi(t)$$

# **II. Connection to Numerical Integration**

<u>Goal</u>: Compute  $\alpha = E[W]$ , where  $W = f(X_0, X_1, \dots, X_T)$ 

Method:

• Generate n iid replications  $W_1, W_2, \ldots, W_n$  of W

 $\blacktriangleright$  Estimate  $\alpha$  via

$$\alpha_n = \frac{1}{n} \sum_{i=1}^n W_i$$

Note that if

then

$$X_{i+1} = g(X_i, \xi_{i+1})$$

$$\mathbf{s/t} \quad X_0 = x$$

$$\alpha = E[W]$$

$$= E [f(X_0, X_1, \dots, X_d)]$$

$$= E \left[ \widetilde{f}(\mathbf{s}_1, \dots, \mathbf{s}_d) \right]$$

$$= \int_{\mathbb{R}^d} \widetilde{f}(z_1, \dots, z_d) \prod_{i=1}^d h_i(z_i) dz_i$$

Such an expectation can be expressed as a *d*-dimensional integral

Typically, with  $d\ {\rm large}$ 

Conversely, if

$$\begin{split} \alpha &= \int_{\mathbb{R}^d} q(z_1, \dots, z_d) dz_1 \dots dz_d \\ &= \int_{\mathbb{R}^d} \frac{q(z_1, \dots, z_d)}{\prod_{i=1}^d h_i(z_i)} \prod_{i=1}^d h_i(z_i) \, d\mathbf{z}_i \\ &= E\left[\frac{q(Z_1, \dots, Z_d)}{\prod_{i=1}^d h_i(Z_i)}\right] \end{split}$$

Every *d*-dimensional integral can be represented as an expectation

Using sampling-based methods to compute (higher dimensional) integrals is known as the *Monte Carlo* method

Stochastic Simulation  $\iff$  *Monte Carlo* Method

## III. The Monte Carlo Method

<u>Goal</u>: Compute  $\alpha = E[W]$ 

Method:

• Generate n iid replications  $W_1, W_2, \ldots, W_n$  of W

Form

$$\alpha_n = \overline{W}_n = \frac{1}{n} \sum_{i=1}^n W_i$$

## III. The Monte Carlo Method

<u>Goal</u>: Compute  $\alpha = E[W]$ 

Method:

Generate n iid replications W<sub>1</sub>, W<sub>2</sub>,..., W<sub>n</sub> of W
 Form

$$\alpha_n = \overline{W}_n = \frac{1}{n} \sum_{i=1}^n W_i$$

Proof of validity: Law of Large Numbers (LLN)

$$\alpha_n \stackrel{a.s.}{\to} \alpha$$

as  $n \to \infty$ 

Convergence rate analysis:

Central Limit Theorem (CLT): If  $\sigma^2 = Var(\mathcal{G}) < \infty$ , then

$$\sqrt{n}(\alpha_n - \alpha) \Rightarrow \sigma N(0, 1)$$

as  $n \to \infty$ 

Informally,

$$\alpha_n \stackrel{D}{\approx} \alpha + \frac{\sigma}{\sqrt{n}} N(0, 1)$$

Convergence rate analysis:

Central Limit Theorem (CLT): If  $\sigma^2 = Var(\mathcal{U}) < \infty$ , then

$$\sqrt{n}(\alpha_n - \alpha) \Rightarrow \sigma N(0, 1)$$

as  $n \to \infty$ 

Informally,

$$\alpha_n \stackrel{D}{\approx} \alpha + \frac{\sigma}{\sqrt{n}} N(0, 1)$$

Implications:

- Slow convergence rate
- $\blacktriangleright$  Problem hardness characterized by a single constant  $\sigma$
- Slow convergence rate suggests error assessment is important

Error assessment via asymptotically valid confidence intervals

$$P\left(\alpha \in \left[\alpha_n - z\frac{\sigma}{\sqrt{n}}, \alpha_n + z\frac{\sigma}{\sqrt{n}}\right]\right) \to 1 - \delta$$

where z is selected to that  $P(-z \leq N(0,1) \leq z) = 1-\delta$ 

Error assessment via asymptotically valid confidence intervals

$$P\left(\alpha \in \left[\alpha_n - z\frac{\sigma}{\sqrt{n}}, \alpha_n + z\frac{\sigma}{\sqrt{n}}\right]\right) \to 1 - \delta$$

where z is selected to that  $P(-z \leq N(0,1) \leq z) = 1-\delta$ 

 $\sigma^2$  is unknown but can be estimated (internally, from the sample) via

$$s_n^2 = \frac{1}{n-1} \sum_{i=1}^n (W_i - \overline{W}_n)^2$$

So,

$$\left[\alpha_n - z\frac{s_n}{\sqrt{n}}, \alpha_n + z\frac{s_n}{\sqrt{n}}\right]$$

is an approximate 100(1  $-\alpha)\%$  Cl

# Asymptotic Validity vs Hard Error Bounds

Asymptotic validity:

• If 
$$\sigma^2 = \operatorname{Var}(Z_1) < \infty$$
, then  

$$P\left(\alpha \in \left[\alpha_n - z\frac{s_n}{\sqrt{n}}, \alpha_n + z\frac{s_n}{\sqrt{n}}\right]\right) \to 1 - \delta$$

as  $n \to \infty$ 

 $\blacktriangleright$  No guarantee for fixed n

#### Asymptotic Validity vs Hard Error Bounds

Asymptotic validity:  $\checkmark$ If  $\sigma^2 = Var(2) < \infty$ , then

$$P\left(\alpha \in \left[\alpha_n - z\frac{s_n}{\sqrt{n}}, \alpha_n + z\frac{s_n}{\sqrt{n}}\right]\right) \to 1 - \delta$$

as  $n \to \infty$ 

 $\blacktriangleright$  No guarantee for fixed n

Hard error bounds:

Chebyshev's inequality:

$$P\left(\alpha \in \left[\alpha_n - \frac{\epsilon}{\sqrt{n}}, \alpha_n + \frac{\epsilon}{\sqrt{n}}\right]\right) \ge 1 - \frac{c^2}{\epsilon^2}$$

if  $P(|W| \le c) = 1$ , so we have a hard error bound

# Asymptotic Validity vs Hard Error Bounds

Asymptotic validity:

• If 
$$\sigma^2 = \operatorname{Var}(Z_1) < \infty$$
, then  

$$P\left(\alpha \in \left[\alpha_n - z\frac{s_n}{\sqrt{n}}, \alpha_n + z\frac{s_n}{\sqrt{n}}\right]\right) \to 1 - \delta$$

as  $n \to \infty$ 

 $\blacktriangleright$  No guarantee for fixed n

Hard error bounds:

Chebyshev's inequality:

$$P\left(\alpha \in \left[\alpha_n - \frac{\epsilon}{\sqrt{n}}, \alpha_n + \frac{\epsilon}{\sqrt{n}}\right]\right) \ge 1 - \frac{c^2}{\epsilon^2}$$

if  $P(|W| \le c) = 1$ , so we have a hard error bound

The great majority of *Monte Carlo* theory focuses on asymptotic validity (using limit theorems)

#### **IV.** Dimensional Insensitivity for Monte Carlo

Goal: Compute

$$\alpha(\widetilde{f}) = E\left[\widetilde{f}(U_1, \dots, U_d)\right] \stackrel{\Delta}{=} E\left[W(\widetilde{f})\right]$$

where the  $U_i$ 's are iid uniform on [0,1]

For any (weighted) integration rule, it is known that

$$\sup_{\widetilde{f} \in C^{r}(k)} \left| \alpha_{c}(\widetilde{f}) - \alpha(\widetilde{f}) \right| = O\left( c^{-r/d} \right)$$

as  $c \to \infty$ 

"curse of dimensionality"

- ▶ Put  $\alpha_n(\widetilde{f}) = n^{-1} \sum_{i=1}^n W_i(\widetilde{f})$
- Chebyshev implies that if  $|\tilde{f}| \leq k$ , then

$$P\left(\left|\alpha_n(\widetilde{f}) - \alpha(\widetilde{f})\right| > \frac{\epsilon}{\sqrt{n}}\right) \le \frac{k^2}{\epsilon^2}$$

► For a given computational budget c, n ≈ c/d. So,

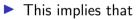
$$P\left(\left|\alpha_{c}(\widetilde{f}) - \alpha(\widetilde{f})\right| > \epsilon \sqrt{\frac{d}{c}}\right) \le \frac{k^{2}}{\epsilon^{2}}$$

"Dimensional insensitivity"

# V. Quasi-Random Sequences

These are deterministic sequences u<sub>1</sub>, u<sub>2</sub>, ... in [0, 1]<sup>d</sup> that are "equidistributed":

$$\sup_{a \in [0,1]^d} \left| \frac{1}{n} \sum_{i=1}^n I(u_i \le a) - \prod_{i=1}^d a_i \right| = O\left(\frac{(\log n)^d}{n}\right)$$



$$\left|\frac{1}{n}\sum_{i=1}^{n}\widetilde{f}(u_{i}) - \alpha(\widetilde{f})\right| = O\left(\frac{(\log n)^{d}}{n}\right)$$

if f has finite Hardy-Krause variation

Can be very effective at integration in moderate d settings

# **VI. Output Analysis**

Suppose we have a simulation-based algorithm for computing  $\boldsymbol{\alpha}$ 

How long do we need to run the simulation to get a required accuracy?

**Output Analysis** 

# **Setting 1: IID Replications**

Goal: Compute  $\alpha = E[W]$ 

Method: Generate iid copies  $W_1, \ldots, W_n$  and estimate via  $\alpha_n = \overline{W}_n$ 

Two types of procedures:

► Fixed sample size:

Choose n and construct confidence interval of unknown size

#### Sequential procedures:

Choose error tolerance  $\epsilon$  and generate samples until confidence interval is of required size

- Chow-Robbins (1965)
- G-Whitt (1992)

## **Setting 2: Smooth Functions of Expectations**

- Goal: Compute  $\alpha = g(E[Z])$
- Estimator:  $\alpha_n = g(\overline{Z}_n)$
- Central Limit Theorem:

$$\alpha_n - \alpha = \nabla g(E[Z]) \left(\overline{Z}_n - E[Z]\right) + o_P(n^{-1/2})$$
$$\frac{n^{1/2}(\alpha_n - \alpha)}{s_n} \Rightarrow N(0, 1)$$

as  $n \to \infty$ , where  $s_n^2 \Rightarrow \sigma^2$  and  $\sigma^2 = \text{Var}(\nabla g(E[Z])(Z - E[Z]))$ 

# **Application**

Goal: Compute

$$\alpha = E_x \left[ \sum_{i=0}^{\infty} e^{-\frac{\zeta}{\epsilon}i} r(X_i) \right]$$

Note that

So,

$$\alpha = E_x \left[ \sum_{i=0}^{\tau(x)-1} e^{-\overset{\flat}{\bullet} i} r(X_i) \right] + E_x \left[ e^{-\overset{\flat}{\bullet} \tau(x)} \right] \cdot \alpha$$

$$\alpha = \frac{E_x \left[ \sum_{i=0}^{\tau(x)-1} e^{-\epsilon i} r(X_i) \right]}{1 - E_x \left[ e^{-\epsilon \tau(x)} \right]} = g(E[Z]),$$

where  $g(z_1, z_2) = z_1/(1 - z_2)$ 

#### Setting 3: Steady-State Simulation

Markov chain X = (X<sub>n</sub> : n ≥ 0) with unique equilibrium distribution π(·)
 Goal: Compute

$$\alpha = \int_{S} r(x)\pi(dx) \left(= E[r(X_{\infty})]\right)$$

Estimator:

$$\alpha_n = \frac{1}{n} \sum_{i=0}^{n-1} r(X_i)$$
$$n^{1/2}(\alpha_n - \alpha) \Rightarrow \sigma N(0, 1)$$

where

$$\sigma^2 = \mathsf{Var}_{\pi}(r(X_0)) + 2\sum_{j=1}^{\infty} \mathsf{Cov}_{\pi}\left(r(X_0), r(X_j)\right)$$

The time-average variance constant (TAVC)  $\sigma^2$  is  $2\pi f(0)$ , where  $f(\cdot)$  is the spectral density of X

There are many simulation settings in which the variance is difficult to estimate:

- Smooth functions of expectations
- Steady-state simulation
- Stochastic gradient descent
- Quantiles



# **VII: Replication**

There are many simulation settings in which the variance is difficult to estimate:

- Goal: Compute  $\alpha$
- ▶ Algorithm: An estimator  $\alpha_n$
- A limit theorem:

$$a_n(\alpha_n - \alpha) \Rightarrow \sigma N(0, 1)$$

Now, repeat the algorithm m iid times (m "replications"):  $\alpha_n^1, \alpha_n^2, \ldots, \alpha_n^m$ Note that

$$\alpha_n^i \stackrel{D}{\approx} N(\alpha, \sigma^2/a_n^2)$$

*m* approximately normal rv's with unknown mean and unknown variance
 Confidence interval: Student-t with *m* - 1 degrees of freedom

#### **VIII: Sub-Sampling**

▶ If  $m \ll n$ .

Then.

Suppose that we wish to compute  $\alpha$  using a *Monte Carlo* algorithm  $\alpha_n$  for which

$$n^a(\alpha_n - \alpha) \Rightarrow W$$

where W is a continuous rv. (It can be non-Gaussian and include "nuisance parameters")

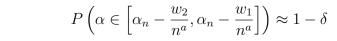
$$m^a(\alpha_m - \alpha) \stackrel{D}{\approx} m^a(\alpha_m - \alpha_n) \stackrel{D}{\approx} W$$

So, construct multiple sub-samples of size m from our n-sample, and use empirical of

$$m^a(\alpha_m^i - \alpha_n), \qquad 1 \le i \le r$$

to estimate  $w_1$ ,  $w_2$  such that

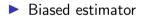
$$P(w_1 \le W \le w_2) \approx 1 - \delta$$



# **IX: Output Analysis in Parallel Computing Context**

- Goal: Compute  $\alpha = E[W]$
- ▶ p parallel processors available
- $\blacktriangleright$  *c* units of compute time
- Run simulations independently on each processor

$$\overline{W}_i(c) = \frac{\sum_{j=1}^{N_i(c)} W_{ij}}{N_i(c)}$$



$$\frac{1}{p}\sum_{i=1}^{p}\overline{W}_{i}(c) \stackrel{D}{\approx} E\left[\overline{W}(c)\right] + \frac{\eta}{\sqrt{pc}}N(0,1)$$

Bias can dominate if p is large G + Heidelberger (1990's)