Latin Supercube Sampling for Very High Dimensional Simulations

by

Art B. Owen

Stanford University

http://www-stat.stanford.edu/reports/owen

Numerical Problems become Statistical in high dimensions

Examples in $[0, 1]^d$

1. Integration$\surd$
2. Approximation
3. Search

Rationale:

Only a very sparse sample of the space is possible, the error depends on the part you don't see, and the error must be estimated somehow.

Common Alternative:

Get good estimate $\hat{I}_0$ and much better estimate $\hat{I}_1$.
Error in $\hat{I}_0 \doteq |\hat{I}_1 - \hat{I}_0|$

Red herring: Function not random.

Integration

$$I = \int_{[0,1]^d} f(X) dX$$

f subsumes

- Domain transformations (to $[0,1]^d$)
- Nonuniform sampling density
- Importance weighting
- Periodizing transformation
- Transformations to reduce effective dimension

Bahvalov showed it is intractable (worst case)

Examples

Transport simulation

Follow trajectory of:

Radioactive particles through shield

Photons to viewing plane in graphics

Heat particles (Laplace's equation)

Financial valuation

Assess value, or value at risk

Stochastic process X_t (e.g. interest rates)

Derivative $Y = f(X_1, \ldots, X_T)$

Want $E(Y)$, $V(Y)$, $Q_{0.05}(Y)$

Boyle, Broadie, Caflisch, Glasserman, Joy, Tan

Examples Ctd.

Queue simulations

Given arrival process $A_1, A_2, \ldots$

and service times $S_1, S_2, \ldots$

How long is queue at time T?

How long until queue is half full?

Optimal Expectations

$$I(t) = \int f(x,t)dx$$

Want $\arg\min_t I(t)$

Experimental design Cohn, Yue

Stochastic linear programming Infanger

Inference

Posterior means

Some bootstraps

$d = 1$, methods and errors

- Midpoint rule, $O(n^{-2})$
- Trapezoid rule, $O(n^{-2})$
- Simpson's rule, $O(n^{-4})$
- Generic rule $n^{-r}\|f^{(r)}\|$

Davis and Rabinowitz

Small $d > 1$, Iterated integrals

by Fubini...

$$\int f(x_1, \dots, x_d) dx$$
$$= \int_0^1 \cdots \int_0^1 f(x_1, \dots, x_d) dx_1 \cdots dx_d$$

Get error $O(n^{-r/d}) \dots n = n_1^d,\ n_1 \geq r$

Same as worst case rate (Bahvalov)

Working definition

"d is **large** if grids are impractical"

High dimensional methods

Monte Carlo

$$I = \int f(x)dx, \quad \sigma^2 = \int (f(x) - I)^2 dx$$

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} f(x_i), \quad x_i \sim U[0,1]^d$$

$$E(\hat{I}) = I, \quad V(\hat{I}) = \frac{\sigma^2}{n}, \quad E(s^2) = \sigma^2$$

Summary:

- ERR $= O_p(n^{-1/2})$ (all d)
- Get sample based estimate of error
- Variance reduction tricks improve const (not rate)

High dimensional methods continued

Quasi-Monte Carlo

Spread x_i uniformly in $[0,1]^d$

Avoid clusters and gaps

Get "representative sample"

See: Niederreiter's (1992) monograph

Error bounds

$$I = \int f(x)dF(x), \quad \hat{I} = \int f(x)F_n(x)$$

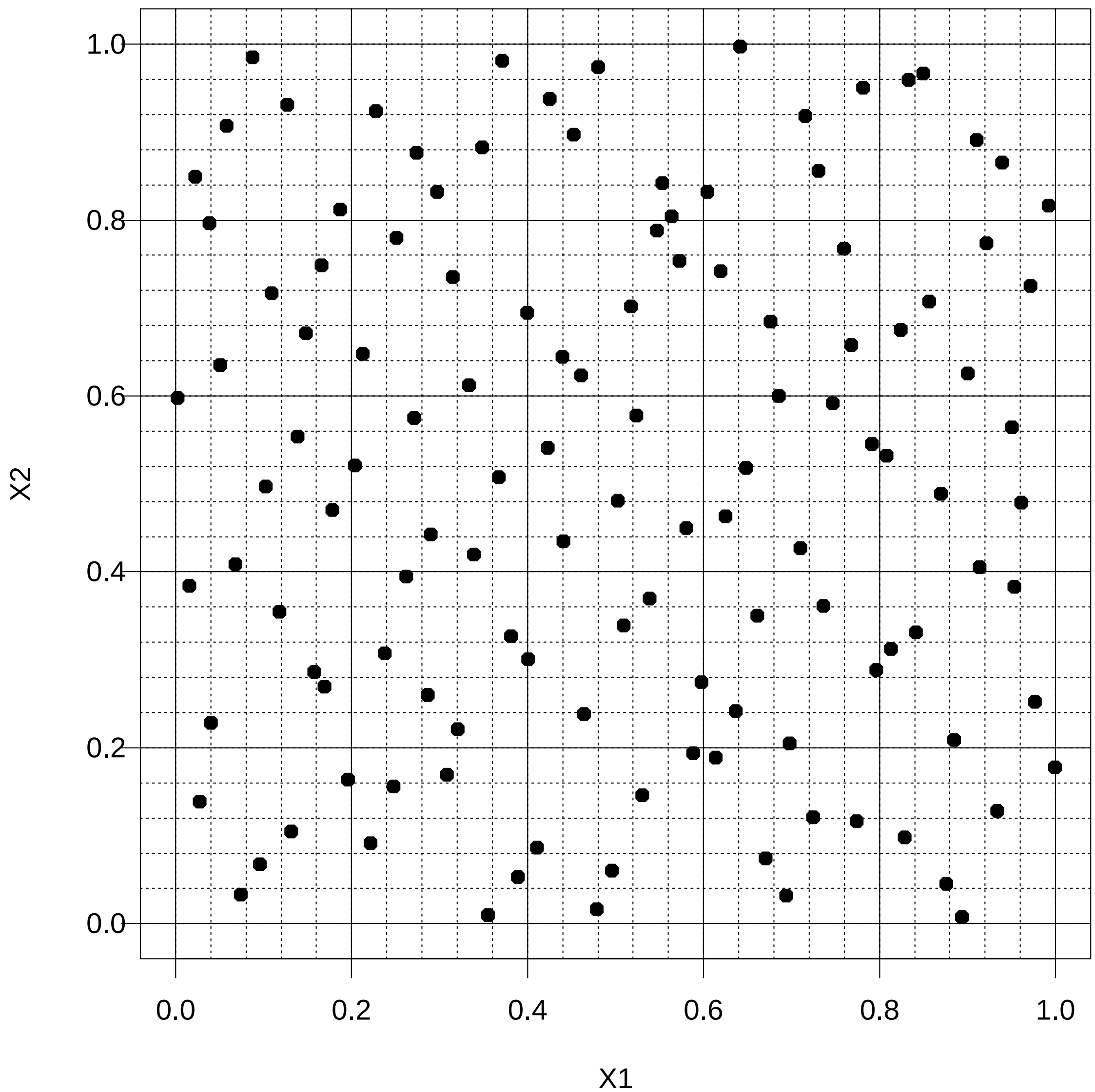
$$F = U[0,1]^d, \quad F_n = U\{x_1, \ldots, x_n\}$$

$$|I - \hat{I}| \leq \|F - F_n\| \times \|f\|^*$$

*Koksma-Hlawka inequality and generalizations

(Niederreiter, Hickernell)

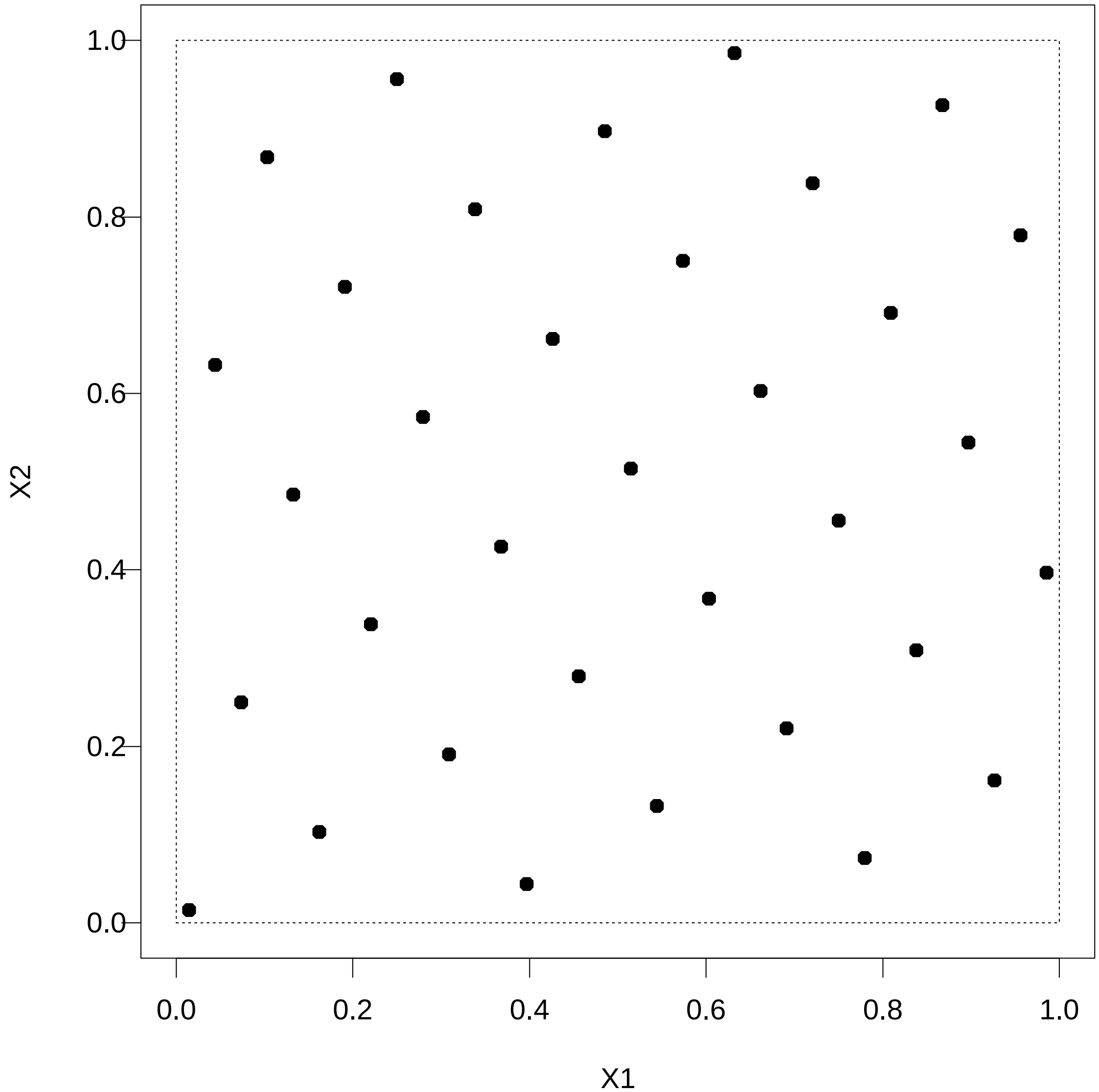
Nets



Two d view of 125 points in $[0,1]^5$

Constructions: Sobol, Faure, Niederreiter, Xing

Lattices



Texts: Sloan & Joe, Fang & Wang, Hua & Wang

Great for smooth periodic functions

QMC vs MC

- QMC can get ERR $= O\left(\frac{1}{n}(\log n)^{d-1}\right)$
- Hard to estimate $\left|\hat{I} - I\right|$ with QMC (Don't just wait for answer to "converge"**!**)
- In examples QMC usually beats MC

For large d

- The gain disappears (Morokoff, Caflisch)
- The gain remains (Paskov, Traub)
- It depends on f (Caflisch, Morokoff, Owen)

C.M.O. Findings

"QMC does well if the effective dimension is not large"

Hybrid methods

- $A_1, \dots, A_n$ a QMC
- $A_i \longrightarrow X_i$ randomized (carefully)
- $X_1, \dots, X_n$ still QMC, but
- each $X_i \sim U[0,1]^d$

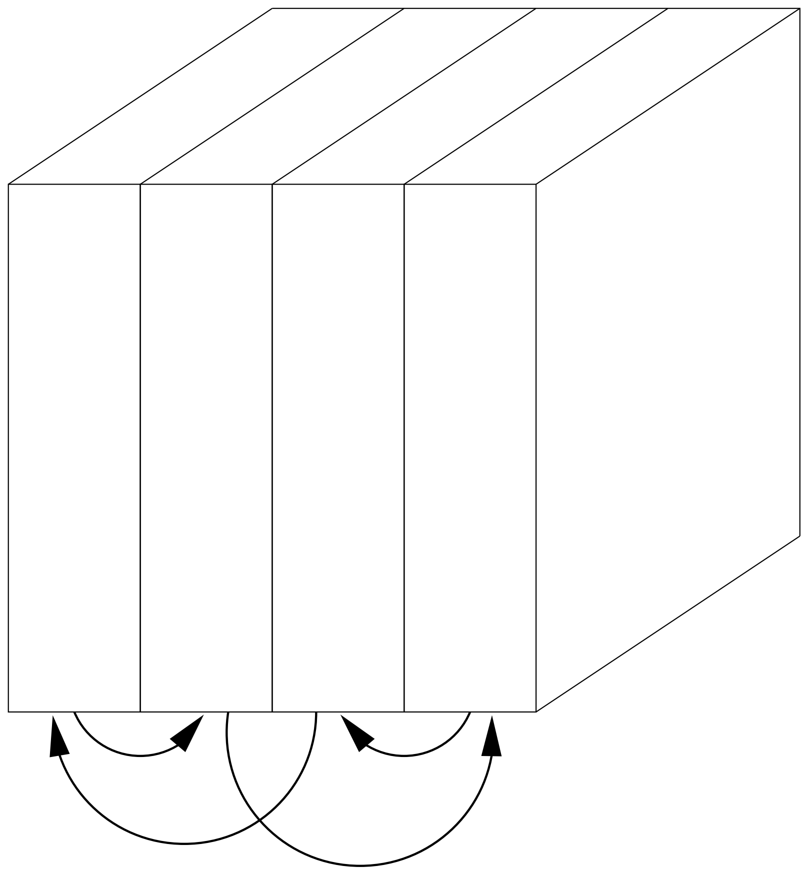
Surprise!

Can get ERR $= O_p\left(n^{-3/2}(\log n)^{(d-1)/2}\right)$

Replication

1. Get $\hat{I}_1, \dots, \hat{I}_r$ iid (small r)
2. Use $\hat{I} = \frac{1}{r}\sum_{j=1}^r \hat{I}_j$
3. and $\hat{V}(\hat{I}) = \frac{1}{r(r-1)}\sum_{j=1}^r (\hat{I}_j - \hat{I})^2$

Scrambled Nets



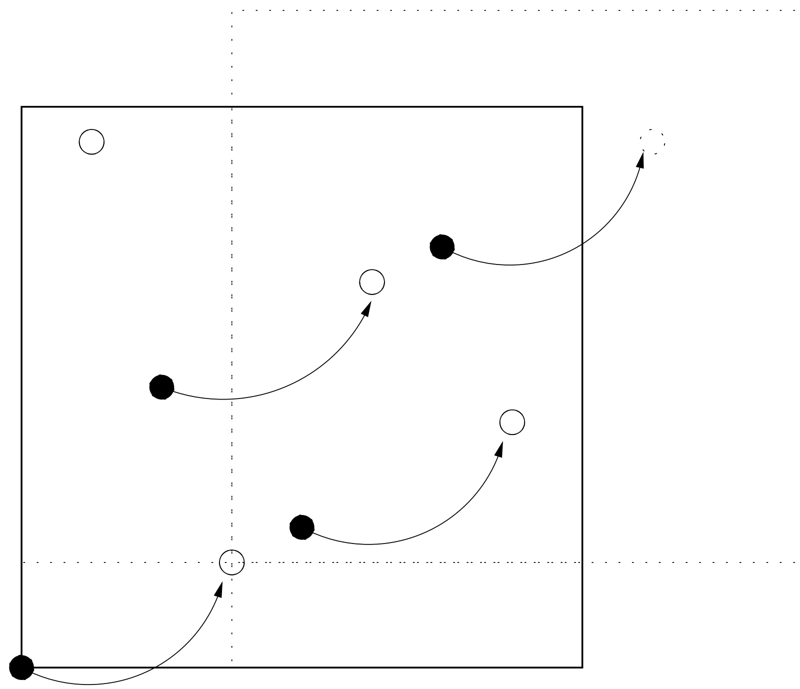
1. Chop $[0,1]^d$ into congruent pieces
2. Randomly permute them
3. Apply recursively to each piece
4. Apply to all d axes

Scrambled Net Results

1. $X_1, \ldots, X_n$ still a net
2. Each $X_i \sim U[0,1]^d$
3. $V_{SNET}(\hat{I}) = o(1/n)$ any f, $n = \lambda b^m$
4. So $V_{SNET}(\hat{I})/V_{MC}(\hat{I}) \to 0$
5. $V_{SNET}(\hat{I}) \le 2.7183 V_{MC}(\hat{I})$, any f, $n = \lambda b^m$ from $(0,d)$-net in base b
6. For smooth f,

$$V_{SNET}(\hat{I}) = O(n^{-3}(\log n)^{d-1})$$

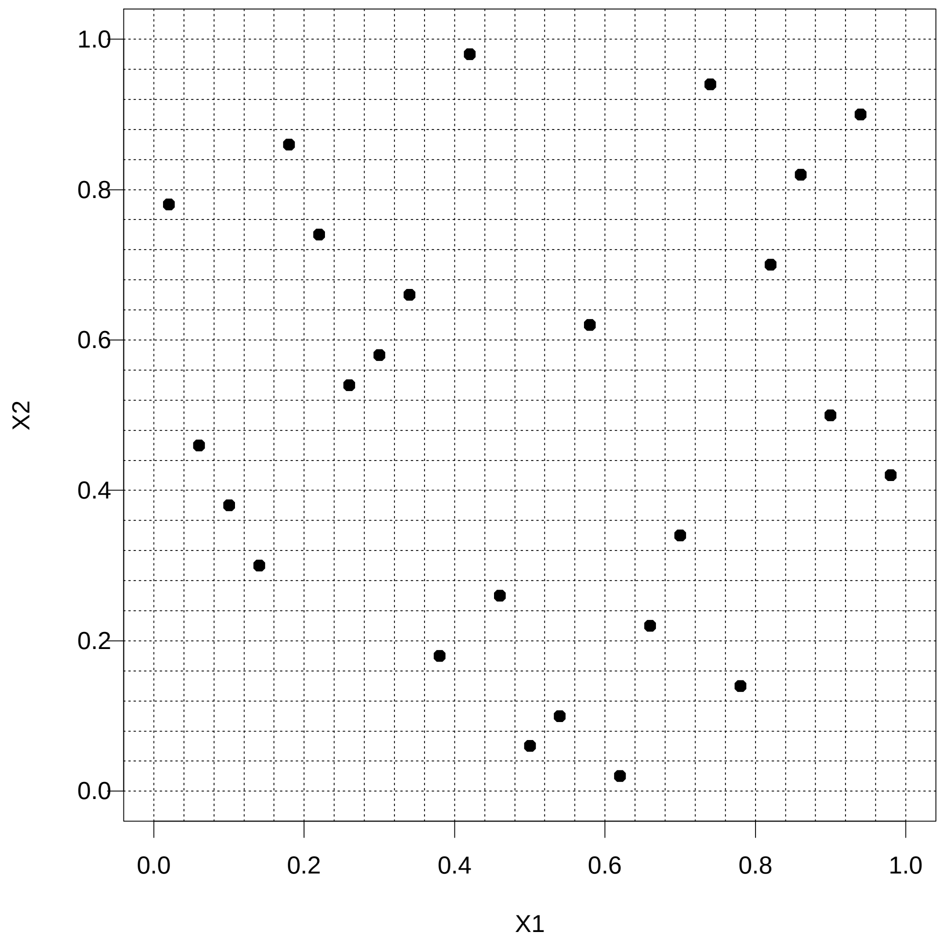
Cranley-Patterson Rotations



1. $A_i = (A_i^1, \ldots, A_i^d)$ in a lattice rule

2. $X_i^j = A_i^j + U^j$ mod 1, $U^j \sim U[0,1]^d$ iid

Latin hypercube sampling



One point per row, one per column

Two versions: centered, and random.

Start with diagonal points, then permute.

Patterson

1. Take midpoint rule $A_i = \frac{i-1/2}{n}$
2. Lift to d dimensions
   (a) $X_i^j = A_{\pi_j(i)}, i = 1, \ldots, n, j = 1, \ldots, d$
   (b) $\pi_j(i)$ indep. random permutations of $1 \ldots n$

McKay, Conover, Beckman

1. Take stratified sample $A_i = \frac{i-V_i}{n}, V_i \sim U[0,1]$
2. Get d independent versions $A_i^j, j = 1, \ldots, d$
3. Lift to d dimensions
   (a) $X_i^j = A_{\pi_j(i)}^j, i = 1, \ldots, n, j = 1, \ldots, d$
   (b) $\pi_j(i)$ indep. random permutations of $1 \ldots n$

LHS Results

1st Never much worse than Monte Carlo (Owen)

$$V_{LHS}(\hat{I}) \leq \frac{n}{n-1} V_{MC}(\hat{I})$$

2nd Additive part of f removed from error (Stein)

$$\begin{aligned} V_{LHS}(\hat{I}) &\doteq \frac{1}{n}\sigma^2(f - f_{\text{Add}}) \\ &= \frac{1}{n}\Big(\sigma^2(f) - \sigma^2(f_{\text{Add}})\Big) \end{aligned}$$

ANOVA of $[0,1]^d$, $d < \infty$

Hoeffding, Efron-Stein, Wahba, Owen, Hickernell

Subsets $u \subseteq \{1, 2, \ldots, d\}$

Effects $f_u(X^u) = f_u(X)$ (by extension)

$f(X) = \sum_u f_u(X)$

Anova example

$$\begin{aligned} f(X^1, X^2) &= 100 + 4X^1 + 8X^2 + 12X^1X^2 \\ f_\emptyset &= 109 \\ f_{\{1\}} &= 10X^1 - 5 \\ f_{\{2\}} &= 14X^2 - 7 \\ f_{\{1,2\}} &= 3(2X^1 - 1)(2X^2 - 1) \end{aligned}$$

Additive part

$$f_{\text{Add}} = f_\emptyset + f_{\{1\}} + \cdots + f_{\{d\}}$$

Anova properties

$$
\begin{aligned}
f(X) &= \sum_u f_u(X^u) \\
f_\emptyset &= I \qquad \text{(Constant)} \\
\int f_u(X) f_v(X) &= 0, \quad u \neq v \\
\int_0^1 f_u(X) dX^j &= 0, \quad j \in u \\
\sigma^2(f) &= \sum_{|u|>0} \sigma^2(f_u) \\
\sigma^2(f_u) &= \int f_u(X)^2, \quad |u| > 0 \\
\sigma^2(f_\emptyset) &= 0
\end{aligned}
$$

Very large dimension

- For large d QMC may require $n \propto d^2$
- Awkward for $d = 1000$
- Worse for $d = \infty$

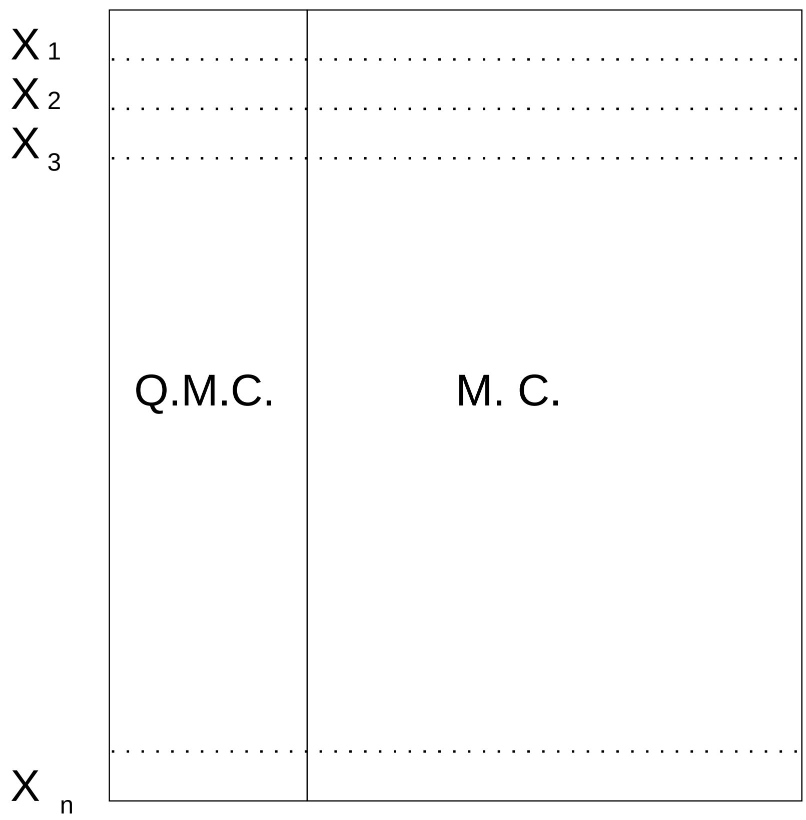
Working definition

“d is **very large** if QMC points hard to compute”

Padding

Spanier, Okten

QMC for s dimensions, MC for $d - s$ dimensions



1. Or, replace QMC by RQMC

2. And/or, replace MC by LHS

MC padding

For RQMC on $A = \{1, 2, \ldots, s\}$ with MC padding

Eventually,

$$V(\hat{I}) \doteq \frac{1}{n}\left[\sigma^2 - \sum_{u \subseteq A} \sigma_u^2\right]$$

Practically, for some $m = m(n)$

$$V(\hat{I}) \doteq \frac{1}{n}\left[\sigma^2 - \sum_{u \subseteq A, |u| \le m} \sigma_u^2\right]$$

Recommendation

Put most important s variables into RQMC set A

LHS padding

For RQMC on $A = \{1, 2, \ldots, s\}$ with LHS padding

Eventually,

$$V(\hat{I}) \doteq \frac{1}{n}\left[\sigma^2 - \sum_{u \subseteq A} \sigma_u^2 - \sum_{j=s+1}^{d} \sigma_{\{j\}}^2\right]$$

Practically, for some $m = m(n)$

$$V(\hat{I}) \doteq \frac{1}{n}\left[\sigma^2 - \sum_{u \subseteq A, |u| \le m} \sigma_u^2 - \sum_{j=s+1}^{d} \sigma_{\{j\}}^2\right]$$

Recommendation

Put most interactive s variables into RQMC set A

Padding, wisely

Engineer f so that $X^1, \ldots, X^s$ are “most important”

Standard Brownian Motion

$$\begin{aligned} X^j \sim U[0,1] \quad &\longrightarrow \quad Z^j \sim N(0,1) \\ &\longrightarrow \quad Y^j = Y^{j-1} + Z^j \end{aligned}$$

Brownian Bridge Encoding

Feynman-Kac, Caflisch-Morokoff-Owen

Given Z^j, generate (conditionally)

$$Z^1 \to Y^d, Z^2 \to Y^{d/2}, Z^3 \to Y^{d/4}, \cdots$$

Principal Components

Acworth, Broadie, Glasserman

1. Use Z^j for jth principal component
2. 5 P.C.s explain 96% of B.M.

Queuing

Fox

1. Draw # arrivals in $[0, T]$ with X^1
2. Draw median arrival time with X^2
3. Draw quartiles using X^3, X^4
4. Etc.
5. Use (R)QMC for first steps

Queuing again

---

1. Draw # arrivals in $[0, T]$ with X^1 (Poisson)
2. Draw # arrivals in $[0, T/2]$ with X^2 (Binomial)
3. Draw # arrivals in $[0, T/4]$ with X^3 (Binomial)
4. Draw # arrivals in $[T/2, 3T/4]$ with X^4 (Binomial)
5. Etc.
6. Use (R)QMC for first steps

IID sampling

Want $Z_1, \ldots, Z_n$ iid

1. Draw $Z_{(1)} = F^{-1}(U_{(1)})$ (Beta)
2. Draw $Z_{(d)} = F^{-1}(U_{(d)})$ (Beta)
3. Draw $Z_{(d/2)} = F^{-1}(U_{(d/2)})$ (Beta)
4. Etc.
5. Assign quantiles to obs (if necessary)
6. Use (R)QMC for first steps

Alternatives

Or, generate $\bar{Z}$, $\bar{Z}^2$ first

Latin Supercube Sampling

- $d = ks$
- Use k copies of (R)QMC points $\mathcal{X}_i \in [0,1]^s$
- $X_i = (\mathcal{X}_{\pi_1(i)}, \mathcal{X}_{\pi_2(i)}, \cdots, \mathcal{X}_{\pi_k(i)})$

	$X^{\{1,2,3,4\}}$	$X^{\{5,6,7,8\}}$	$X^{\{9,10,11,12\}}$
X_1	$\mathcal{X}_{353}$	$\mathcal{X}_{19}$	$\mathcal{X}_{989}$
X_2	$\mathcal{X}_{67}$	$\mathcal{X}_{67}$	$\mathcal{X}_{296}$
X_3	$\mathcal{X}_{123}$	$\mathcal{X}_{567}$	$\mathcal{X}_{721}$
X_4	$\mathcal{X}_{421}$	$\mathcal{X}_{755}$	$\mathcal{X}_{433}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
X_{1000}	$\mathcal{X}_{921}$	$\mathcal{X}_{304}$	$\mathcal{X}_{251}$

Examples

---

1. (R)QMC on 5 P.C.s from each B.M. used (finance)
2. (R)QMC for each collision (transport problems)
3. (R)QMC for each collision feature (dx, dy etc.)
4. (R)QMC for each arrival/service stream (queuing)

LSS Error Analysis, $d < \infty$

$$\hat{I} - I = \hat{I} - I_G + I_G - I$$

$I_G =$ Average over "big grid"

$$I_G = \frac{1}{n^k} \sum_{i_1=1}^{n} \cdots \sum_{i_k=1}^{n} f(\mathcal{X}_{i_1} \cdots \mathcal{X}_{i_k})$$

Sampling Error

$\hat{I} - I_G \equiv k$ dim LHS error

Quadrature Error

$I_G - I \equiv$ sum of k (R)QMC errors (Fubini)

(R)QMC Sampling Distribution

Partition inputs into k sets:

- Use $\mathcal{X}^r \in [0,1]^{A_r}$
- $A_r \subseteq \{1,2,\dots,d\}$
- $A_r \cap A_q = \emptyset$, $r \neq q$
- $\cup_{r=1}^k A_r = \{1,2,\dots,d\}$
- $X = (\mathcal{X}^1,\dots,\mathcal{X}^k)$

Sampling Error

$$
\begin{aligned}
E(\hat{I} - I_G) &= 0 \\
V(\hat{I} - I_G) &\doteq \frac{1}{n}\left[\sigma^2 - \sum_{r=1}^k \sum_{u \subseteq A_r} \sigma_u^2\right]
\end{aligned}
$$

(R)QMC Quadrature Error

- $|I_G - I| \doteq O(kE)$, $E = s$ dim (R)QMC err
- So $|I_G - I| = o(n^{-1/2})$
- With luck: asymptotics relevant, $|I_G - I|$ negligible

QMC vs RQMC

- QMC: $I_G - I$ nonrandom, a bias
- RQMC: $E(I_G - I) = 0$ random,

  contributes to variance

If $I_G - I$ not negligible

- In RQMC errors cancel (in replications)
- In QMC errors don't cancel

What if $d = \infty$?

1. Usual derivation of $V_{LHS}(\hat{I})$ crashes:
   Have to average over volume $[1 - 1/n]^d \to 0$
2. Uncountably many ANOVA terms to sum!
3. **What is** interaction of $X^2, X^3, X^5, X^7, \cdots$?

Is f "approximately finite dimensional"?

1. $f(X_i)$ must only use initial segment
   $X^1, \ldots, X^{M(i)}$
2. Leading X^j usually most important.
3. Maybe "all but ϵ" of variance is in first variables

Martingale Truncation

Williams

For $s \geq 1$

$$\begin{aligned} & f^s(x^1, \ldots, x^s) \\ = \quad & E(f(X) | X^1 = x^1, \ldots, X^s = x^s) \end{aligned}$$

For $X \in [0,1]^\infty$ take

$$f^s(X) = f^s(X^1, \ldots, X^s)$$

Then

$$E(f^{s+1}(X) | X^{\{1,2,\ldots,s\}}) = f^s(X)$$

$Y^s = f^s(X), \; s \geq 1$ is a martingale

Finite variance does it

If $\int f(X)^2 < \infty$ then $\forall \epsilon > 0, \exists s < \infty$

$$E\Big([f^s(X) - f(X)]^2 \Big) < \epsilon$$

Consequences

$$\begin{aligned} V_{LHS}(\hat{I}) &\leq \frac{n}{n-1} V_{MC}(\hat{I}), \quad d = \infty \\ V_{LHS}(\hat{I}) &\doteq \frac{1}{n} \left[\sigma^2 - \sum_{j=1}^{\infty} \sigma^2_{\{j\}} \right] \\ \sigma^2_u &= 0, \quad |u| = \infty \end{aligned}$$

And ... LSS works for $k = \infty$

Conclusions

“It depends on f”

1. Large $d \Rightarrow$ integration intractable
2. . . . in the worst case

Success for large d means

- f was somehow “special”,
- and our method could exploit it,
- *but not* “curse of d lifted”

Tasks

1. Find special structures
2. ways to exploit them
3. ways to induce them

RQMC and LLS exploit lower “effective dimension”