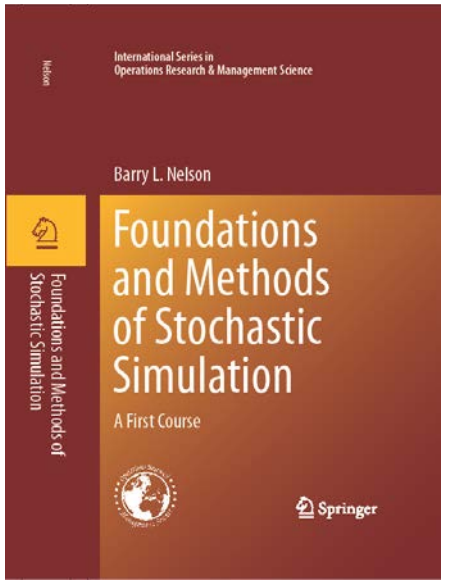


Chapter 8.2, 8.3 & 8.4: Steady-state Simulation Simulation Optimization

©Barry L. Nelson

Northwestern University

February 2015

Steady-state Simulation

Set up

We will restrict attention to the steady-state mean μ (which includes probabilities) and use the expected waiting time of the $M/G/1$ queue as our example.

Will only consider estimators of the form

$$\begin{aligned} \bar{Y}(n,m,d) &= \frac{1}{n}\sum_{j=1}^{n}\bar{Y}_j(m,d) \\ &= \frac{1}{n}\sum_{j=1}^{n}\frac{1}{m-d}\sum_{i=d+1}^{m}Y_{ij} \end{aligned}$$

where m is the run length, d is the data discarded, and n is the number of replications.

Remarks

- This is an experiment design problem: choose *n*, *m* and *d*.
- Considering continuous-time processes (e.g., queue length) introduces no particular difficulties.
  - Run length and deletion amount become *times*.
- We will formulate 2 versions of the "steady-state simulation problem."

Extended asymptotic analysis

$$\mathrm{MSE}\left(\bar{Y}(n,m,d)\right) \approx \frac{\beta(d)^2}{(m-d)^2} + \frac{\gamma^2}{n(m-d)}$$

where

$$\beta(d) = \sum_{i=d+1}^{\infty} \left(\mathrm{E}(Y_i) - \mu\right)$$

which we expect to decrease in absolute value in d.

Example: AR(1) process

$$\mathrm{MSE}\left(\bar{Y}(n,m,d)\right) \approx \frac{(y_0-\mu)^2\varphi^{2d+2}}{(m-d)^2(1-\varphi)^2} + \frac{\sigma^2}{n(m-d)(1-\varphi)^2}$$

Fixed-precision problem

Given $\varepsilon > 0$ or $0 < \kappa < 1$, choose n, m and d so that

$$\sqrt{\mathrm{MSE}\left(\bar{Y}(n, m, d)\right)} \leq \varepsilon$$

or

$$\frac{\sqrt{\mathrm{MSE}\left(\bar{Y}(n, m, d)\right)}}{\mu} \leq \kappa$$

This statement of the problem favors large d to effectively eliminate bias, and $n > 1$ replications to make it easier to quantify error, and makes sense when you can afford to spend as much time as needed.

Fixed-precision approach

1. Determine a run length m and deletion point $\widehat{d}$ such that the bias $\beta(\widehat{d})/(m - \widehat{d})$ is effectively 0.
2. Measure the remaining statistical error via a confidence interval and increase the number of replications until the desired precision has been achieved.

This approach trades efficiency for effectiveness in quantifying error.

Step 1: Estimating the deletion point

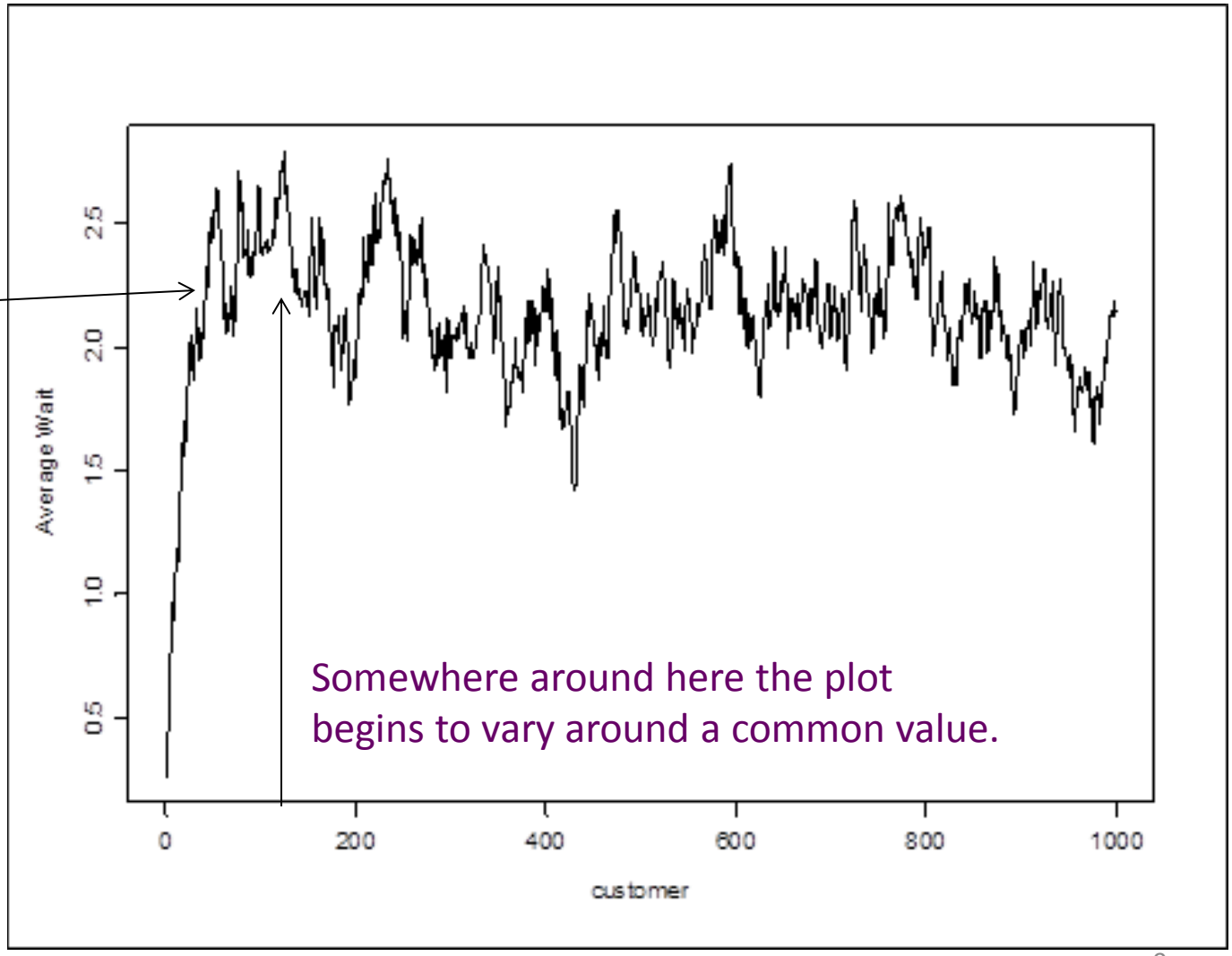
While $\beta(d)/(m-d)$ is not directly estimable, a sufficient condition for $\beta(d)/(m-d) \approx 0$ is that $\mathrm{E}(Y_i) - \mu \approx 0$ for all $i > d$.

This will be true if $\mathrm{E}(Y_i)$ is essentially unchanging for $i > d$.

Therefore, we solve Step 1 by estimating the $\mathrm{E}(Y_i)$ vs. i curve and observing where it appears to stop changing, using replications and smoothing to make the trend apparent.

Mean plot for M/G/1

The *i*th point on the plot is the average across 100 replications of the *i*th customer's waiting time.



Step 2: Fixed precision

Take $m \approx 10\widehat{d}$ (rough rule of thumb).

Form a confidence interval for μ

$$\bar{Y}(n, m, \widehat{d}) \pm t_{1-\alpha/2, n-1} \frac{S(n, m, \widehat{d})}{\sqrt{n}}$$

where

$$S^2(n, m, \widehat{d}) = \frac{1}{n-1} \sum_{j=1}^{n} \left(\bar{Y}_j(m, \widehat{d}) - \bar{Y}(n, m, \widehat{d}) \right)^2.$$

Increase n until you reach the desired precision.

Fixed-budget problem

Given a budget of N observations, solve

$$\min_{n,m,d} \mathrm{MSE}\left(\bar{Y}(n,m,d)\right)$$

$$\begin{aligned} \text{subject to } nm &\leq N \\ d &< m \end{aligned}$$

Recalling that

$$\mathrm{MSE}\left(\bar{Y}(n,m,d)\right) \approx \frac{\beta(d)^2}{(m-d)^2} + \frac{\gamma^2}{N-nd}$$

this formulation favors solutions with $n = 1$ replication and $m = N$.

Deletion with a fixed budget

No deletion: Since the bias component of MSE goes down as $1/m^2$ while variance only diminishes as $1/m$.

Use system knowledge: Use insight about how long it might take the real system to "warm up" to set a deletion point.

Data-driven deletion point: The first two approaches try to eliminate bias, while the third tries to actually minimize the MSE of the resulting estimator.

1. Plot the cumulative sample mean $\sum_{i=1}^{t} Y_i/t$ vs. t, for $t = 1, 2, \ldots, m$.
2. Test for bias: $H_0 : \mathrm{E}(Y_{d+1}) = \cdots = \mathrm{E}(Y_m)$.
3. Estimate the MSE-optimal deletion point.

Marginal standard error (MSER) rule

The $\mathrm{MSER}(d)$ statistic is

$$\mathrm{MSER}(d) = \frac{1}{(m-d)^2} \sum_{i=d+1}^{m} \left(Y_i - \bar{Y}(m,d)\right)^2$$

with estimated deletion point chosen to minimize $\mathrm{MSER}(d)$. Specific rules include

$$\widehat{d} = \mathrm{argmin}_{d=0,\ldots,\lfloor m/2 \rfloor} \mathrm{MSER}(d)$$

and

$$\tilde{d} = \min\left\{d : \mathrm{MSER}(d) \leq \min[\mathrm{MSER}(d-1), \mathrm{MSER}(d+1)]\right\}.$$

Why MSER?

Intuition: Either a strong trend in the mean of the series $Y_1, Y_2, \ldots, Y_m$ (which suggests deletion should be large) or substantial marginal variance $\mathrm{Var}(Y_i)$ (which suggests deletion should be small) will cause MSER to be large.

Thus, minimizing MSER balances these contributions.

Theory: Under very general conditions, for every $d = 0, 1, 2, \ldots$

$$\lim_{m \to \infty} \frac{\mathrm{MSE}(\bar{Y}(m, d))}{\mathrm{E}\left(\mathrm{MSER}(d)\right)} = \text{constant}$$

Thus, the expected value of MSER is proportional to MSE.

Example: AR(1)

For the AR(1) we can derive the expectation of MSER and compare to the actual MSE:

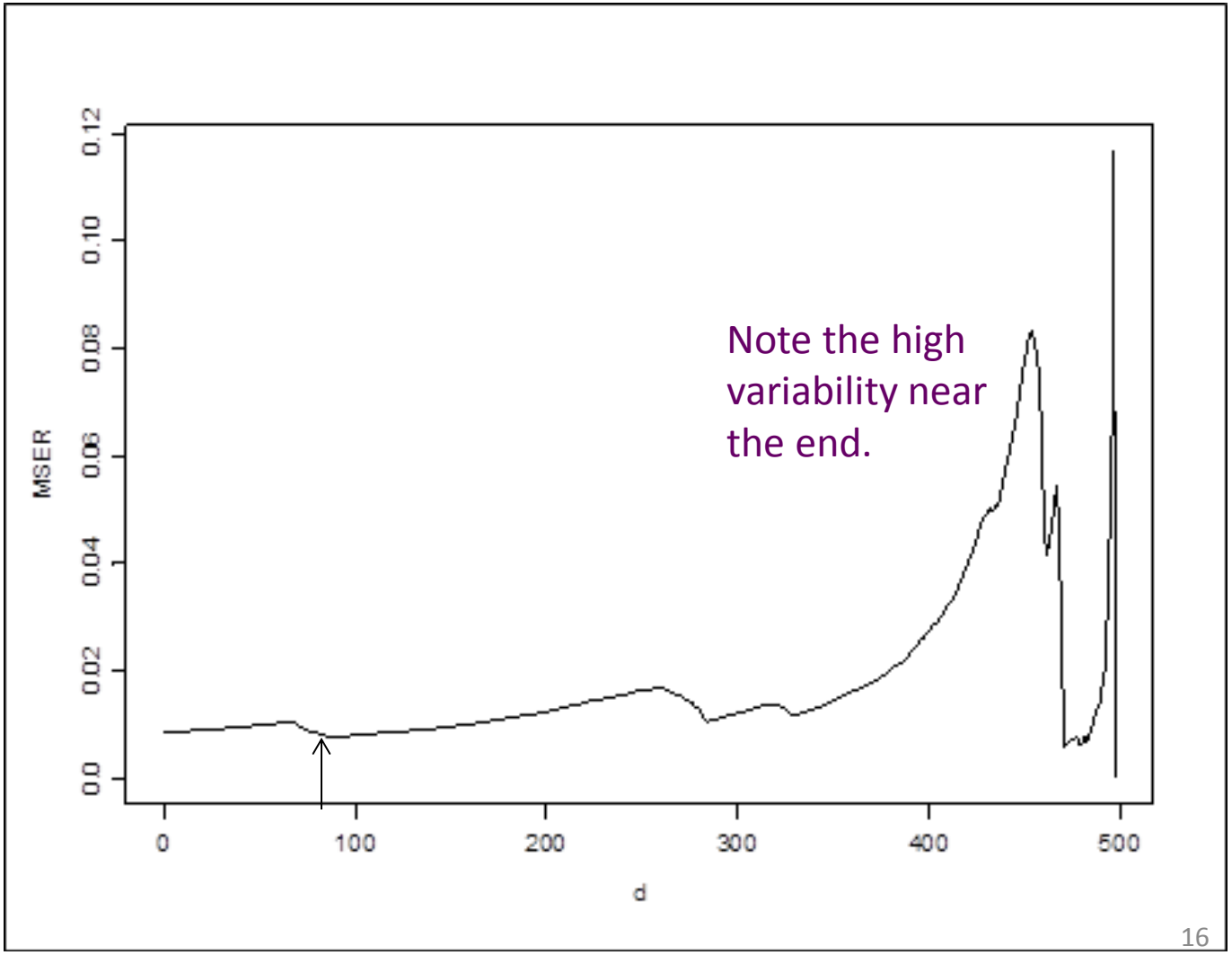
$$\mathrm{E}\left(\mathrm{MSER}(d)\right) \approx \frac{(1-\varphi)^2}{(1-\varphi^2)}\left(\frac{(y_0-\mu)^2\varphi^{2d+2}}{(m-d)^2(1-\varphi)^2} + \frac{\sigma^2}{n(m-d)(1-\varphi)^2}\right)$$

$$\mathrm{MSE}\left(\bar{Y}(n,m,d)\right) \approx \frac{(y_0-\mu)^2\varphi^{2d+2}}{(m-d)^2(1-\varphi)^2} + \frac{\sigma^2}{n(m-d)(1-\varphi)^2}$$

MSER plot for M/G/1

MSER(d) from one replication of m = 500 waiting times.

Both rules give d = 88 in this case.



Calculating MSER

$$\sum_{i=d+1}^{m} \left(Y_i - \bar{Y}(m,d)\right)^2 = \sum_{i=d+1}^{m} Y_i^2 - \frac{1}{m-d}\left(\sum_{i=d+1}^{m} Y_i\right)^2$$

Thus, MSER can be computed in one pass through the data by starting from the end and working backward:

1. Set $s = 0, q = 0$
2. For $d = m - 1$ to 0
   (a) $s = s + Y_{d+1}$
   (b) $q = q + Y_{d+1}^2$
   (c) $\mathrm{MSER}(d) = \left(q - s^2/(m-d)\right)/(m-d)^2$
3. Next d

Error estimation with a fixed budget

We argued earlier that for m large $\mathrm{Var}\left(\bar{Y}(m)\right) \approx \gamma^2/m$.

For $b < m$, but still large, $\mathrm{Var}\left(\bar{Y}(b)\right) \approx \gamma^2/b$ as well. Therefore, if both m and b are large,

$$\mathrm{Var}\left(\bar{Y}(m)\right) \approx \frac{b}{m}\mathrm{Var}(\bar{Y}(b))$$

and we can estimate $\mathrm{Var}\left(\bar{Y}(b)\right)$ by forming $k = m/b$ batch means

$$\underbrace{Y, \ldots, Y}_{\text{deleted}}, \underbrace{Y_1, \ldots, Y_b}_{\bar{Y}_1(b)}, \underbrace{Y_{b+1}, \ldots, Y_{2b}}_{\bar{Y}_2(b)}, \ldots, \underbrace{Y_{(k-1)b+1}, \ldots, Y_{kb}}_{\bar{Y}_k(b)}.$$

Batch means variance estimator

The natural estimator of $\mathrm{Var}\left(\bar{Y}(b)\right)$ is

$$S^2(k) = \frac{1}{k-1}\sum_{j=1}^{k}\left(\bar{Y}_j(b) - \bar{Y}(m)\right)^2.$$

This yields $\widehat{\mathrm{Var}}\left(\bar{Y}(m)\right) = (b/m)S^2(k)$ or $\widehat{\gamma}^2 = bS^2(k)$.

An approximate confidence interval for μ is

$$\bar{Y}(m) \pm t_{1-\alpha/2,k-1}\frac{\widehat{\gamma}}{\sqrt{m}}$$

which is algebraically equivalent to

$$\bar{Y}(m) \pm t_{1-\alpha/2,k-1}\frac{S(k)}{\sqrt{k}}.$$

When will this "work"?

We need $b \geq b^\star$ (equivalently $k \leq k^\star$) where

$$\sigma^2 \left(1 + 2\sum_{i=1}^{b^\star} \rho_i\right) \approx \gamma^2 = \sigma^2 \left(1 + 2\sum_{i=1}^{\infty} \rho_i\right)$$

so that a batch effectively captures the signficant autocorrelations.

A sufficient condition is that the batch means

$$\bar{Y}_1(b), \bar{Y}_2(b), \ldots, \bar{Y}_k(b)$$

are approximately independent; many batching algorithms have been based on this idea.

Simplest (sensible) algorithm

1. Make a single replication of length $m = N$.
2. Apply MSER to obtain a deletion point d (or for an even simpler procedure, set $d = 0$).
3. Divide the remaining $m - d$ observations into from $10 \leq k \leq 30$ batches, looking for a value of k that divides $m - d$ close to evenly (if there are data left over, delete from the beginning).
4. Compute the sample mean and form the batch means confidence interval.

Why "$10 \leq k \leq 30$"?

Remember that the amount of data is fixed. The bigger the number of batches k (smaller the batch size b) the greater the risk that we do not cover the correlation structure.

The benefit from additional dof diminishes rapidly.

k	$\frac{t_{0.975,k-1}}{z_{0.975}}$
2	6.48
5	1.42
10	1.15
20	1.07
30	1.04
60	1.02
∞	1.00

Summary

1. There is a tradeoff between bias and variance, and both matter, as reflected in the MSE.
2. Bias diminishes faster than variance with increasing run length, but does not diminish at all with increasing numbers of replications. Thus, no solution will have a very large number of short replications.
3. How you attack the steady-state simulation problem depends on whether you can afford to treat it as a fixed-precision or a fixed-budget problem. When data are cheap (i.e., fast) a fixed-precision approach may waste data but it gives more assurance that you get what you want.

Simulation Optimization

Simulation optimization formulation

$$\begin{aligned} &\min \ \theta(\mathbf{x}) \\ &\mathbf{x} \in \mathsf{C} \end{aligned}$$

where $\theta(\mathbf{x})$ is the performance measure of interest, and C is the feasible set or region for the d-dimensional scenario variable $\mathbf{x}$.

What makes this a *simulation optimization* (SO) problem is the need to estimate the scenario performance measure $\theta(\mathbf{x})$ using a simulation-based estimator $\widehat{\theta}(\mathbf{x}; T, n, \mathbf{U})$.

We may also have to estimate whether some of the constraints are satisfied (later).

Experiment design for SO

$$\widehat{\theta}(\mathbf{x}; T, n, \mathbf{U})$$

$\mathbf{x}$ defines the scenario, and may be categorical, discrete-valued or continuous-valued.

T, n are the run length and number of replications (only one may be relevant).

$\mathbf{U}$ are the underlying (pseudo)random numbers, which in simulation can be *assigned*.

In SO there is always a fundamental tradeoff between search and selection (exploration and exploitation).

Example: SAN resource allocation

Suppose activity j is exponentially distributed with mean τ_j, but we can reduce it to x_j at a cost of $c_j(\tau_j - x_j)$.

$$\begin{aligned} &\min\ \theta(x_1, x_2, \ldots, x_5) \\ &\sum_{j=1}^{5} c_j(\tau_j - x_j) \leq b \\ &x_j \geq \ell_j,\ j = 1, 2, 3, 4, 5 \end{aligned}$$

where $\theta(\mathbf{x}) = \mathrm{E}(Y(\mathbf{x}))$ and

$$\begin{aligned} Y(\mathbf{x}) \quad = \quad & \max\{A_1(x_1) + A_4(x_4), \\ & A_1(x_1) + A_3(x_3) + A_5(x_5), A_2(x_2) + A_5(x_5)\} \end{aligned}$$

The natural estimator for scenario $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{i5})$ is

$$\widehat{\theta}(\mathbf{x}_i) = \frac{1}{n_i} \sum_{j=1}^{n_i} Y_j(\mathbf{x}_i)$$

where $Y_1(\mathbf{x}_i), Y_2(\mathbf{x}_i), \ldots$ are i.i.d. replications of $\mathbf{x}_i$.

- If we reduce the mean of activity i by allocating at most one extra worker then we can simulate all scenarios.
- If the budget is very large relative to the cost of an individual worker, there may be so many scenarios we cannot simulate them all.
- If the reduction in mean activity time comes from allocating some sort of capacity or power, then we might be able to treat $\mathbf{x}_i$ as continuous valued.

Errors in SO

Whatever the SO algorithm is, it will terminate having simulated $K < \infty$ scenarios. The errors are...

1. *The optimal scenario is never simulated*, meaning

$$\mathbf{x}^\star \notin \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K\}.$$

2. *The best scenario that was simulated is not selected*, meaning

$$\widehat{\mathbf{x}}^\star \neq \mathbf{x}_B = \text{argmin}_{i=1,2,\ldots,K}\, \theta(\mathbf{x}_i).$$

3. *The estimated objective function value of the selected scenario is not very precise*, meaning

$$\left|\widehat{\theta}(\widehat{\mathbf{x}}^\star) - \theta(\widehat{\mathbf{x}}^\star)\right| \text{ is large.}$$

Solution for 1: Convergence

Globally convergent: If $\widehat{\mathbf{x}}_r^\star$ is the estimated optimal on iteration r then we want

$$\Pr\left\{\lim_{r\to\infty} \widehat{\mathbf{x}}_r^\star = \mathbf{x}^\star\right\} = 1.$$

Locally convergent: $\mathbf{x}'$ is *locally optimal* if

$$\theta(\mathbf{x}') \leq \theta(\mathbf{x}) \text{ for all } \mathbf{x} \in \mathsf{N}(\mathbf{x}').$$

If L is the set of locally optimal solutions, then we want

$$\Pr\left\{\widehat{\mathbf{x}}_r^\star \notin \mathsf{L} \text{ infinitely often}\right\} = 0.$$

Solution for 2: Correct selection

Consider the correct-selection event

$$\begin{aligned} \text{CS} &= \{\text{select } \mathbf{x}_B\} \\ &= \left\{\widehat{\theta}(\mathbf{x}_B) < \widehat{\theta}(\mathbf{x}_i), i = 1, 2, \ldots, K,\ i \neq B\right\} \\ &= \bigcap_{i=1, i\neq B}^{K} \left\{\widehat{\theta}(\mathbf{x}_B) < \widehat{\theta}(\mathbf{x}_i)\right\}. \end{aligned}$$

We want a guarantee that

$$\Pr\{\text{CS}\} \geq 1 - \alpha.$$

Solution for 3: Clean up

We might be willing to spend some additional simulation effort after the SO algorithm terminates to guarantee that

$$\theta(\widehat{\mathbf{x}}^\star) \in \widehat{\theta}(\widehat{\mathbf{x}}^\star) \pm \delta.$$

That is, the estimated value of the selected solution is within $\pm\delta$ of its true value.

We can often do this in conjunction with correct selection.

Random number assignment and correct selection

Suppose the $\widehat{\theta}$'s are approximately normally distributed. What is the probability we make an error?

$$\Pr\left\{\widehat{\theta}(\mathbf{x}_B) \geq \widehat{\theta}(\mathbf{x}_i)\right\} = \Pr\left\{\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) \geq 0\right\}$$

$$= \Pr\left\{\frac{\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) - (\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i))}{\sqrt{\mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i)\right]}} \geq \frac{-(\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i))}{\sqrt{\mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i)\right]}}\right\}$$

$$= \Pr\left\{Z \geq \frac{\theta(\mathbf{x}_i) - \theta(\mathbf{x}_B)}{\sqrt{\mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i)\right]}}\right\}$$

Common random numbers

The variance of the difference is

$$\begin{aligned}&\mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i)\right]\\ &= \mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_B)\right] + \mathrm{Var}\left[\widehat{\theta}(\mathbf{x}_i)\right] - 2\mathrm{Cov}\left[\widehat{\theta}(\mathbf{x}_B), \widehat{\theta}(\mathbf{x}_i)\right].\end{aligned}$$

If each scenario $\mathbf{x}_i$ is assigned independent random numbers $\mathbf{U}_i$, then the estimators are independent and $\mathrm{Cov}\left[\widehat{\theta}(\mathbf{x}_i;\mathbf{U}_i), \widehat{\theta}(\mathbf{x}_B;\mathbf{U}_B)\right] = 0.$

Common random numbers means letting $\mathbf{U}_i = \mathbf{U}_B = \mathbf{U}$, which tends to make $\mathrm{Cov}\left[\widehat{\theta}(\mathbf{x}_i,\mathbf{U}), \widehat{\theta}(\mathbf{x}_B,\mathbf{U})\right] > 0$ thus *reducing the probability of error*.

CRN: Intuition from SAN

Suppose $x_{i1} = x_{i2} = \cdots = x_{i5} = x_i$, and $x_{h1} = x_{h2} = \cdots = x_{h5} = x_h$. Then

$$\begin{aligned} Y(\mathbf{x}_h) &= \max\{-\ln(1-U_1)x_h - \ln(1-U_4)x_h, \\ &\qquad -\ln(1-U_1)x_h - \ln(1-U_3)x_h - \ln(1-U_5)x_h, \\ &\qquad -\ln(1-U_2)x_h - \ln(1-U_5)x_h\} \\ &= \frac{x_h}{x_i}\max\{-\ln(1-U_1)x_i - \ln(1-U_4)x_i, \\ &\qquad -\ln(1-U_1)x_i - \ln(1-U_3)x_i - \ln(1-U_5)x_i, \\ &\qquad -\ln(1-U_2)x_i - \ln(1-U_5)x_i\} \\ &= \frac{x_h}{x_i} Y(\mathbf{x}_i). \end{aligned}$$

$$\Rightarrow \mathrm{Cov}[Y(\mathbf{x}_i), Y(\mathbf{x}_h)] = \frac{x_h}{x_i}\mathrm{Var}[Y(\mathbf{x}_i)] > 0.$$

Lessons learned

- Two aspects of this example are important more generally:
  1. **Monotonicity** of random numbers → inputs → outputs.
     - The inverse cdf method helps here.
  2. **Synchronization** of how the random numbers are used by each scenario.
     - Random number "streams" help here, but may not be enough.

Design & analysis for correct selection

Suppose there are only K scenarios $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K$, we would like a guarantee of selecting the best among them and there is time to simulate them all.

This is the domain of *ranking & selection* which has been quite successful in simulation applications and software.

Small-sample validity usually depends on i.i.d. normality; there are asymptotic justifications when normality and independence do not apply.

The guarantees

$$\Pr\{\mathbf{x}_B \in I\} \geq 1 - \alpha$$
$$\Pr\{\text{select } \mathbf{x}_B | \theta(\mathbf{x}_i) - \theta(\mathbf{x}_B) \geq \delta, \forall i \neq B\} \geq 1 - \alpha.$$

1. Given $n_i \geq 2$ observations from scenario $\mathbf{x}_i$, set

$$t_i = t_{(1-\alpha)^{\frac{1}{K-1}},\ n_i-1}$$

the $(1-\alpha)^{\frac{1}{K-1}}$ quantile of the t distribution with $n_i - 1$ degrees of freedom, for $i = 1, 2, \ldots, K$.

2. Calculate the sample means $\bar{Y}(\mathbf{x}_i; n_i)$ and sample variances

$$S^2(\mathbf{x}_i) = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} \left(Y_j(\mathbf{x}_i) - \bar{Y}(\mathbf{x}_i; n_i) \right)^2$$

for $i = 1, 2, \ldots, K$, and also the threshold

$$W_{ih} = \left(t_i^2 \frac{S^2(\mathbf{x}_i)}{n_i} + t_h^2 \frac{S^2(\mathbf{x}_h)}{n_h} \right)^{1/2}$$

for all $i \neq h$.

3. Form the subset

$$I = \left\{ \mathbf{x}_i : \bar{Y}(\mathbf{x}_i; n_i) \leq \bar{Y}(\mathbf{x}_h; n_i) + W_{ih} \text{ for all } h \neq i \right\}.$$

Foundation

The following is behind many subset selection procedures:

$$
\begin{aligned}
&\Pr\{\mathbf{x}_B \in I\} \\
&\quad = \Pr\left\{\bar{Y}(\mathbf{x}_B; n_B) \leq \bar{Y}(\mathbf{x}_h; n_h) + W_{Bh},\ h \neq B\right\} \\
&\quad = \Pr\left\{\bar{Y}(\mathbf{x}_B; n_B) - \bar{Y}(\mathbf{x}_h; n_h) - [\theta(\mathbf{x}_B) - \theta(\mathbf{x}_h)] \leq \right. \\
&\qquad\qquad \left. W_{Bh} - [\theta(\mathbf{x}_B) - \theta(\mathbf{x}_h)],\ h \neq B\right\} \\
&\quad \geq \Pr\left\{\bar{Y}(\mathbf{x}_B; n_B) - \bar{Y}(\mathbf{x}_h; n_h) - [\theta(\mathbf{x}_B) - \theta(\mathbf{x}_h)] \leq W_{Bh},\ h \neq B\right\}.
\end{aligned}
$$

The statistic

$$\bar{Y}(\mathbf{x}_i; n_i) - \bar{Y}(\mathbf{x}_h; n_h) - [\theta(\mathbf{x}_i) - \theta(\mathbf{x}_h)]$$

has mean 0 for all $i \neq h$, allowing the W_{ih}'s to be derived that give the desired probability based only on their variances.

1. Choose $n_0 \geq 2$. Set

$$\eta = \frac{1}{2}\left[\left(\frac{2\alpha}{K-1}\right)^{-2/(n_0-1)} - 1\right].$$

2. Let $I = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K\}$, $t^2 = 2\eta(n_0 - 1)$. Obtain $Y_j(\mathbf{x}_i), j = 1, 2, \ldots, n_0$ for $\mathbf{x}_i \in I$, compute $\bar{Y}(\mathbf{x}_i; n_0)$ and for all $i \neq h$

$$S_{ih}^2 = \frac{1}{n_0 - 1}\sum_{j=1}^{n_0}\left(Y_j(\mathbf{x}_i) - Y_j(\mathbf{x}_h) - \left[\bar{Y}(\mathbf{x}_i; n_0) - \bar{Y}(\mathbf{x}_h; n_0)\right]\right)^2.$$

Set $r = n_0$.

3. Set $I^{\text{old}} = I$. Let

$$I = \left\{\mathbf{x}_i : \mathbf{x}_i \in I^{\text{old}} \text{ and } \bar{Y}(\mathbf{x}_i; r) \leq \bar{Y}(\mathbf{x}_h; r) + W_{ih}(r), \forall h \in I^{\text{old}}, h \neq i\right\},$$

where

$$W_{ih}(r) = \max\left\{0, \frac{\delta}{2r}\left(\frac{t^2 S_{ih}^2}{\delta^2} - r\right)\right\}.$$

4. If $|I| = 1$ stop and select that scenario. Else simulate $Y_{r+1}(\mathbf{x}_i)$ for $\mathbf{x}_i \in I$, update sample means, set $r = r + 1$ and go to Step 3.

Foundation

Since we assume $\theta(\mathbf{x}_i) - \theta(\mathbf{x}_B) \geq \delta$

$$
\begin{aligned}
&\Pr\left\{\widehat{\theta}(\mathbf{x}_B) < \widehat{\theta}(\mathbf{x}_i)\right\} \\
&= \Pr\left\{\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) < 0\right\} \\
&= \Pr\left\{\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) - [\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i)] < -[\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i)]\right\} \\
&\geq \Pr\left\{\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) - [\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i)] \leq \delta\right\}.
\end{aligned}
$$

The statistic

$$\widehat{\theta}(\mathbf{x}_B) - \widehat{\theta}(\mathbf{x}_i) - (\theta(\mathbf{x}_B) - \theta(\mathbf{x}_i))$$

has mean 0 so the procedure can be designed to provide the desired guarantee considering only δ, the variances and the number of replications.

Remarks

- This subset procedure does not exploit CRN, but can be made to do so if equal n_i.
- The selection procedure (KN) exploits CRN (note the use of the variance of the difference).
- KN is efficient because it quickly eliminates inferior scenarios, but is sequential and coordinated.
- These procedures can be used to "clean up" after a search to insure that the best of those simulated is selected and estimated well.

Adaptive random search (ARS)

ARS is the (convergent) standard used when C is discrete, but too large to simulate everything.

On the ith iteration, ARS algorithms have a distribution $P_i(\mathbf{x})$ on the scenarios $\mathbf{x} \in$ C used to *sample* one or more scenarios from C to *simulate*.

The distribution $P_i(\cdot)$ may depend on its "memory" of some or all of the scenarios that have been simulated and perhaps the output data obtained from simulating them.

There will also be an estimation set to *simulate*, and a *value* $V(\mathbf{x})$ for each simulated scenario.

Generic ARS algorithm

Initialize: Set the iteration counter to $i = 1$.

Sample: Choose an estimation set, which is a collection scenarios $\mathsf{E}_i \subset \mathsf{C}$ where some or all of the scenarios were chosen by sampling according to $P_i(\cdot)$ and others were retained from previous iterations.

Simulate: Apply the simulation allocation rule to simulate the scenarios $\mathbf{x} \in \mathsf{E}_i$.

Evaluate: Update the value $V(\mathbf{x})$ for all $\mathbf{x} \in \mathsf{E}_i$ and choose as $\widehat{\mathbf{x}}^{\star}_{i+1}$ the scenario with the best value $V(\mathbf{x})$.

Iterate: Update the algorithm memory, let $i = i + 1$ and go to **Sample**.

Common sampling distributions

- A distribution that puts positive probability on a small number of feasible scenarios in a neighborhood of $\widehat{\mathbf{x}}_i^\star$. Typically, $P_i(\cdot)$ and the neighborhood structure connect C.
- A distribution that puts positive probability on a "promising" subset of C that may be large or small, but is not necessarily a neighborhood of $\widehat{\mathbf{x}}_i^\star$. Typically such distributions use memory in an intelligent way to concentrate the search.
- A distribution that puts positive probability on all of C. Typically the distribution changes as a function of the iteration and the memory, focusing probabilistically on promising regions of C.

Example: Adaptive hyperbox algorithm

Value: $\widehat{\theta}(\mathbf{x})$ accumulated over all output data generated for scenario $\mathbf{x}$ (typically sample mean).

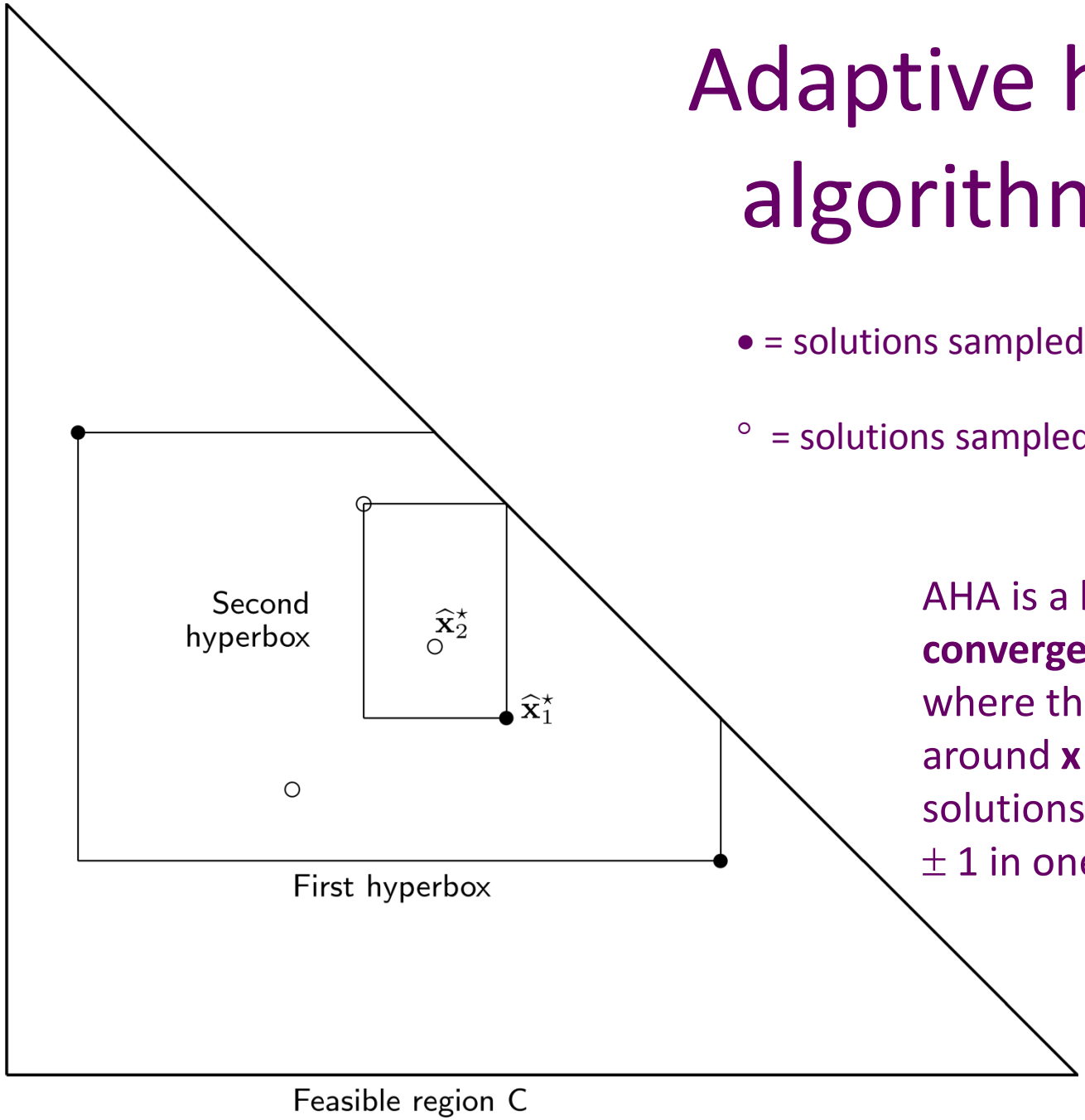
Memory: All scenarios that have been simulated and their estimated objective function values.

Distribution $P_i(\cdot)$**:** Positive probability on scenarios that are feasible and are in or on a hyperbox that surrounds $\widehat{\mathbf{x}}_i^\star$. The hyperbox is defined by scenarios that have been simulated and are closest to $\widehat{\mathbf{x}}_i^\star$ in one or more coordinate.

Estimation set E_i**:** The current sample best $\widehat{\mathbf{x}}_i^\star$ and any scenarios sampled from $P_i(\cdot)$.

Simulation allocation rule: Whenever a scenario is in the estimation set it must receive additional simulation.

Adaptive hyperbox algorithm (AHA)



- • = solutions sampled in first iteration
- ◦ = solutions sampled in second iteration

AHA is a **locally convergent** algorithm where the neighborhood around **x** contains the solutions that differ by $\pm$ 1 in one coordinate.

Moving in improving directions

Suppose that $\theta(\mathbf{x})$ is continuous and differentiable in $\mathbf{x}$ and C is convex. Then we might consider a gradient-based search such as *stochastic approximation*.

Starting from $\mathbf{x}_0$, implement the recursion

$$\mathbf{x}_{i+1} = \mathbf{x}_i - a_i \widehat{\nabla}\theta(\mathbf{x}_i)$$

where a_i is a sequence such that $a_i \to 0$, but $\sum_{i=1}^{\infty} a_i = \infty$.

The intuition is that when $\nabla\theta(\mathbf{x}_i) = 0$ a stationary point has been reached while having $a_i \to 0$ (but not too fast) mutes the impact of variability in $\widehat{\nabla}\theta$.

This motivates the need for gradient estimators.

SAN example

Recall that the basic output can be expressed in two ways:

$$\begin{aligned} Y(\mathbf{x}) &= \max\{A_1(x_1) + A_4(x_4), \\ &\qquad A_1(x_1) + A_3(x_3) + A_5(x_5), \\ &\qquad A_2(x_2) + A_5(x_5)\} \\ &= \max\{-\ln(1-U_1)x_1 - \ln(1-U_4)x_4, \\ &\qquad -\ln(1-U_1)x_1 - \ln(1-U_3)x_3 - \ln(1-U_5)x_5, \\ &\qquad -\ln(1-U_2)x_2 - \ln(1-U_5)x_5\} \end{aligned}$$

and $\theta(\mathbf{x}) = \mathrm{E}\,(Y(\mathbf{x}))$.

We will illustrate the key ideas by estimating $\partial\theta(\mathbf{x})/\partial x_1$.

Finite difference

The definition of derivative gives a natural approximation.

$$\begin{aligned} Y(\mathbf{x}+\Delta x_1) \;=\; \max\{&-\ln(1-U_1)(x_1+\Delta x_1)-\ln(1-U_4)x_4,\\ &-\ln(1-U_1)(x_1+\Delta x_1)-\ln(1-U_3)x_3\\ &-\ln(1-U_5)x_5,\\ &-\ln(1-U_2)x_2-\ln(1-U_5)x_5\} \end{aligned}$$

The term Δx_1 is used to denote the vector $(\Delta x_1, 0, 0, \ldots, 0)^\top$.

$$\mathrm{FD}(x_1) = \frac{Y(\mathbf{x}+\Delta x_1) - Y(\mathbf{x})}{\Delta x_1}.$$

Using CRN helps, but we have a bias-variance trade-off in Δx_1 and it requires $(d+1)\times n$ simulations to estimate the full gradient.

Infinitesimal perturbation analysis

Suppose that we fix the pseudorandom numbers. If $A(x_1)$ is on the longest path, then it will also be on the longest path at $\mathbf{x}+\Delta x_1$ for Δx_1 small enough.

$$\lim_{\Delta x_1 \to 0} \frac{Y(\mathbf{x}+\Delta x_1) - Y(\mathbf{x})}{\Delta x_1}$$

$$= \lim_{\Delta x_1 \to 0} \frac{-\ln(1-u_1)(x_1+\Delta x_1) - (-\ln(1-u_1)x_1)}{\Delta x_1}$$

$$= -\ln(1-u_1) = \frac{-\ln(1-u_1)x_1}{x_1} = \frac{A_1(x_1)}{x_1}\,.$$

If it is not on the longest path, then

$$\lim_{\Delta x_1 \to 0} \frac{Y(\mathbf{x}+\Delta x_1) - Y(\mathbf{x})}{\Delta x_1} = 0.$$

IPA

$$\text{IPA}(x_1) = \frac{A_1(x_1)}{x_1} I\{A_1(x_1) \text{ is on the longest path}\}$$

IPA gradient estimators as sometimes called "sample-path derivatives" because we (literally) take the derivative of the output.

The validity of IPA gradient estimation hinges on the validity of an interchange of differentiation and expectation:

$$\frac{\partial \theta(\mathbf{x})}{\partial x_i} = \frac{\partial \mathrm{E}[Y(\mathbf{x})]}{\partial x_i} \stackrel{?}{=} \mathrm{E}\left[\frac{\partial Y(\mathbf{x})}{\partial x_i}\right].$$

Conditions are easy to state (mathematically) but difficult to verify for simulations.

Likelihood ratio method

Let $g(\mathbf{a}) = \max\{a_1 + a_4, a_1 + a_3 + a_5, a_2 + a_5\}$.

$$
\begin{aligned}
& \mathrm{E}\left[Y(\mathbf{x})\frac{f_1(A_1|x_1+\Delta x_1)}{f_1(A_1|x_1)}\right] \\
&= \mathrm{E}\left[Y(\mathbf{x})\frac{\exp\{-A_1/(x_1+\Delta x_1)\}/(x_1+\Delta x_1)}{\exp\{-A_1/x_1\}/x_1}\right] \\
&= \int_0^\infty \cdots \int_0^\infty g(\mathbf{a})\frac{f_1(a_1|x_1+\Delta x_1)}{f_1(a_1|x_1)}\prod_{j=1}^{5} f_j(a_j|x_j)\,d\mathbf{a} \\
&= \int_0^\infty \cdots \int_0^\infty g(\mathbf{a}) f_1(a_1|x_1+\Delta x_1)\prod_{j=2}^{5} f_j(a_j|x_j)\,d\mathbf{a} \\
&= \mathrm{E}[Y(\mathbf{x}+\Delta x_1)] = \theta(\mathbf{x}+\Delta x_1).
\end{aligned}
$$

LR

Therefore, from a single simulation replication we can get a finite-difference estimator:

$$\frac{Y(\mathbf{x})\dfrac{f_1(A_1|x_1+\Delta x_1)}{f_1(A_1|x_1)}-Y(\mathbf{x})}{\Delta x_1}$$
$$= \frac{Y(\mathbf{x})}{f_1(A_1|x_1)}\times\frac{f_1(A_1|x_1+\Delta x_1)-f_1(A_1|x_1)}{\Delta x_1}.$$

As we did with FD to motivate IPA, we can now take the $\lim_{\Delta x_1\to 0}$

$$\mathrm{LR}(x_1)=\frac{Y(\mathbf{x})}{f_1(A_1|x_1)}\frac{\partial f_1(A_1|x_1)}{\partial x_1}=Y(\mathbf{x})\frac{\partial \ln f_1(A_1|x_1)}{\partial x_1}.$$

SAN example

For the exponential distribution of activity 1

$$\frac{\partial \ln f_1(A_1|x_1)}{\partial x_1} = \frac{\partial}{\partial x_1}\left(-\frac{A_1}{x_1} - \ln x_1\right) = \frac{A_1}{x_1^2} - \frac{1}{x_1} = \frac{A_1 - x_1}{x_1^2}.$$

Therefore

$$\mathrm{LR}(x_1) = Y(\mathbf{x})\left(\frac{A_1 - x_1}{x_1^2}\right).$$

To estimate the gradient we average over $n > 1$ replications.

Again there are technical conditions to satisfy.

For LR to be applicable x must be the parameter of an input distribution.

Choice of gradient estimator

1. FD is always available.

   Using central differences reduces bias at the cost of $2d \times n$ simulations.

   Good if d is small.

2. IPA provides low-variance gradient estimators, but is often not viable and requires more programming.

3. LR conditions are easier to verify, but the gradient estimators are more variable especially when when x is the parameter of more than one random variate per replication because the weight is a *product* of the input distribution for each instance.

Sample average approximation

We have been solving

$$\begin{aligned} &\min \ \theta(\mathbf{x}) \\ &\mathbf{x} \in \mathsf{C} \end{aligned}$$

using estimator $\widehat{\theta}(\mathbf{x}; n, \mathbf{U})$ of the objective function.

Consider instead solving

$$\begin{aligned} &\min \ \widehat{\theta}(\mathbf{x}; n, \mathbf{U} = \mathbf{u}) \\ &\mathbf{x} \in \mathsf{C} \end{aligned}$$

after the random stuff has been generated. This is now a *deterministic* optimization problem.

SAN example

The output from replication j of the SAN simulation can be expressed as

$$\begin{aligned} y_j(\mathbf{x}) &= \max\{a_{1j}(x_1) + a_{4j}(x_4), \\ &\qquad a_{1j}(x_1) + a_{3j}(x_3) + a_{5j}(x_5), \\ &\qquad a_{2j}(x_2) + a_{5j}(x_5)\} \\ &= \max\{-\ln(1-u_{1j})x_1 - \ln(1-u_{4j})x_4, \\ &\qquad -\ln(1-u_{1j})x_1 - \ln(1-u_{3j})x_3 - \ln(1-u_{5j})x_5, \\ &\qquad -\ln(1-u_{2j})x_2 - \ln(1-u_{5j})x_5\} \end{aligned}$$

The objective of the corresponding SAA problem is

$$\min \ \widehat{\theta}(\mathbf{x}; n, \mathbf{U} = \mathbf{u}) = \min \frac{1}{n} \sum_{j=1}^{n} y_j(\mathbf{x}).$$

If we replace the $\max$ operator by three inequality constraints then the SAA problem can be formulated as a linear program:

$$
\begin{aligned}
\min \quad & \frac{1}{n}\sum_{j=1}^{n} y_j \\
y_j \;&\geq\; -\ln(1-u_{1j})x_1 - \ln(1-u_{4j})x_4 \\
y_j \;&\geq\; -\ln(1-u_{1j})x_1 - \ln(1-u_{3j})x_3 - \ln(1-u_{5j})x_5 \\
y_j \;&\geq\; -\ln(1-u_{2j})x_2 - \ln(1-u_{5j})x_5, \;\; j = 1, 2, \ldots, n \\
b \;&\geq\; \sum_{k=1}^{5} c_k(\tau_k - x_k) \\
x_k \;&\geq\; \ell_k, \;\; k = 1, 2, \ldots, 5.
\end{aligned}
$$

This linear program has $n+5$ decision variables $y_1, y_2, \ldots, y_n$, $x_1, x_2, \ldots, x_5$ and $3n + 6$ constraints.

Uniform convergence

What we hope is that $\theta(\widehat{\mathbf{x}}_n^\star) \to \theta(\mathbf{x}^\star)$, but pointwise convergence is not enough since $\widehat{\mathbf{x}}_n^\star$ is not a fixed value of $\mathbf{x}$ but rather the solution to the SAA problem over *all* $\mathbf{x} \in \mathsf{C}$.

Uniform convergence means that, for any $\epsilon > 0$, with probability 1 there exists an n' such that

$$\sup_{\mathbf{x} \in \mathsf{C}} \left| \widehat{\theta}(\mathbf{x}; n; \mathbf{U}) - \theta(\mathbf{x}) \right| \leq \epsilon$$

for all $n \geq n'$. This is the essential condition.

Stochastic Constraints

Extended SO formulation:

$$\begin{aligned} &\min\ \theta(\mathbf{x}) \\ &\mathbf{x} \in \mathsf{C} \\ &c_\ell(\mathbf{x}) \geq q_\ell,\ \ell = 1, 2, \ldots, v \end{aligned}$$

where $\theta(\mathbf{x}), c_1(\mathbf{x}), c_2(\mathbf{x}), \ldots, c_v(\mathbf{x})$ must all be estimated.

Assume that for any scenario $\mathbf{x}$ we can observe simulation outputs $Y(\mathbf{x}), C^{(1)}(\mathbf{x}), C^{(2)}(\mathbf{x}), \ldots, C^{(v)}(\mathbf{x})$ such that

$$\begin{aligned} \mathrm{E}\left(Y(\mathbf{x})\right) &= \theta(\mathbf{x}) \\ \mathrm{E}\left(C^{(\ell)}(\mathbf{x})\right) &= c_\ell(\mathbf{x}),\ \ell = 1, 2, \ldots, v \end{aligned}$$

Examples

Call center: $\mathbf{x}$ is staffing schedule

$\theta(\mathbf{x})$ staffing cost

$c(\mathbf{x})$ is the fraction of calls answered within 2 minutes

$q = 0.92$

Manufacturing: $\mathbf{x}$ determines a plant layout and machine capacities

$\theta(\mathbf{x})$ work in process

$c(\mathbf{x})$ is throughput

$q = 5000$ parts/day

Feasibility checking

Suppose we just want to check if $\mathbf{x}$ is feasible using $\bar{C}(\mathbf{x})$.

$$\begin{aligned}\Pr\{\bar{C}(\mathbf{x}) \geq q\} &= \Pr\left\{\frac{\sqrt{n}\left(\bar{C}(\mathbf{x}) - c(x)\right)}{\sigma(\mathbf{x})} \geq \frac{\sqrt{n}\left(q - c(\mathbf{x})\right)}{\sigma(\mathbf{x})}\right\} \\ &\stackrel{n\to\infty}{\longrightarrow} \Pr\{\mathrm{N}(0,1) \geq \lambda\}\end{aligned}$$

There are three cases for λ:

1. If $\mathbf{x}$ is infeasible then $q - c(\mathbf{x}) > 0$ so $\lambda = \infty$.
2. If $\mathbf{x}$ is strictly feasible then $q - c(\mathbf{x}) < 0$ so $\lambda = -\infty$.
3. If $\mathbf{x}$ is tight then $q - c(\mathbf{x}) = 0$ so λ is 0. *This means that no matter how much simulation we do, we cannot determine with certainty if a constraint is tight.*

Compromise

As opposed to feasible-infeasible, we accept some sloppiness:

If $\mathrm{E}[C(\mathbf{x})] \geq q + \varepsilon$ it is **desirable** to declare $\mathbf{x}$ is feasible: $\mathbf{x} \in D$

If $q - \varepsilon \leq \mathrm{E}[C(\mathbf{x})] < q + \varepsilon$ it is **acceptable** to declare $\mathbf{x}$ is feasible or infeasible: $\mathbf{x} \in A$

If $\mathrm{E}[C(\mathbf{x})] < q - \varepsilon$ it is **unacceptable** to declare $\mathbf{x}$ is feasible: $\mathbf{x} \in U$

We would like $\Pr\{D \subset \mathcal{F} \subset (D \cup A)\} \geq 1 - \alpha$.

Set of solutions declared feasible

Constraint-guided search

A natural approach is to add the constraints to the objective as a penalty.

Let $i = 0, 1, 2, \ldots$ be the iteration index, and let $n_i(\mathbf{x})$ be the total number of replications obtained from scenario $\mathbf{x}$ through iteration i.

Define the estimated value of scenario $\mathbf{x}$ through iteration i to be

$$\widehat{\theta}(\mathbf{x}) = \bar{Y}(\mathbf{x}) + \beta_i \max\left\{0, q - \bar{C}(\mathbf{x})\right\}$$

where the averages include all observations.

Do we want $\beta_i \to \infty$?

Probability of a large penalty

Consider the probability of a penalty larger than, say, $\delta > 0$:

$$
\begin{aligned}
&\Pr\left\{\beta_i \max\left\{0, q - \bar{C}(\mathbf{x})\right\} > \delta\right\} \\
&\quad \geq \quad \Pr\left\{\beta_i \left(q - \bar{C}(\mathbf{x})\right) > \delta\right\} \\
&\quad = \quad \Pr\left\{\bar{C}(\mathbf{x}) - q < -\frac{\delta}{\beta_i}\right\} \\
&\quad = \quad \Pr\left\{\frac{\sqrt{n_i(\mathbf{x})}\left(\bar{C}(\mathbf{x}) - c(\mathbf{x})\right)}{\sigma} < \frac{\sqrt{n_i(\mathbf{x})}}{\sigma}\left(q - c(\mathbf{x}) - \frac{\delta}{\beta_i}\right)\right\} \\
&\quad \to \quad \Pr\left\{\mathrm{N}(0,1) < \frac{\sqrt{n_i(\mathbf{x})}}{\sigma}\left(q - c(\mathbf{x}) - \frac{\delta}{\beta_i}\right)\right\}
\end{aligned}
$$

Analysis

1. If $\mathbf{x}$ is infeasible then we want this probability to get large quickly. Since $q - c(\mathbf{x}) > 0$ and $\delta > 0$, this means we would like β_i to become large quickly (big penalty), so $\beta_i \to \infty$ is fine.

2. If $\mathbf{x}$ is feasible but not tight, then we want this probability to get small quickly (no penalty). Since $q - c(\mathbf{x}) < 0$ and $\delta > 0$, we would like β_i to remain small and ideally go to 0.

3. If $\mathbf{x}$ is tight then we also want this probability to get small quickly (no penalty). Since $q - c(\mathbf{x}) = 0$ we *need* β_i to go to 0 to avoid a penalty.

Therefore, penalties should adapt to the observed (in)feasibility.