

Randomized Quasi-Monte Carlo: Theory, Choice of Discrepancy, and Applications

Pierre L'Ecuyer and David Munger

Informatique et Recherche Opérationnelle, Université de Montréal

Randomized Quasi-Monte Carlo: Theory, Choice of Discrepancy, and Applications featuring randomly-shifted lattice rules

Pierre L'Ecuyer and David Munger

Informatique et Recherche Opérationnelle, Université de Montréal

1. Monte Carlo (MC), quasi-MC (QMC), randomized QMC (RQMC).
2. Lattice rules and RQMC variance.
3. Weighted discrepancies and choice of weights.
4. Several examples.

Basic Monte Carlo setting

Want to estimate

$$\mu = \mu(f) = \int_{[0,1]^s} f(\mathbf{u}) d\mathbf{u} = \mathbb{E}[f(\mathbf{U})]$$

where $f : [0, 1)^s \rightarrow \mathbb{R}$ and \mathbf{U} is a uniform r.v. over $[0, 1)^s$.

Standard Monte Carlo:

- ▶ Generate n independent copies of \mathbf{U} , say $\mathbf{U}_1, \dots, \mathbf{U}_n$;
- ▶ estimate μ by $\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n f(\mathbf{U}_i)$.

Basic Monte Carlo setting

Want to estimate

$$\mu = \mu(f) = \int_{[0,1]^s} f(\mathbf{u}) d\mathbf{u} = \mathbb{E}[f(\mathbf{U})]$$

where $f : [0, 1]^s \rightarrow \mathbb{R}$ and \mathbf{U} is a uniform r.v. over $[0, 1]^s$.

Standard Monte Carlo:

- ▶ Generate n independent copies of \mathbf{U} , say $\mathbf{U}_1, \dots, \mathbf{U}_n$;
- ▶ estimate μ by $\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n f(\mathbf{U}_i)$.

Almost sure convergence as $n \rightarrow \infty$ (strong law of large numbers).

For confidence interval of level $1 - \alpha$, can use central limit theorem:

$$\mathbb{P} \left[\mu \in \left(\hat{\mu}_n - \frac{c_\alpha S_n}{\sqrt{n}}, \hat{\mu}_n + \frac{c_\alpha S_n}{\sqrt{n}} \right) \right] \approx 1 - \alpha,$$

where S_n^2 is any consistent estimator of $\sigma^2 = \text{Var}[f(\mathbf{U})]$.

Quasi-Monte Carlo (QMC)

Replace the random points \mathbf{U}_i by a set of **deterministic** points

$P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$ that cover $[0, 1)^s$ **more evenly**.

This P_n is called a **highly-uniform** or **low-discrepancy point set** if some measure of **discrepancy** between the empirical distribution of P_n and the uniform distribution $\rightarrow 0$ faster than for independent random points.

Quasi-Monte Carlo (QMC)

Replace the random points \mathbf{U}_i by a set of **deterministic** points

$P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$ that cover $[0, 1)^s$ **more evenly**.

This P_n is called a **highly-uniform** or **low-discrepancy point set** if some measure of **discrepancy** between the empirical distribution of P_n and the uniform distribution $\rightarrow 0$ faster than for independent random points.

If infinite sequence of points: **low-discrepancy sequence**.

For P_n , take the first n points, for any n .

Quasi-Monte Carlo (QMC)

Replace the random points \mathbf{U}_i by a set of **deterministic** points

$P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$ that cover $[0, 1)^s$ **more evenly**.

This P_n is called a **highly-uniform** or **low-discrepancy point set** if some measure of **discrepancy** between the empirical distribution of P_n and the uniform distribution $\rightarrow 0$ faster than for independent random points.

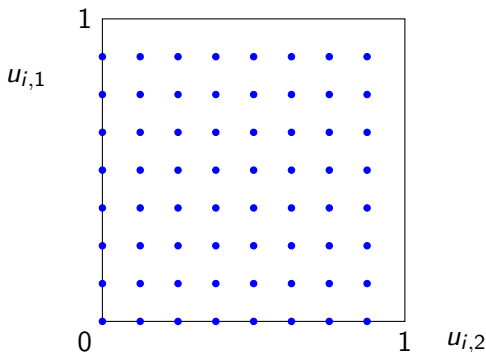
If infinite sequence of points: **low-discrepancy sequence**.

For P_n , take the first n points, for any n .

Main construction methods: **lattice rules** and **digital nets**
(Korobov, Hammersley, Halton, Sobol', Faure, Niederreiter, etc.)

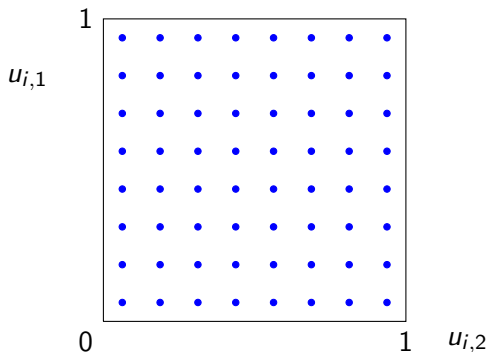
Simplistic solution: rectangular grid

$P_n = \{(i_1/d, \dots, i_s/d) \text{ such that } 0 \leq i_j < d \ \forall j\}$ where $n = d^s$.



Simplistic solution: rectangular grid

$P_n = \{(i_1/d, \dots, i_s/d) \text{ such that } 0 \leq i_j < d \ \forall j\}$ where $n = d^s$.

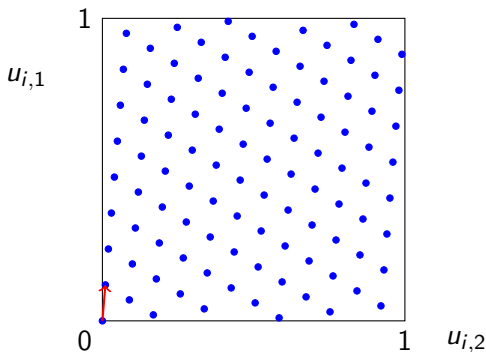


Quickly becomes impractical when s increases.

And each one-dimensional projection has only d distinct points, each two-dimensional projections has only d^2 distinct points, etc.

Example: lattice with $s = 2$, $n = 101$, $a = 12$

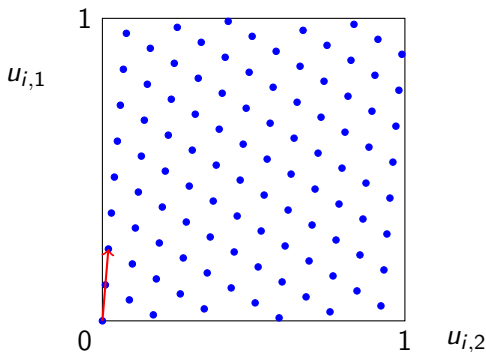
$$\begin{aligned}
 P_n &= \{(x/m, (ax/m) \bmod 1) : x = 0, \dots, m-1\} \\
 &= \{(x/101, (12x/101) \bmod 1) : x = 0, \dots, 100\}
 \end{aligned}$$



Here, each one-dimensional projection is $\{0, 1/n, \dots, (n-1)/n\}$.

Example: lattice with $s = 2$, $n = 101$, $a = 12$

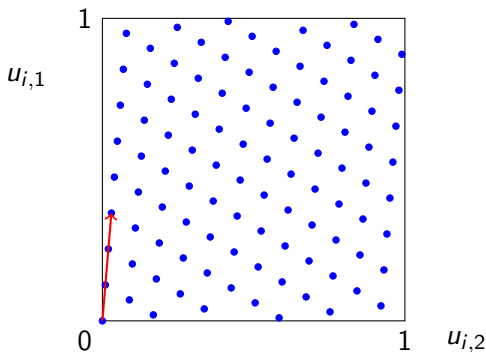
$$\begin{aligned}
 P_n &= \{(x/m, (ax/m) \bmod 1) : x = 0, \dots, m-1\} \\
 &= \{(x/101, (12x/101) \bmod 1) : x = 0, \dots, 100\}
 \end{aligned}$$



Here, each one-dimensional projection is $\{0, 1/n, \dots, (n-1)/n\}$.

Example: lattice with $s = 2$, $n = 101$, $a = 12$

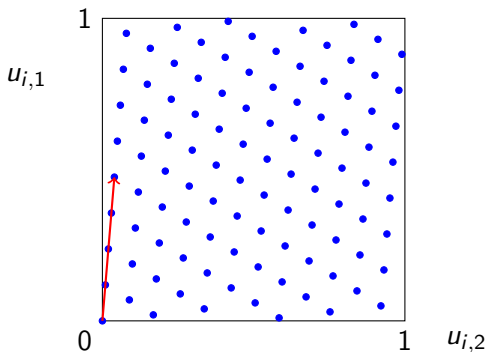
$$\begin{aligned}
 P_n &= \{(x/m, (ax/m) \bmod 1) : x = 0, \dots, m-1\} \\
 &= \{(x/101, (12x/101) \bmod 1) : x = 0, \dots, 100\}
 \end{aligned}$$



Here, each one-dimensional projection is $\{0, 1/n, \dots, (n-1)/n\}$.

Example: lattice with $s = 2$, $n = 101$, $a = 12$

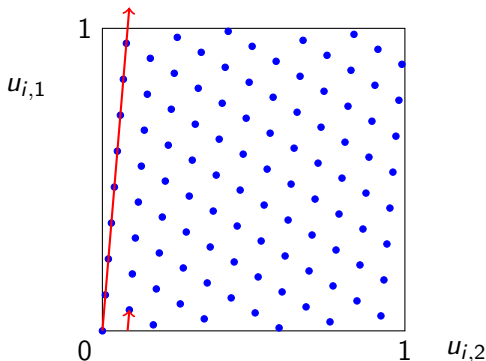
$$\begin{aligned}
 P_n &= \{(x/m, (ax/m) \bmod 1) : x = 0, \dots, m-1\} \\
 &= \{(x/101, (12x/101) \bmod 1) : x = 0, \dots, 100\}
 \end{aligned}$$



Here, each one-dimensional projection is $\{0, 1/n, \dots, (n-1)/n\}$.

Example: lattice with $s = 2$, $n = 101$, $a = 12$

$$\begin{aligned}
 P_n &= \{(x/m, (ax/m) \bmod 1) : x = 0, \dots, m-1\} \\
 &= \{(x/101, (12x/101) \bmod 1) : x = 0, \dots, 100\}
 \end{aligned}$$



Here, each one-dimensional projection is $\{0, 1/n, \dots, (n-1)/n\}$.

Two problems: (1) point at $(0,0)$ and (2) how to estimate the error?

Randomized quasi-Monte Carlo (RQMC)

An RQMC estimator of μ has the form

$$\hat{\mu}_{n,\text{rqmc}} = \frac{1}{n} \sum_{i=0}^{n-1} f(\mathbf{U}_i),$$

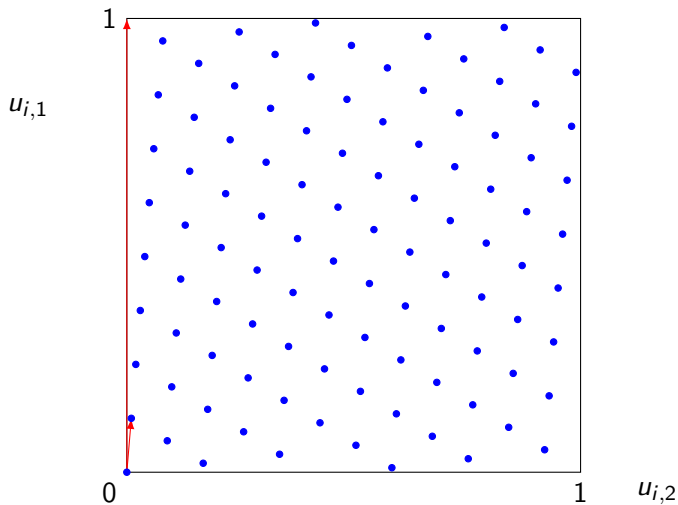
with $P_n = \{\mathbf{U}_0, \dots, \mathbf{U}_{n-1}\} \subset (0, 1)^s$ an RQMC point set:

- (i) each point \mathbf{U}_i has the uniform distribution over $(0, 1)^s$;
- (ii) P_n as a whole is a low-discrepancy point set.

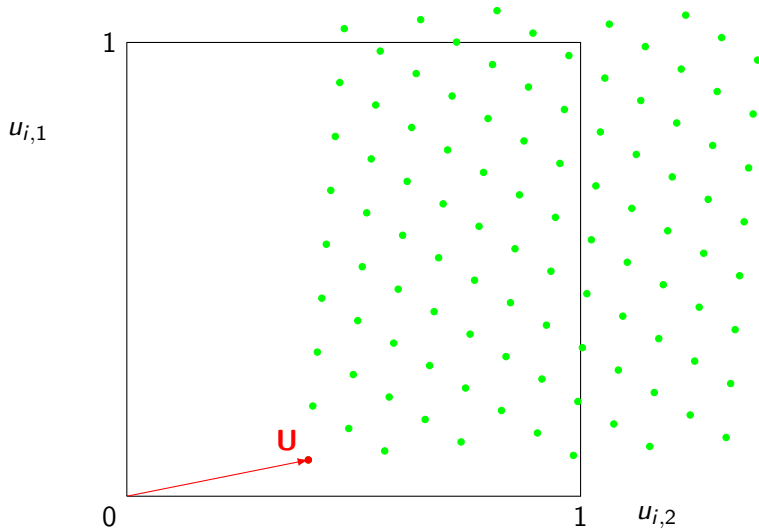
$$\mathbb{E}[\hat{\mu}_{n,\text{rqmc}}] = \mu \quad (\text{unbiased}).$$

Can perform m independent realizations X_1, \dots, X_m of $\hat{\mu}_{n,\text{rqmc}}$, then estimate μ and $\text{Var}[\hat{\mu}_{n,\text{rqmc}}]$ by their sample mean \bar{X}_m and sample variance S_m^2 (also unbiased).

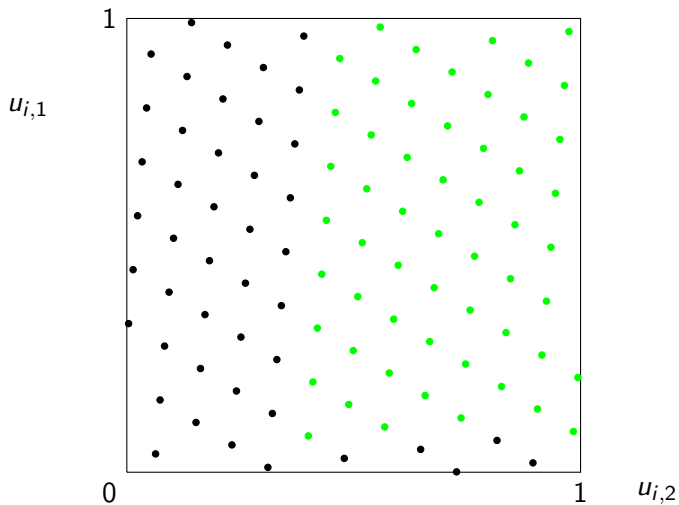
Example: lattice with $s = 2$, $n = 101$, $a = 12$



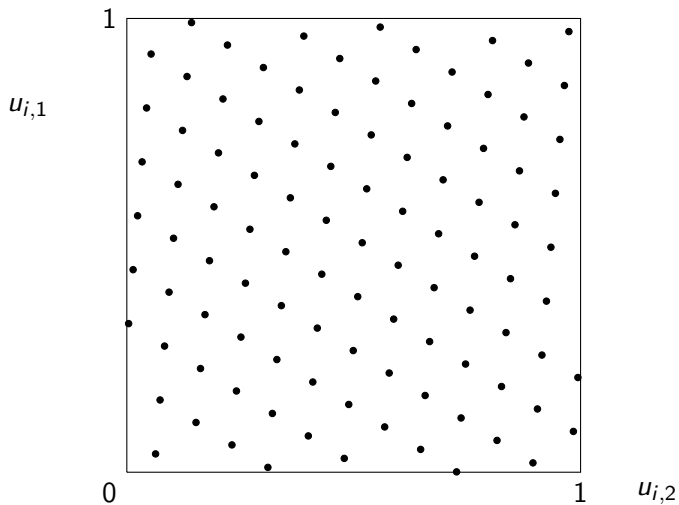
Example: lattice with $s = 2$, $n = 101$, $a = 12$



Example: lattice with $s = 2$, $n = 101$, $a = 12$



Example: lattice with $s = 2$, $n = 101$, $a = 12$



Generalized antithetic variates and RQMC

$$\begin{aligned}\text{Var}[\hat{\mu}_{n,\text{rqmc}}] &= \frac{1}{n^2} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \text{Cov}[f(\mathbf{U}_i), f(\mathbf{U}_j)] \\ &= \frac{\text{Var}[f(\mathbf{U}_i)]}{n} + \frac{2}{n^2} \sum_{i < j} \text{Cov}[f(\mathbf{U}_i), f(\mathbf{U}_j)].\end{aligned}$$

We want to make the last sum as negative as possible.

Special cases:

- antithetic variates ($n = 2$),
- Latin hypercube sampling (LHS),
- randomized quasi-Monte Carlo (RQMC).

Error and variance bounds

Koksma-Hlawka-type inequalities (worst-case error):

$$|\hat{\mu}_{n,\text{rqmc}} - \mu| \leq V(f) \cdot D(P_n)$$

for all f in some Hilbert space or Banach space \mathcal{H} , where

$V(f) = \|f - \mu\|_{\mathcal{H}}$ is the **variation** of f , and $D(P_n)$ is the **discrepancy** of P_n .

Error and variance bounds

Koksma-Hlawka-type inequalities (worst-case error):

$$|\hat{\mu}_{n,\text{rqmc}} - \mu| \leq V(f) \cdot D(P_n)$$

for all f in some Hilbert space or Banach space \mathcal{H} , where

$V(f) = \|f - \mu\|_{\mathcal{H}}$ is the variation of f , and $D(P_n)$ is the discrepancy of P_n .

For RQMC (P_n is random):

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] \leq V^2(f) \cdot \mathbb{E}[D^2(P_n)].$$

Error and variance bounds

Koksma-Hlawka-type inequalities (worst-case error):

$$|\hat{\mu}_{n,\text{rqmc}} - \mu| \leq V(f) \cdot D(P_n)$$

for all f in some Hilbert space or Banach space \mathcal{H} , where

$V(f) = \|f - \mu\|_{\mathcal{H}}$ is the variation of f , and $D(P_n)$ is the discrepancy of P_n .

For RQMC (P_n is random):

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] \leq V^2(f) \cdot \mathbb{E}[D^2(P_n)].$$

With MC: $D(P_n) \approx O(n^{-1/2})$.

Error and variance bounds

Koksma-Hlawka-type inequalities (worst-case error):

$$|\hat{\mu}_{n,\text{rqmc}} - \mu| \leq V(f) \cdot D(P_n)$$

for all f in some Hilbert space or Banach space \mathcal{H} , where

$V(f) = \|f - \mu\|_{\mathcal{H}}$ is the variation of f , and $D(P_n)$ is the discrepancy of P_n .

For RQMC (P_n is random):

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] \leq V^2(f) \cdot \mathbb{E}[D^2(P_n)].$$

With MC: $D(P_n) \approx O(n^{-1/2})$.

“Classical” Koksma-Hlawka (worst-case) inequality for QMC: f must have finite variation in the sense of Hardy and Krause (implies no discontinuity not aligned with the axes), and several known constructions achieve $D(P_n) = O(n^{-1}(\ln n)^s) = O(n^{-1+\delta})$.

For certain Hilbert spaces of smooth functions f with square-integrable partial derivatives of order up to α : $D(P_n) = O(n^{-\alpha+\delta})$.

Lattice rules

Integration lattice:

$$L_s = \left\{ \mathbf{v} = \sum_{j=1}^s z_j \mathbf{v}_j \text{ such that each } z_j \in \mathbb{Z} \right\},$$

where $\mathbf{v}_1, \dots, \mathbf{v}_s \in \mathbb{R}^s$ are linearly independent over \mathbb{R} and where L_s contains \mathbb{Z}^s . **Lattice rule:** Take $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\} = L_s \cap [0, 1)^s$.

Lattice rules

Integration lattice:

$$L_s = \left\{ \mathbf{v} = \sum_{j=1}^s z_j \mathbf{v}_j \text{ such that each } z_j \in \mathbb{Z} \right\},$$

where $\mathbf{v}_1, \dots, \mathbf{v}_s \in \mathbb{R}^s$ are linearly independent over \mathbb{R} and where L_s contains \mathbb{Z}^s . **Lattice rule:** Take $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\} = L_s \cap [0, 1)^s$.

Lattice rule of **rank 1:** $\mathbf{u}_i = i\mathbf{v}_1 \bmod 1$ for $i = 0, \dots, n-1$.

Korobov rule: $\mathbf{v}_1 = \mathbf{z}/n = (1, a, a^2 \bmod n, \dots)/n$.

Lattice rules

Integration lattice:

$$L_s = \left\{ \mathbf{v} = \sum_{j=1}^s z_j \mathbf{v}_j \text{ such that each } z_j \in \mathbb{Z} \right\},$$

where $\mathbf{v}_1, \dots, \mathbf{v}_s \in \mathbb{R}^s$ are linearly independent over \mathbb{R} and where L_s contains \mathbb{Z}^s . **Lattice rule:** Take $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\} = L_s \cap [0, 1)^s$.

Lattice rule of **rank 1:** $\mathbf{u}_i = i\mathbf{v}_1 \bmod 1$ for $i = 0, \dots, n-1$.

Korobov rule: $\mathbf{v}_1 = \mathbf{z}/n = (1, a, a^2 \bmod n, \dots)/n$.

For any $\mathbf{u} \subset \{1, \dots, s\}$, the **projection** $L_s(\mathbf{u})$ of L_s is also a lattice, with point set $P_n(\mathbf{u})$.

Lattice rules

Integration lattice:

$$L_s = \left\{ \mathbf{v} = \sum_{j=1}^s z_j \mathbf{v}_j \text{ such that each } z_j \in \mathbb{Z} \right\},$$

where $\mathbf{v}_1, \dots, \mathbf{v}_s \in \mathbb{R}^s$ are linearly independent over \mathbb{R} and where L_s contains \mathbb{Z}^s . **Lattice rule:** Take $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\} = L_s \cap [0, 1)^s$.

Lattice rule of **rank 1:** $\mathbf{u}_i = i\mathbf{v}_1 \bmod 1$ for $i = 0, \dots, n-1$.

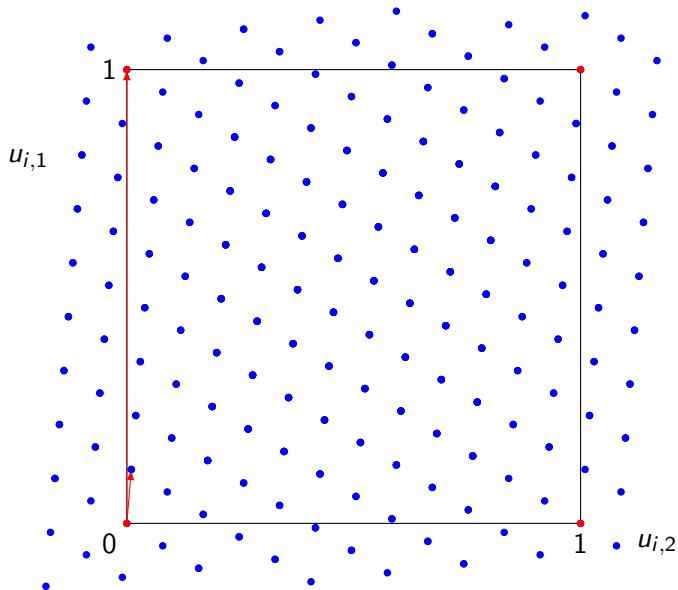
Korobov rule: $\mathbf{v}_1 = \mathbf{z}/n = (1, a, a^2 \bmod n, \dots)/n$.

For any $\mathbf{u} \subset \{1, \dots, s\}$, the **projection** $L_s(\mathbf{u})$ of L_s is also a lattice, with point set $P_n(\mathbf{u})$.

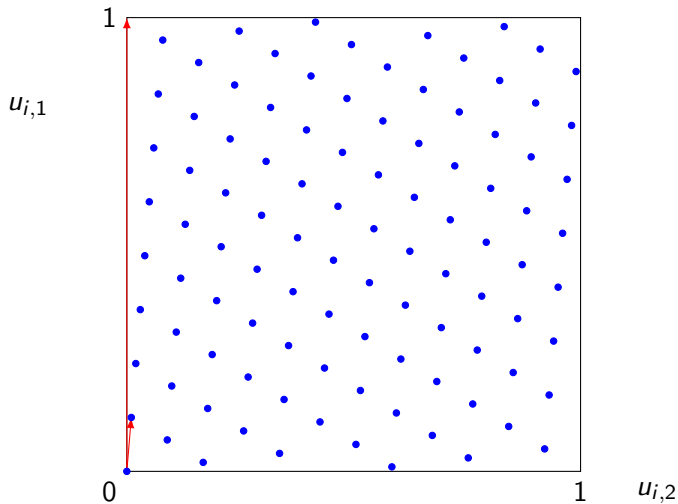
Random shift modulo 1: generate a single point \mathbf{U} uniformly over $(0, 1)^s$ and add it to each point of P_n , modulo 1, coordinate-wise:

$\mathbf{U}_i = (\mathbf{u}_i + \mathbf{U}) \bmod 1$. Each \mathbf{U}_i is uniformly distributed over $[0, 1)^s$.

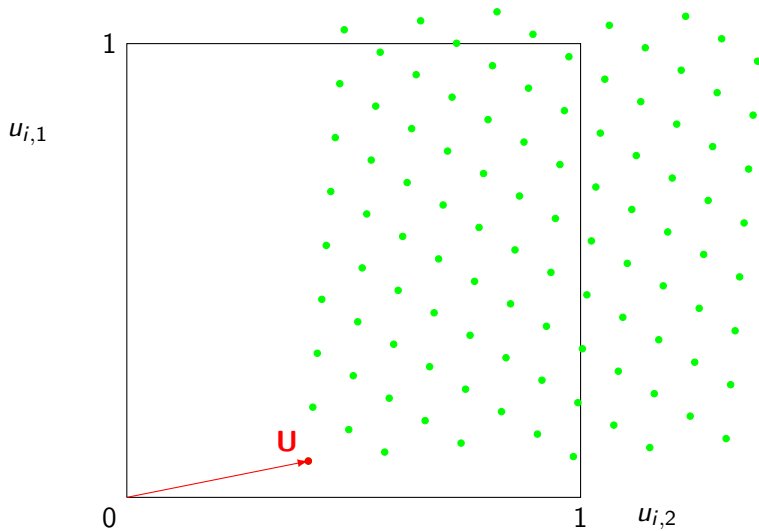
Randomly-shifted lattice: Two-dim. example



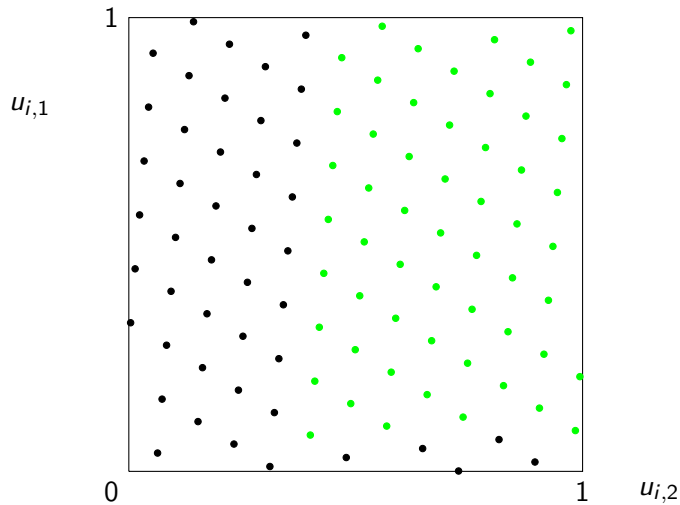
Randomly-shifted lattice: Two-dim. example



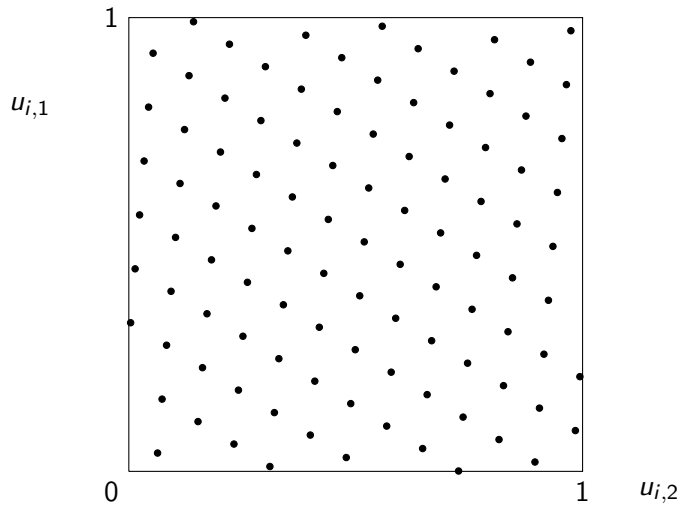
Randomly-shifted lattice: Two-dim. example



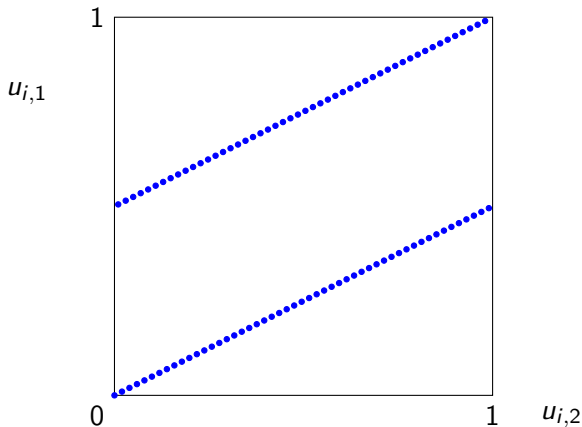
Randomly-shifted lattice: Two-dim. example



Randomly-shifted lattice: Two-dim. example



Example of a poor lattice: $s = 2$, $n = 101$, $a = 51$



Good uniformity in one dimension, but not in two!

Variance expression

Suppose f has Fourier expansion

$$f(\mathbf{u}) = \sum_{\mathbf{h} \in \mathbb{Z}^s} \hat{f}(\mathbf{h}) e^{2\pi \sqrt{-1} \mathbf{h}^t \mathbf{u}}.$$

For a **randomly shifted lattice**, the exact variance is (always)

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} |\hat{f}(\mathbf{h})|^2,$$

where $L_s^* = \{\mathbf{h} \in \mathbb{R}^s : \mathbf{h}^t \mathbf{v} \in \mathbb{Z} \text{ for all } \mathbf{v} \in L_s\} \subseteq \mathbb{Z}^s$ is the dual lattice.

From the viewpoint of variance reduction, an **optimal lattice for f** minimizes the square discrepancy $D^2(P_n) = \text{Var}[\hat{\mu}_{n,\text{rqmc}}]$.

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} |\hat{f}(\mathbf{h})|^2.$$

If f has square-integrable mixed partial derivatives up to order α , and the periodic continuation of its derivatives up to order $\alpha - 1$ is **continuous** across the unit cube boundaries, then

$$|\hat{f}(\mathbf{h})|^2 = \mathcal{O}((\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}).$$

Moreover, there is a $\mathbf{v}_1 = \mathbf{v}_1(n)$ such that

$$\mathcal{P}_{2\alpha} \stackrel{\text{def}}{=} \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha} = \mathcal{O}(n^{-2\alpha+\delta}).$$

This is the variance for a **worst-case f** having

$$|\hat{f}(\mathbf{h})|^2 = (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}.$$

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} |\hat{f}(\mathbf{h})|^2.$$

If f has square-integrable mixed partial derivatives up to order α , and the periodic continuation of its derivatives up to order $\alpha - 1$ is **continuous** across the unit cube boundaries, then

$$|\hat{f}(\mathbf{h})|^2 = \mathcal{O}((\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}).$$

Moreover, there is a $\mathbf{v}_1 = \mathbf{v}_1(n)$ such that

$$\mathcal{P}_{2\alpha} \stackrel{\text{def}}{=} \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha} = \mathcal{O}(n^{-2\alpha+\delta}).$$

This is the variance for a **worst-case** f having

$$|\hat{f}(\mathbf{h})|^2 = (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}.$$

Beware of hidden factor in \mathcal{O} when s is large.

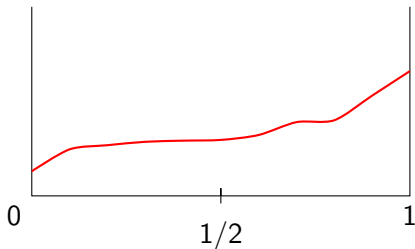
This worst-case function may be far from representative in applications.

Baker's transformation

Want to make the periodic continuation **continuous**.

If $f(0) \neq f(1)$, define \tilde{f} by $\tilde{f}(1-u) = \tilde{f}(u) = f(2u)$ for $0 \leq u \leq 1/2$.

This \tilde{f} has the same integral as f and $\tilde{f}(0) = \tilde{f}(1)$.

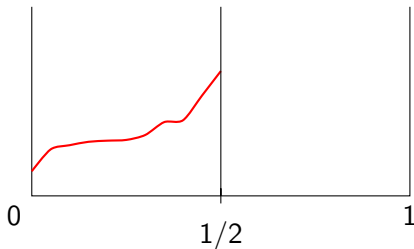


Baker's transformation

Want to make the periodic continuation **continuous**.

If $f(0) \neq f(1)$, define \tilde{f} by $\tilde{f}(1-u) = \tilde{f}(u) = f(2u)$ for $0 \leq u \leq 1/2$.

This \tilde{f} has the same integral as f and $\tilde{f}(0) = \tilde{f}(1)$.

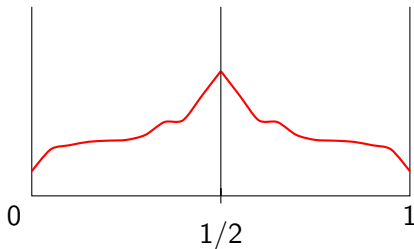


Baker's transformation

Want to make the periodic continuation **continuous**.

If $f(0) \neq f(1)$, define \tilde{f} by $\tilde{f}(1-u) = \tilde{f}(u) = f(2u)$ for $0 \leq u \leq 1/2$.

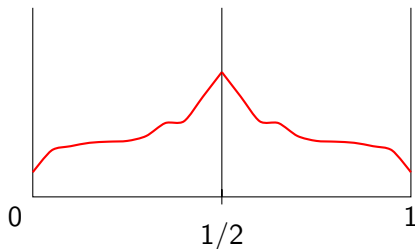
This \tilde{f} has the same integral as f and $\tilde{f}(0) = \tilde{f}(1)$.



Baker's transformation

Want to make the periodic continuation **continuous**.

If $f(0) \neq f(1)$, define \tilde{f} by $\tilde{f}(1-u) = \tilde{f}(u) = f(2u)$ for $0 \leq u \leq 1/2$.
This \tilde{f} has the same integral as f and $\tilde{f}(0) = \tilde{f}(1)$.

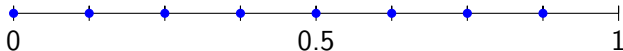


For smooth f , can reduce the variance to $O(n^{-4+\delta})$ (Hickernell 2002).
The resulting \tilde{f} also symmetric with respect to $u = 1/2$.

In practice, we transform the points \mathbf{U}_i instead of f .

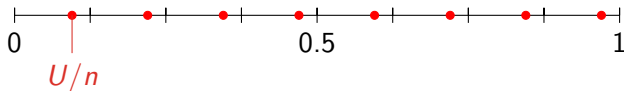
One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n)$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.



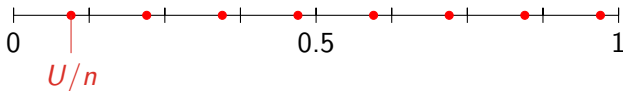
One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n)$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.



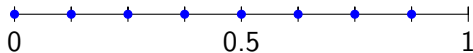
One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n)$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.



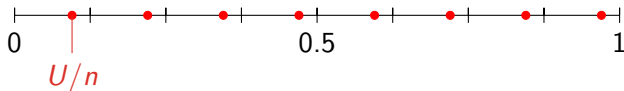
Random shift followed by [baker's transformation](#).

Along each coordinate, stretch everything by a factor of 2 and fold. Same as replacing U_j by $\min[2U_j, 2(1 - U_j)]$.



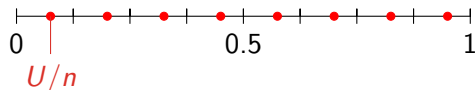
One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n]$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.



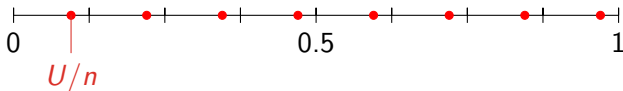
Random shift followed by [baker's transformation](#).

Along each coordinate, stretch everything by a factor of 2 and fold. Same as replacing U_j by $\min[2U_j, 2(1 - U_j)]$.



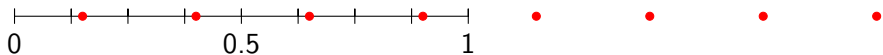
One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n)$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.



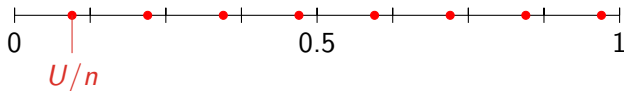
Random shift followed by [baker's transformation](#).

Along each coordinate, stretch everything by a factor of 2 and fold. Same as replacing U_j by $\min[2U_j, 2(1 - U_j)]$.



One-dimensional case

Here the random shift can be seen as uniform over $[0, 1/n]$ and the shifted points become $\{U/n, (1+U)/n, \dots, (n-1+U)/n\}$.

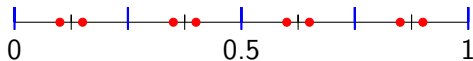


Random shift followed by [baker's transformation](#).

Along each coordinate, stretch everything by a factor of 2 and fold.

Same as replacing U_j by $\min[2U_j, 2(1 - U_j)]$.

Gives [locally antithetic](#) points in intervals of size $2/n$.



$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} |\hat{f}(\mathbf{h})|^2$$

is unpractical, because:

- ▶ the Fourier coefficients are usually unknown,
- ▶ there are infinitely many,
- ▶ must do it for each f .

We nevertheless want to see how far we can go in that direction.

Searching for a lattice that minimizes

$$\text{Var}[\hat{\mu}_{n,\text{rqmc}}] = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} |\hat{f}(\mathbf{h})|^2$$

is unpractical, because:

- ▶ the Fourier coefficients are usually unknown,
- ▶ there are infinitely many,
- ▶ must do it for each f .

We nevertheless want to see how far we can go in that direction.

Let us start with a simple function for which we know the Fourier expansion.

Even then, the discrepancy involves an infinite number of terms!

Possible ideas: Truncate the sum to a finite subset B :

$$\sum_{\mathbf{0} \neq \mathbf{h} \in L_s^* \cap B} |\hat{f}(\mathbf{h})|^2,$$

or to the largest q square coefficients $|\hat{f}(\mathbf{h})|^2$. But hard to implement!

Dual-space exploration

The following makes sense if the $|\hat{f}(\mathbf{h})|^2$ tend to decrease with each $|h_j|$.

Start with a large set \mathcal{L} of lattices (or generating vectors \mathbf{v}_1 , for given n).

Search for vectors \mathbf{h} with large weights $w(\mathbf{h}) = |\hat{f}(\mathbf{h})|^2$, via a neighborhood search starting at $\mathbf{h} = \mathbf{0}$, keeping a sorted list (as in Dijkstra's shortest path algorithm), and eliminate (successively) from \mathcal{L} the lattices whose dual contains \mathbf{h} for the next largest $w(\mathbf{h})$, until a single lattice remains.

Example of neighborhood $\mathcal{N}(\mathbf{h})$: only one coordinate differs, by one unit.

Dual-space exploration

The following makes sense if the $|\hat{f}(\mathbf{h})|^2$ tend to decrease with each $|h_j|$.

Start with a large set \mathcal{L} of lattices (or generating vectors \mathbf{v}_1 , for given n).

Search for vectors \mathbf{h} with large weights $w(\mathbf{h}) = |\hat{f}(\mathbf{h})|^2$, via a neighborhood search starting at $\mathbf{h} = \mathbf{0}$, keeping a sorted list (as in Dijkstra's shortest path algorithm), and eliminate (successively) from \mathcal{L} the lattices whose dual contains \mathbf{h} for the next largest $w(\mathbf{h})$, until a single lattice remains.

Example of neighborhood $\mathcal{N}(\mathbf{h})$: only one coordinate differs, by one unit.

Component-by-component version: For $j = 1, 2, \dots, s$, we apply the algorithm for a set \mathcal{L} of j -dimensional lattices with common (fixed) $j - 1$ first coordinates, and determine the j th coordinate by visiting the j -dimensional vectors \mathbf{h} .

Algorithm Dual-Space-Exploration(lattice set \mathcal{L} , weights w);

$Q \leftarrow \mathcal{N}(\mathbf{0})$ // vectors \mathbf{h} to be visited, sorted by weight $w(\mathbf{h})$;

$\mathcal{M} \leftarrow \mathcal{N}(\mathbf{0})$ // vectors \mathbf{h} who already entered Q ;

while $|\mathcal{L}| > 1$ **do**

$\mathbf{h} \leftarrow$ remove first from Q ;

for all lattices $L_s \in \mathcal{L}$ such that $\mathbf{h} \in L_s^*$ **do**

remove L_s from \mathcal{L} ;

if $|\mathcal{L}| = 1$ **then**

return the single lattice $L_s \in \mathcal{L}$ and exit;

end if

end for

for all $\mathbf{h}' \in \mathcal{N}(\mathbf{h}) \setminus \mathcal{M}$ **do**

add \mathbf{h}' to \mathcal{M} and to Q with priority (weight) $w(\mathbf{h}')$;

end for

end while

An example

Take the product V-shaped function

$$f(\mathbf{u}) = \prod_{j=1}^s \frac{|4u_j - 2| + c_j}{1 + c_j},$$

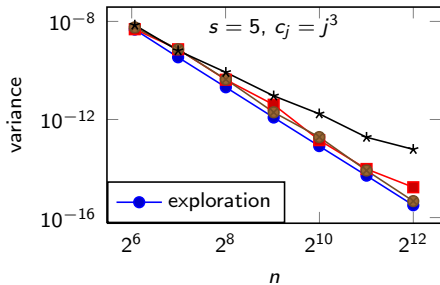
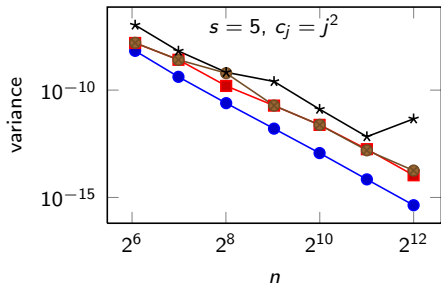
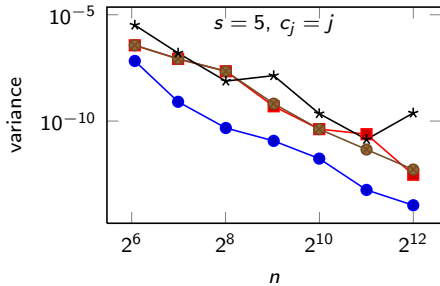
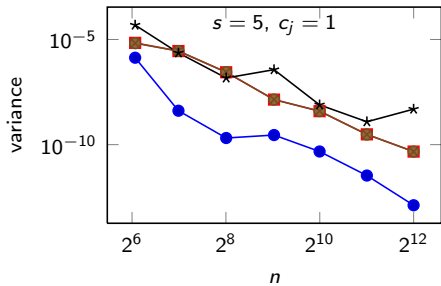
so

$$\hat{f}(\mathbf{h}) = \prod_{\{j : h_j \text{ is odd}\}} \frac{4}{(1 + c_j)\pi^2 h_j^2}.$$

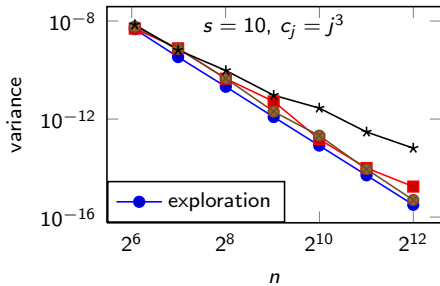
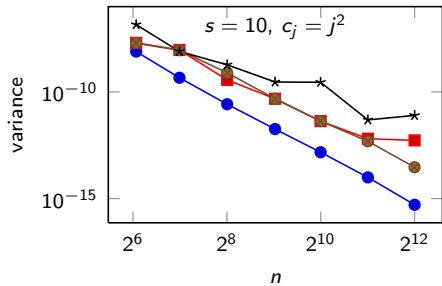
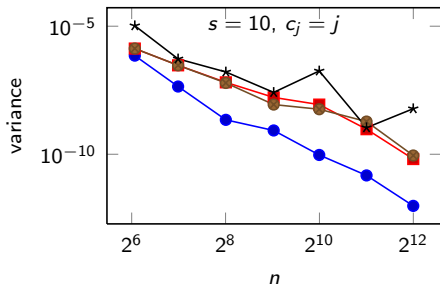
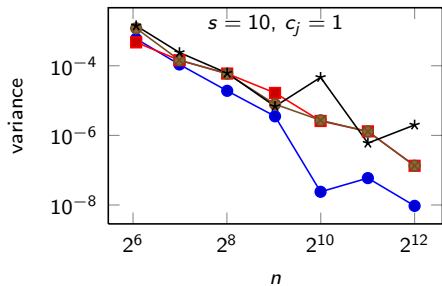
Dimensions $s = 5$ and 10.

Constants $c_j = 1, j, j^2, j^3$.

Estimated variance vs n for $s = 5$



Estimated variance vs n for $s = 10$



ANOVA decomposition

The Fourier expansion has too many terms to handle. As a cruder expansion, we can write $f(\mathbf{u}) = f(u_1, \dots, u_s)$ as:

$$f(\mathbf{u}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} f_{\mathbf{u}}(\mathbf{u}) = \mu + \sum_{i=1}^s f_{\{i\}}(u_i) + \sum_{i,j=1}^s f_{\{i,j\}}(u_i, u_j) + \dots$$

where

$$f_{\mathbf{u}}(\mathbf{u}) = \int_{[0,1]^{|\bar{\mathbf{u}}|}} f(\mathbf{u}) d\mathbf{u}_{\bar{\mathbf{u}}} - \sum_{\mathbf{v} \subset \mathbf{u}} f_{\mathbf{v}}(\mathbf{u}_{\mathbf{v}}),$$

and the Monte Carlo variance decomposes as

$$\sigma^2 = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \sigma_{\mathbf{u}}^2, \quad \text{where } \sigma_{\mathbf{u}}^2 = \text{Var}[f_{\mathbf{u}}(\mathbf{U})].$$

Sensitivity indices: $S_{\mathbf{u}} = \sigma_{\mathbf{u}}^2 / \sigma^2$. Can be estimated by MC or RQMC.

Heuristic intuition: Make sure the projections of P_n are very uniform for the important subsets \mathbf{u} (i.e., with large $S_{\mathbf{u}}$).

Shift-invariant discrepancy

In a reproducing kernel Hilbert space (RKHS) with kernel K , and **randomly-shifted** points, the relevant discrepancy corresponds to the shift-invariant kernel

$$K_{\text{sh}}(\mathbf{u}_i, \mathbf{u}_j) := \mathbb{E}[K(\mathbf{U}_i, \mathbf{U}_j)] = \mathbb{E}[K(\mathbf{u}_i + \mathbf{U}, \mathbf{u}_j + \mathbf{U})] = \mathbb{E}[K(\mathbf{u}_i - \mathbf{u}_j + \mathbf{U}, \mathbf{U})].$$

The mean square discrepancy can be written as

$$\mathbb{E}[D^2(P_n)] = \frac{1}{n} \sum_{\mathbf{0} \neq \mathbf{h} \in \mathbb{Z}^s} w(\mathbf{h}) \sum_{i=0}^{n-1} e^{2\pi \sqrt{-1} \mathbf{h}^t \mathbf{u}_i}.$$

and the corresponding square variation is

$$V^2(f) = \sum_{\mathbf{0} \neq \mathbf{h} \in \mathbb{Z}^s} \frac{|\hat{f}(\mathbf{h})|^2}{w(\mathbf{h})}.$$

Key issue: choice of the weights $w(\mathbf{h})$.

Regrouping by projections

Denote $\mathbf{u}(\mathbf{h}) = \mathbf{u}(h_1, \dots, h_s)$ the set of indices j for which $h_j \neq 0$. We have

$$\mathbb{E}[D^2(P_n)] = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \sum_{\mathbf{h}: \mathbf{u}(\mathbf{h}) = \mathbf{u}} w(\mathbf{h}) \frac{1}{n} \sum_{i=0}^{n-1} e^{2\pi \sqrt{-1} \mathbf{h}^t \mathbf{u}_i} = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} D_{\mathbf{u}}^2(P_n)$$

and

$$V^2(f) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \sum_{\mathbf{h}: \mathbf{u}(\mathbf{h}) = \mathbf{h}} \frac{|\hat{f}(\mathbf{h})|^2}{w(\mathbf{h})} = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} V^2(f_{\mathbf{u}}).$$

The RKHS decomposes as a direct sum and the RQMC variance has a corresponding decomposition. Variance bound:

$$\text{Var}[\hat{\mu}_{n, \text{rqmc}}] = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \text{Var}[\hat{\mu}_{n, \text{rqmc}}(f_{\mathbf{u}})] \leq \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} V^2(f_{\mathbf{u}}) \cdot \mathbb{E}[D_{\mathbf{u}}^2(P_n(\mathbf{u}))].$$

The mean square discrepancy for each \mathbf{u} should be weighted by $V^2(f_{\mathbf{u}})$, which is typically unknown. The problem of choosing K remains.

Example: a weighted Sobolev space

Space of functions with integrable partial derivatives. RKHS with kernel

$$K(\mathbf{u}, \mathbf{x}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} 2\pi^2 [B_2((u_j - x_j) \bmod 1)/2 + (u_j - 0.5)(x_j - 0.5)]$$

where $B_2(u) = u^2 - u + 1/6$. The **shift-invariant kernel** is

$$K_{\text{sh}}(\mathbf{u}, \mathbf{x}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} 2\pi^2 B_2((u_j - x_j) \bmod 1)$$

and the corresponding mean square discrepancy for a randomly-shifted lattice rule with $\mathbf{v}_1 = (v_1, \dots, v_s)$ is

$$\mathbb{E}[D^2(P_n)] = \frac{1}{n} \sum_{i=1}^n \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} 2\pi^2 B_2((i v_j/n) \bmod 1).$$

From the Fourier expansion of B_2 , we also have

$$\begin{aligned}\mathbb{E}[D^2(P_n)] &= \frac{1}{n} \sum_{i=1}^n \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} \sum_{h_j \neq 0} h_j^{-2} e^{2\pi\sqrt{-1}ih_j v_j/n} \\ &= \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} \gamma_{\mathbf{u}(\mathbf{h})} \prod_{j \in \mathbf{u}(\mathbf{h})} h_j^2 = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} w(\mathbf{h}).\end{aligned}$$

For those weights, we have $w(\mathbf{h}) = |\hat{f}(\mathbf{h})|^2$ for the function

$$f(\mathbf{u}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} (2\pi)^{|\mathbf{u}|} \gamma_{\mathbf{u}}^{1/2} \prod_{j \in \mathbf{u}} B_1(u_j),$$

so $\mathbb{E}[D^2(P_n)]$ is the RQMC variance for this f .

On the other hand, the ANOVA variance components for this f are

$$\sigma_{\mathbf{u}}^2 = (4\pi^2)^{|\mathbf{u}|} \gamma_{\mathbf{u}} \prod_{j \in \mathbf{u}} \text{Var}[B_1(U)] = (3/\pi^2)^{-|\mathbf{u}|} \gamma_{\mathbf{u}},$$

because $\text{Var}[B_1(U)] = 1/12$. The optimal weights are then

$$\gamma_{\mathbf{u}} = (3/\pi^2)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2 \approx (0.30396)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2,$$

if we believe that this worst-case function is representative of our f .

Using the same kernel and a different heuristic argument, [Wang and Sloan \(2006\)](#) come up with weights that generalize to (they do this for product weights only):

$$\gamma_u^2 = (45/\pi^4)^{|u|} \sigma_u^2,$$

that is,

$$\gamma_u = (\sqrt{45}/\pi^2)^{|u|} \sigma_u \approx 0.6797^{|u|} \sigma_u,$$

Using the same kernel and a different heuristic argument, [Wang and Sloan \(2006\)](#) come up with weights that generalize to (they do this for product weights only):

$$\gamma_u^2 = (45/\pi^4)^{|u|} \sigma_u^2,$$

that is,

$$\gamma_u = (\sqrt{45}/\pi^2)^{|u|} \sigma_u \approx 0.6797^{|u|} \sigma_u,$$

With $\gamma_u = 1$, we obtain the [classical \(unweighted\) \$\mathcal{P}_2\$](#) .

Weighted $\mathcal{P}_{2\alpha}$:

$$\mathcal{P}_{2\alpha} = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} \gamma_{\mathbf{u}(\mathbf{h})} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}$$

Variance for a worst-case function whose square Fourier coefficients are

$$|\hat{f}(\mathbf{h})|^2 = \gamma_{\mathbf{u}(\mathbf{h})} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}.$$

This is the RQMC variance for a function of the form

$$f(\mathbf{u}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \sqrt{\gamma_{\mathbf{u}}} \prod_{j \in \mathbf{u}} \frac{(2\pi)^\alpha}{\alpha!} B_\alpha(u_j).$$

We also have

$$\sigma_{\mathbf{u}}^2 = \gamma_{\mathbf{u}} \left[\text{Var}[B_\alpha(U)] \frac{(4\pi^2)^\alpha}{(\alpha!)^2} \right]^{|\mathbf{u}|} = \gamma_{\mathbf{u}} \left[|B_{2\alpha}(0)| \frac{(4\pi^2)^\alpha}{(2\alpha)!} \right]^{|\mathbf{u}|}.$$

For $\alpha = 1$, we should take $\gamma_{\mathbf{u}} = (3/\pi^2)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2 \approx (0.30396)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2$.

For $\alpha = 2$, we should take $\gamma_{\mathbf{u}} = [45/\pi^4]^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2 \approx (0.46197)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2$.

Weighted $\mathcal{P}_{2\alpha}$:

$$\mathcal{P}_{2\alpha} = \sum_{\mathbf{0} \neq \mathbf{h} \in L_s^*} \gamma_{\mathbf{u}(\mathbf{h})} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}$$

Variance for a worst-case function whose square Fourier coefficients are

$$|\hat{f}(\mathbf{h})|^2 = \gamma_{\mathbf{u}(\mathbf{h})} (\max(1, h_1), \dots, \max(1, h_s))^{-2\alpha}.$$

This is the RQMC variance for a function of the form

$$f(\mathbf{u}) = \sum_{\mathbf{u} \subseteq \{1, \dots, s\}} \sqrt{\gamma_{\mathbf{u}}} \prod_{j \in \mathbf{u}} \frac{(2\pi)^\alpha}{\alpha!} B_\alpha(u_j).$$

We also have

$$\sigma_{\mathbf{u}}^2 = \gamma_{\mathbf{u}} \left[\text{Var}[B_\alpha(U)] \frac{(4\pi^2)^\alpha}{(\alpha!)^2} \right]^{|\mathbf{u}|} = \gamma_{\mathbf{u}} \left[|B_{2\alpha}(0)| \frac{(4\pi^2)^\alpha}{(2\alpha)!} \right]^{|\mathbf{u}|}.$$

For $\alpha = 1$, we should take $\gamma_{\mathbf{u}} = (3/\pi^2)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2 \approx (0.30396)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2$.

For $\alpha = 2$, we should take $\gamma_{\mathbf{u}} = [45/\pi^4]^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2 \approx (0.46197)^{|\mathbf{u}|} \sigma_{\mathbf{u}}^2$.

The ratios **weight / variance** should decrease exponentially with $|\mathbf{u}|$.

Heuristics for choosing the weights

Idea 1: take $\gamma_u \approx \sigma_u^2$ or $\gamma_u \approx S_u$ for each u . Too simplistic.

The weight γ_u should be proportional to $V^2(f_u)$, which is generally not proportional to σ_u^2 .

Heuristics for choosing the weights

Idea 1: take $\gamma_u \approx \sigma_u^2$ or $\gamma_u \approx S_u$ for each u . Too simplistic.

The weight γ_u should be proportional to $V^2(f_u)$, which is generally not proportional to σ_u^2 .

Idea 2: try to have γ_u proportional to $V^2(f_u)$ for each u . With a weighted \mathcal{P}_α -type criterion, this gives $\gamma_u = \rho^{|u|} \sigma_u^2$ for a constant ρ .

In general, one can define a simple parametric model for the square variations and then estimate the parameters by matching the ANOVA variances (e.g., Wang and Sloan 2006).

For example, $\gamma_u = \prod_{j \in u} \gamma_j$ for some constants $\gamma_j \geq 0$ (product weights).

Fewer parameters: take $\gamma_j = a\beta^j$ for $a, \beta > 0$ (geometric).

Heuristics for choosing the weights

Idea 1: take $\gamma_u \approx \sigma_u^2$ or $\gamma_u \approx S_u$ for each u . Too simplistic.

The weight γ_u should be proportional to $V^2(f_u)$, which is generally not proportional to σ_u^2 .

Idea 2: try to have γ_u proportional to $V^2(f_u)$ for each u . With a weighted \mathcal{P}_α -type criterion, this gives $\gamma_u = \rho^{|u|} \sigma_u^2$ for a constant ρ .

In general, one can define a simple parametric model for the square variations and then estimate the parameters by matching the ANOVA variances (e.g., Wang and Sloan 2006).

For example, $\gamma_u = \prod_{j \in u} \gamma_j$ for some constants $\gamma_j \geq 0$ (product weights).

Fewer parameters: take $\gamma_j = a\beta^j$ for $a, \beta > 0$ (geometric).

Idea 3: Just take simple order-dependent weights. For example, $\gamma_u = 1$ for $|u| \leq d$ and $\gamma_u = 0$ otherwise. Wang (2007) recommends this with $d = 2$.

Proposal 4: A strategy for order-dependent weights.

Assume $\gamma_{\mathbf{u}} = \Gamma_{|\mathbf{u}|}$. Need to select $\Gamma_1, \dots, \Gamma_s$.

For each \mathbf{u} , let $v_{\mathbf{u}}^2$ be an estimate of the square variation $V^2(f_{\mathbf{u}})$.

Strategy: take Γ_r as the average

$$\Gamma_r = \binom{s}{r}^{-1} \sum_{\{\mathbf{u}:|\mathbf{u}|=r\}} v_{\mathbf{u}}^2.$$

Here, the scaling of weights does not matter.

Proposal 5: A strategy for product weights.

Ignore **one-dimensional** projections; they are the same for all lattices.

The idea is to fit the estimated square variations over all **two-dimensional** projections via a least-squares procedure. Then we rescale all the weights by a constant factor to match the ratio of average square variations over the **three-dimensional** projections to that over the two-dimensional projections.

Let τ_j be the unscaled weight for projection j . We first minimize

$$R = \sum_{k=1}^s \sum_{j=1}^{k-1} \left(\tau_j \tau_k - v_{\{j,k\}}^2 \right)^2.$$

Differentiating w.r.t. τ_j and equating to 0, we obtain, for each j ,

$$\tau_j \sum_{k=1, k \neq j}^s \tau_k^2 = \sum_{k=1, k \neq j}^s \tau_k v_{\{j,k\}}^2.$$

This can be solved by an iterative fixed-point algorithm:

$$\tau_j^{(0)} = \max_{k,l=1,\dots,s} v_{\{k,l\}}, \quad \tau_j^{(i+1)} = \frac{\sum_{k=1, k \neq j}^s \tau_k^{(i)} v_{\{j,k\}}^2}{\sum_{k=1, k \neq j}^s \left(\tau_k^{(i)}\right)^2},$$

for $i = 1, 2, \dots$.

We then rescale the weights via $\gamma_j = c\tau_j$ where the constant c satisfies

$$\frac{\sum_{k=1}^s \sum_{j=1}^{k-1} \tau_j \tau_k}{c \sum_{k=1}^s \sum_{j=1}^{k-1} \sum_{l=1}^{j-1} \tau_j \tau_k \tau_l} = \frac{\sum_{k=1}^s \sum_{j=1}^{k-1} v_{\{j,k\}}^2}{\sum_{k=1}^s \sum_{j=1}^{k-1} \sum_{l=1}^{j-1} v_{\{j,k,l\}}^2}.$$

Idea 6: Control the shortest vector in dual lattice, for each projection.

Spectral test for LCGs (Knuth, etc.):

$$\min_{2 \leq r \leq t_1} \frac{\ell_{\{1, \dots, r\}}}{\ell_r^*(n)}$$

where $\ell_{\mathbf{u}}$ is the length of a shortest vector in $L^*(\mathbf{u})$ and $\ell_r^*(n)$ is a theoretical upper bound on this length, in r dimensions.

Advantages: Computing time of $\ell_{\mathbf{u}}$ are almost independent of n , although exponential in $|\mathbf{u}|$. Poor lattices can be eliminated quickly: search is fast.

Idea 6: Control the shortest vector in dual lattice, for each projection.

Lemieux and L'Ecuyer (2000, etc.) maximize

$$M_{t_1, \dots, t_d} = \min \left[\min_{2 \leq r \leq t_1} \frac{\ell_{\{1, \dots, r\}}}{\ell_r^*(n)}, \min_{2 \leq r \leq d} \min_{\substack{u = \{j_1, \dots, j_r\} \subset \{1, \dots, s\} \\ 1 = j_1 < \dots < j_r \leq t_r}} \frac{\ell_u}{\ell_r^*(n)} \right],$$

where ℓ_u is the length of a shortest vector in $L^*(u)$ and $\ell_r^*(n)$ is a theoretical upper bound on this length, in r dimensions.

Advantages: Computing time of ℓ_u are almost independent of n , although exponential in $|u|$. Poor lattices can be eliminated quickly: search is fast.

This can of course be generalized by adding weights to projections.

Searching for lattice parameters

Korobov lattices. Search over all admissible integers a , for $\mathbf{v}_1 = (1, a, a^2, \dots, \dots)/n$.

Searching for lattice parameters

Korobov lattices. Search over all admissible integers a , for $\mathbf{v}_1 = (1, a, a^2, \dots, \dots)/n$.

Component by component (CBC) construction.

Let $v_{1,1} = 1/n$;

For $j = 2, 3, \dots, s$, find $z \in \{1, \dots, n-1\}$, $\gcd(z, n) = 1$, such that $(v_{1,1}, v_{1,2}, \dots, v_{1,j} = z/n)$ minimizes the selected discrepancy for the first j dimensions.

Searching for lattice parameters

Korobov lattices. Search over all admissible integers a , for $\mathbf{v}_1 = (1, a, a^2, \dots, \dots)/n$.

Component by component (CBC) construction.

Let $v_{1,1} = 1/n$;

For $j = 2, 3, \dots, s$, find $z \in \{1, \dots, n-1\}$, $\gcd(z, n) = 1$, such that $(v_{1,1}, v_{1,2}, \dots, v_{1,j} = z/n)$ minimizes the selected discrepancy for the first j dimensions.

Partial randomized CBC construction.

Let $v_{1,1} = 1/n$;

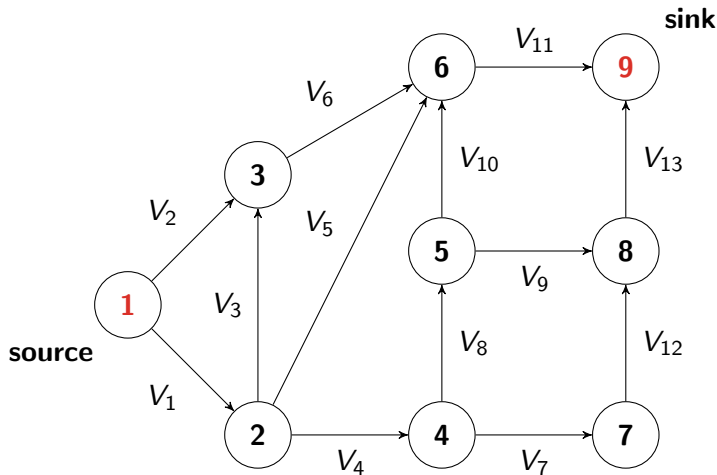
For $j = 2, 3, \dots, s$, try r random $z \in \{1, \dots, n-1\}$, $\gcd(z, n) = 1$, and retain the one that $(v_{1,1}, v_{1,2}, \dots, v_{1,j} = z/n)$ minimizes the selected discrepancy for the first j dimensions.

Example: stochastic activity network

Each arc j has random length $V_j = F_j^{-1}(U_j)$.

Let $T = f(U_1, \dots, U_{13}) =$ length of longest path from node 1 to node 9.

Want to estimate $q(x) = \mathbb{P}[T > x]$ for a given constant x .



To estimate $q(x)$ by **MC**, we generate n independent realizations of T , say T_1, \dots, T_n , and $(1/n) \sum_{i=1}^n \mathbb{I}[T_i > x]$.

For **RQMC**, we replace the n realizations of (U_1, \dots, U_{13}) by the n points of a randomly-shifted lattice.

To estimate $q(x)$ by **MC**, we generate n independent realizations of T , say T_1, \dots, T_n , and $(1/n) \sum_{i=1}^n \mathbb{I}[T_i > x]$.

For **RQMC**, we replace the n realizations of (U_1, \dots, U_{13}) by the n points of a randomly-shifted lattice.

CMC estimator. Generate the V_j 's only for the 8 arcs that **do not** belong to the cut $\mathcal{L} = \{5, 6, 7, 9, 10\}$, and replace $\mathbb{I}[T > x]$ by its **conditional expectation** given those V_j 's, $\mathbb{P}[T > x \mid \{V_j, j \notin \mathcal{L}\}]$.

This makes the integrand **continuous** in the U_j 's.

To estimate $q(x)$ by **MC**, we generate n independent realizations of T , say T_1, \dots, T_n , and $(1/n) \sum_{i=1}^n \mathbb{I}[T_i > x]$.

For **RQMC**, we replace the n realizations of (U_1, \dots, U_{13}) by the n points of a randomly-shifted lattice.

CMC estimator. Generate the V_j 's only for the 8 arcs that **do not** belong to the cut $\mathcal{L} = \{5, 6, 7, 9, 10\}$, and replace $\mathbb{I}[T > x]$ by its **conditional expectation** given those V_j 's, $\mathbb{P}[T > x \mid \{V_j, j \notin \mathcal{L}\}]$.

This makes the integrand **continuous** in the U_j 's.

Illustration: $V_j \sim \text{Normal}(\mu_j, \sigma_j^2)$ for $j = 1, 2, 4, 11, 12$, and $V_j \sim \text{Exponential}(1/\mu_j)$ otherwise.

The μ_j : 13.0, 5.5, 7.0, 5.2, 16.5, 14.7, 10.3, 6.0, 4.0, 20.0, 3.2, 3.2, 16.5.

ANOVA decomposition

– Show quickly the full table of ANOVA variances, in separate text file.
Then summarize by cardinality of projections.

There are six paths from 1 to 9:

$$\mathcal{P} = \{\{1, 5, 11\}, \{2, 6, 11\}, \{1, 3, 6, 11\}, \{1, 4, 7, 12, 13\}, \{1, 4, 8, 9, 13\}, \{1, 4, 8, 10, 11\}\}$$

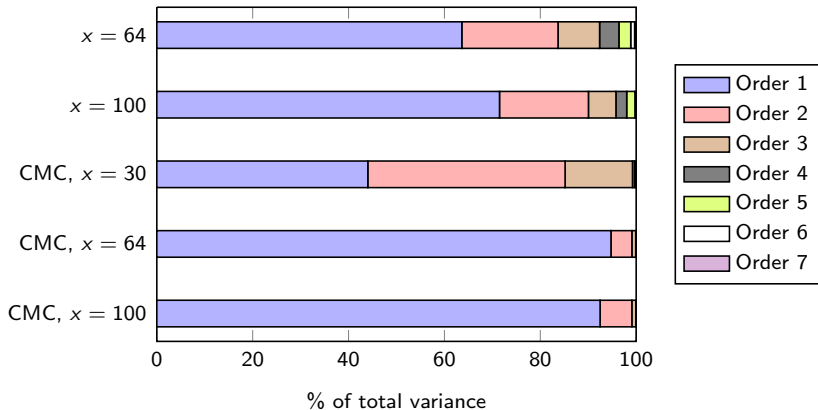
We may think that the important projections are only the subsets of those paths:

$$\mathcal{P}' = \{\mathbf{v} \subseteq \mathbf{u} \in \mathcal{P}\}.$$

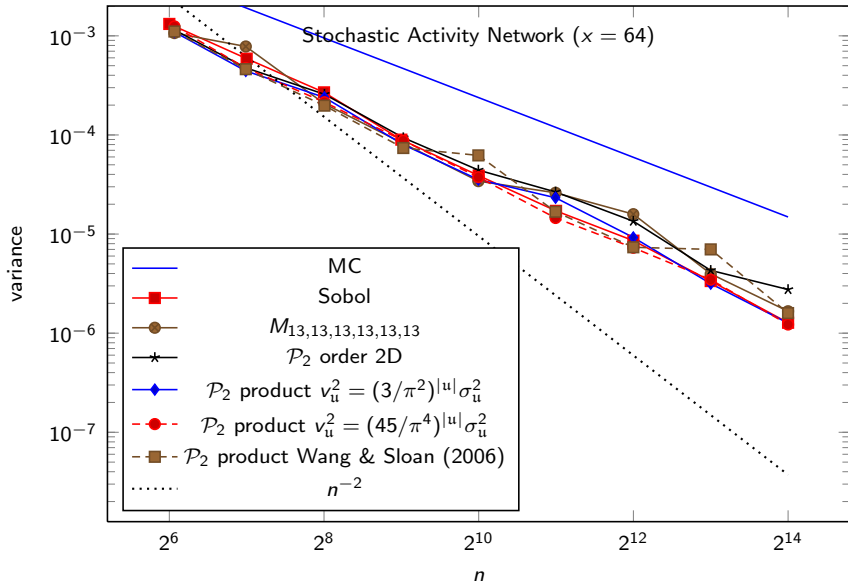
Fraction of the total variance that lies in these projections:

	$x = 30$	$x = 64$	$x = 100$
crude MC		80.6 %	96.3 %
conditional MC	88.8 %	99.5 %	100 %

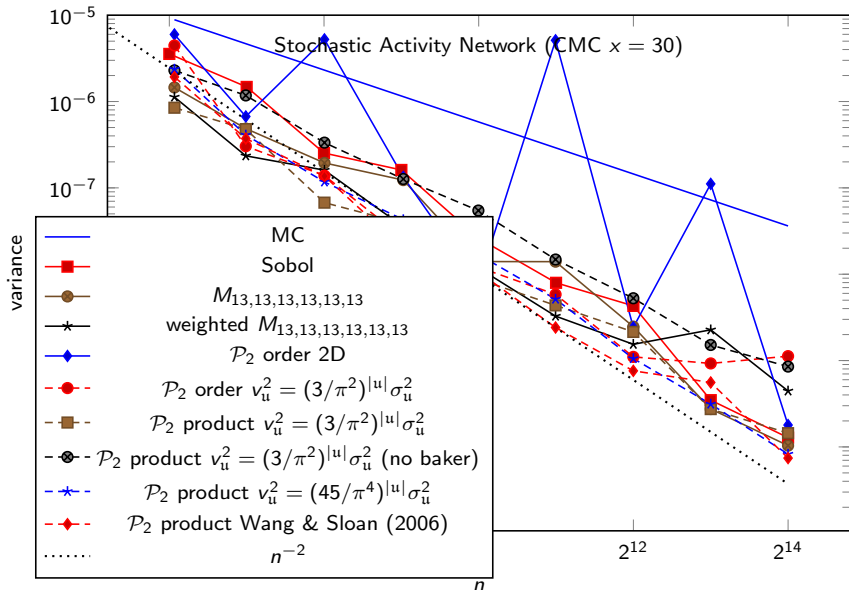
ANOVA Variances for the Stochastic Activity Network



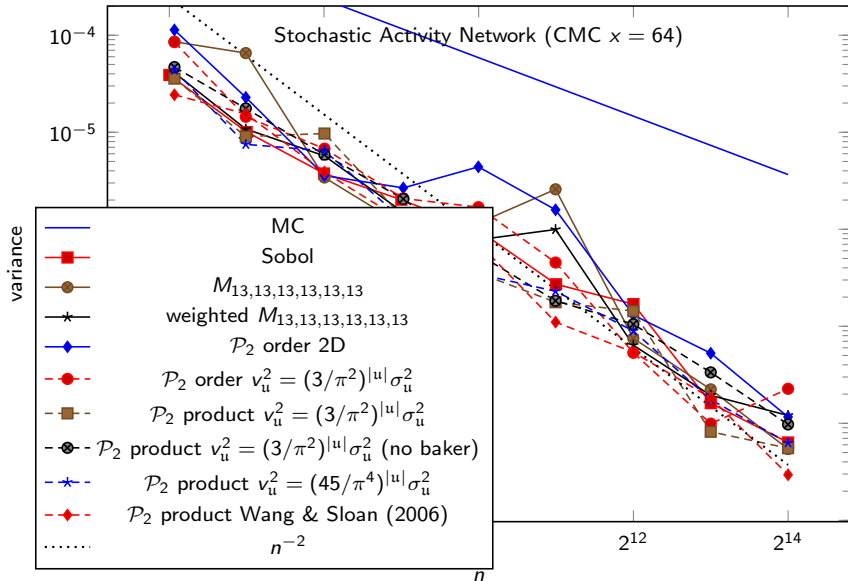
Lattices of Rank 1 with CBC



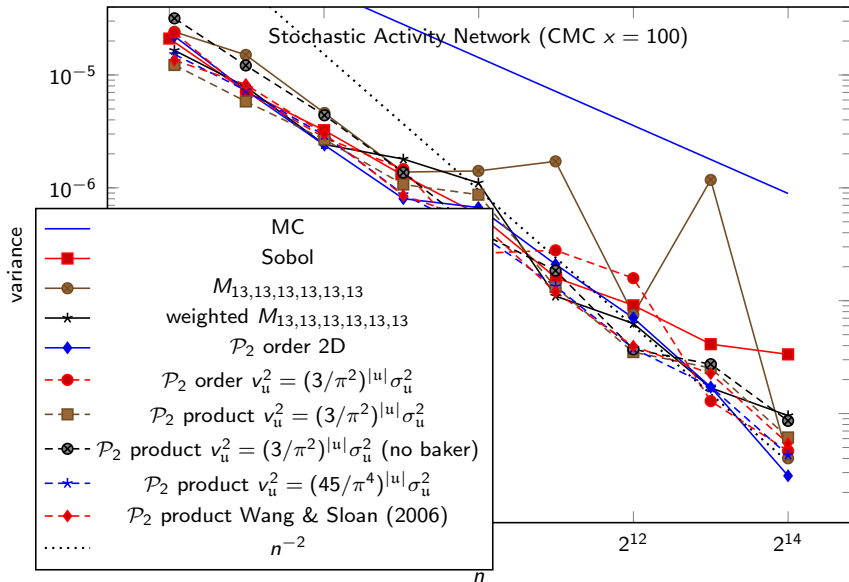
Lattices of Rank 1 with CBC



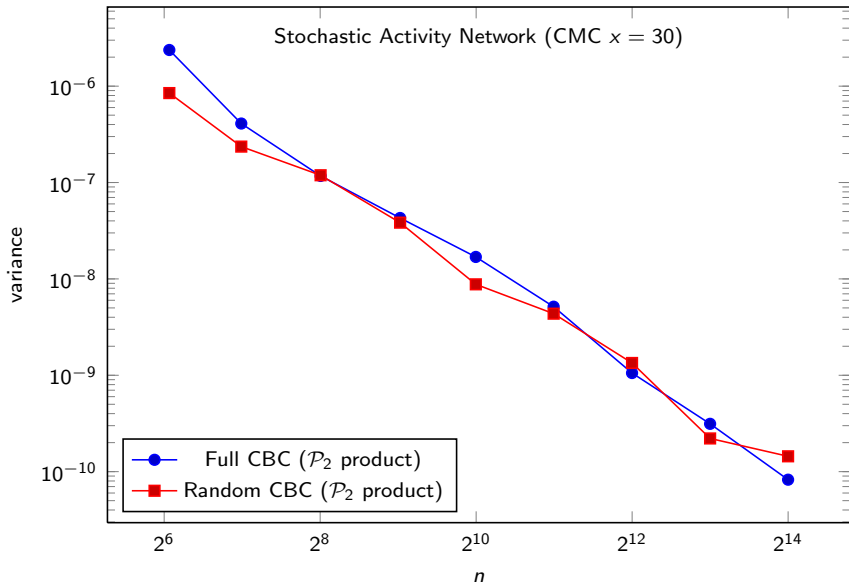
Lattices of Rank 1 with CBC



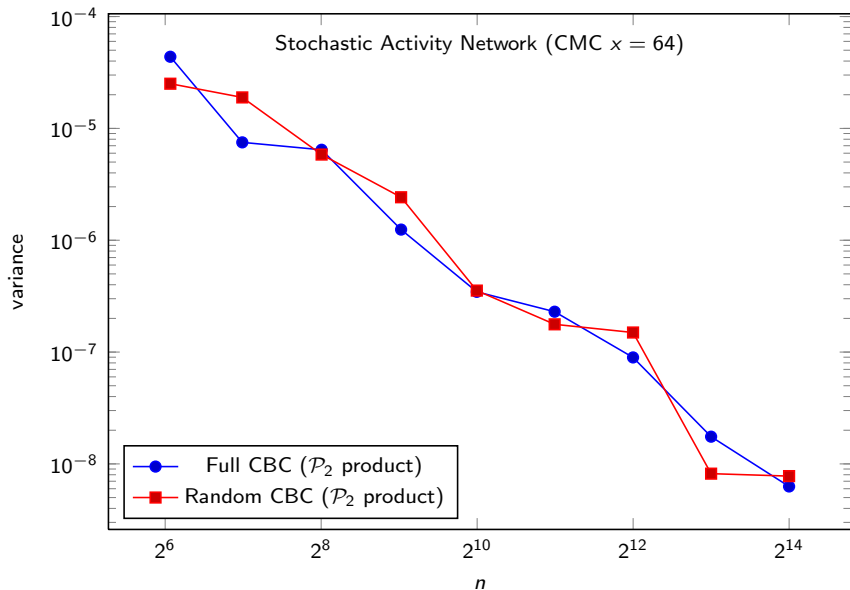
Lattices of Rank 1 with CBC



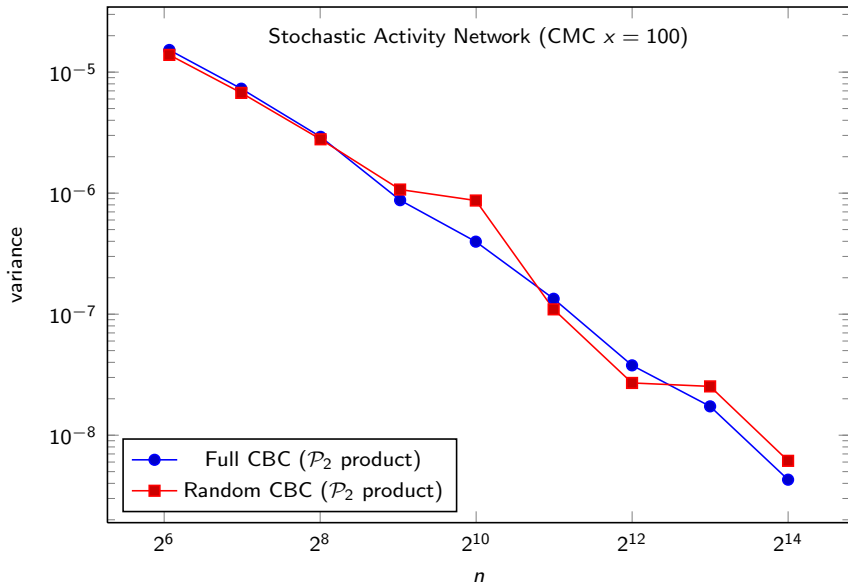
Random vs. Full CBC



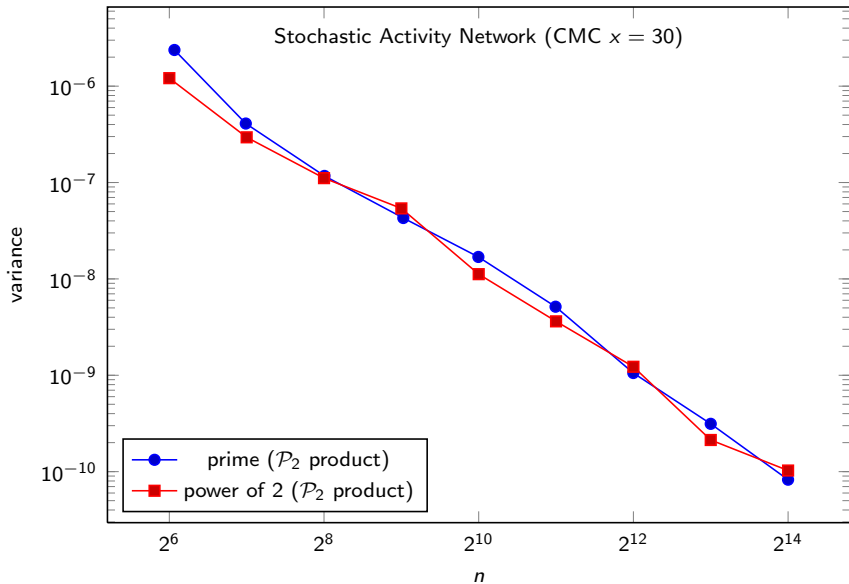
Random vs. Full CBC



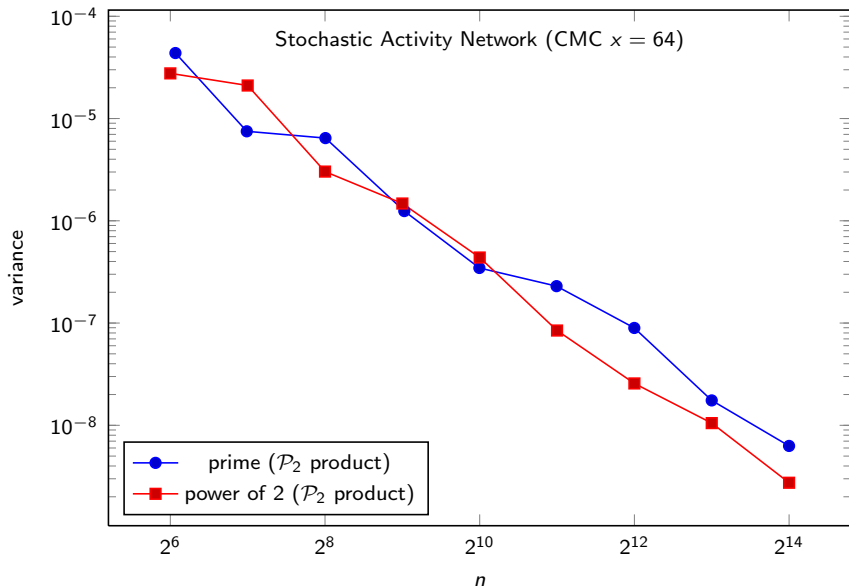
Random vs. Full CBC



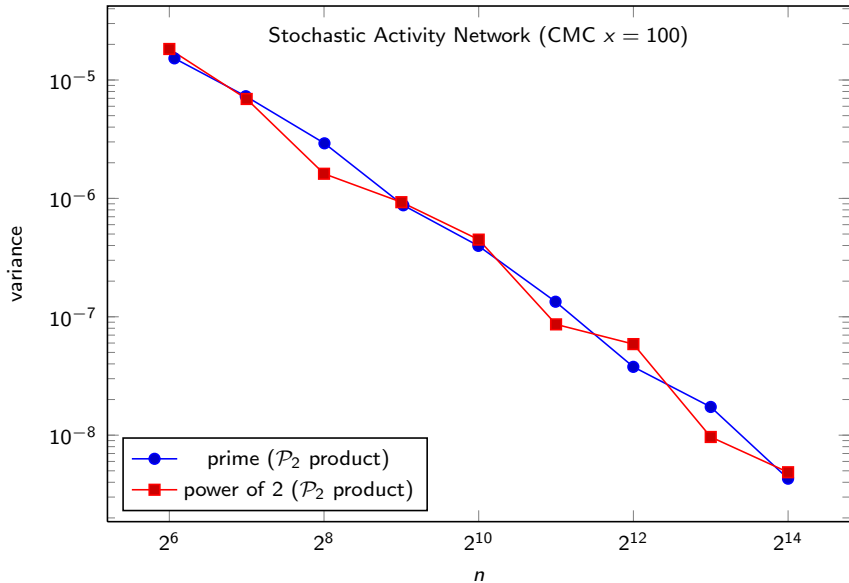
Prime vs. Power-of-2 Number of Points



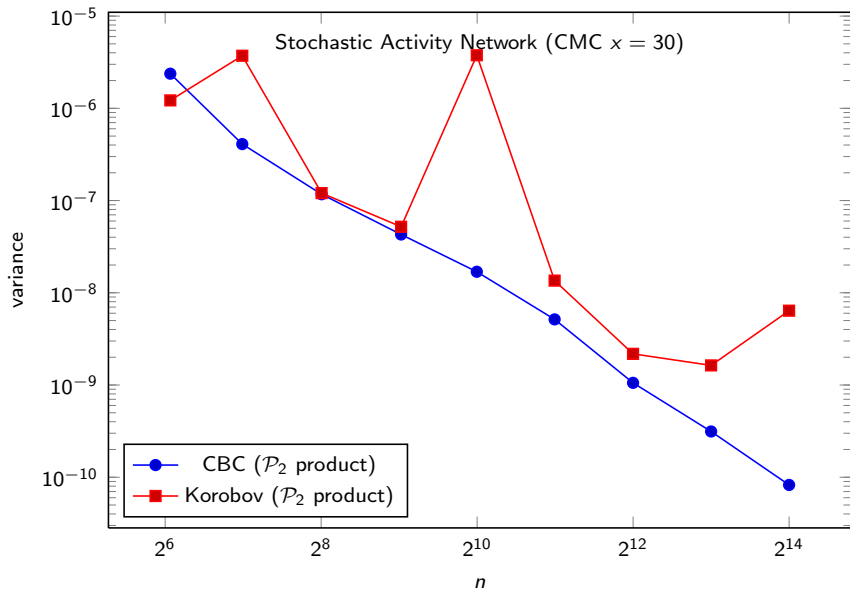
Prime vs. Power-of-2 Number of Points



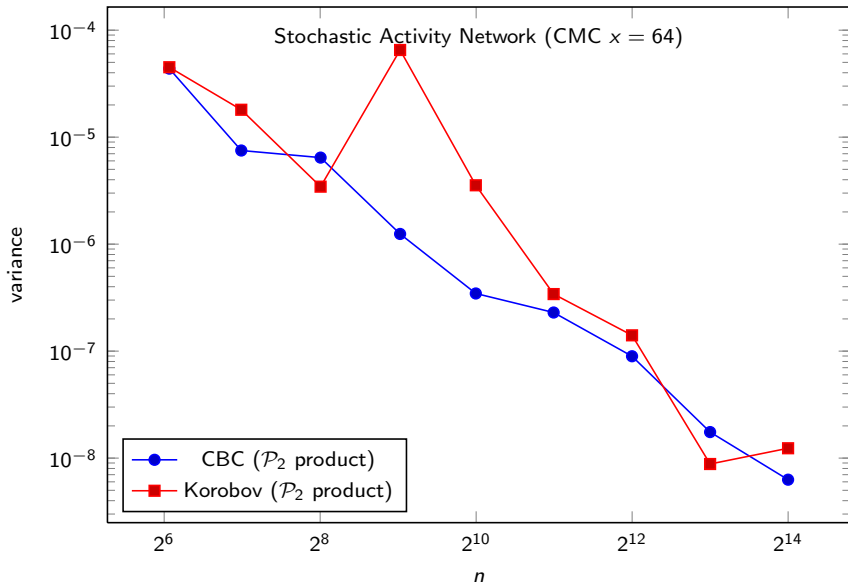
Prime vs. Power-of-2 Number of Points



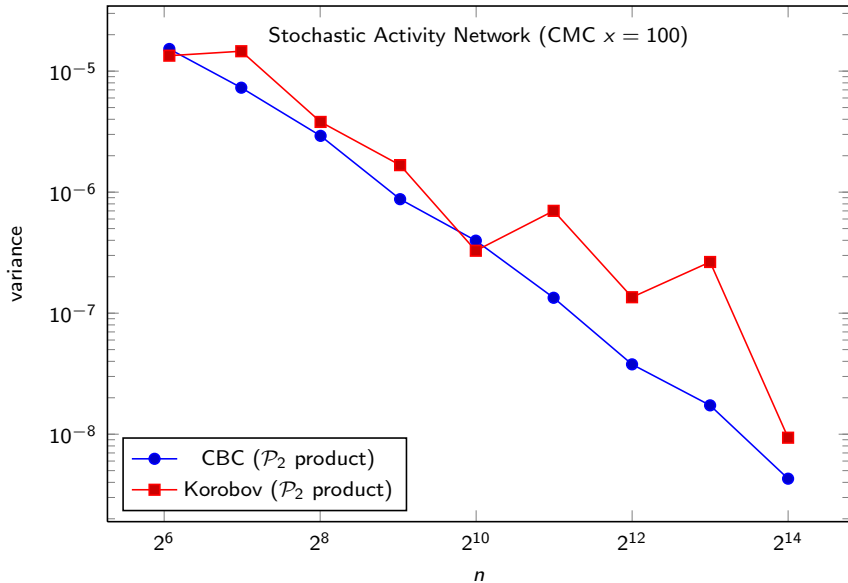
Korobov vs. CBC



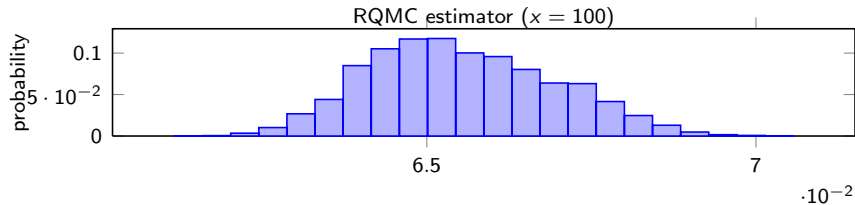
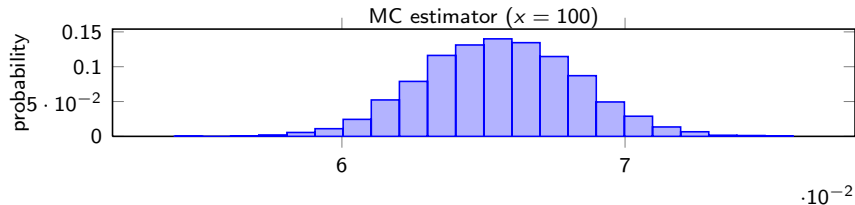
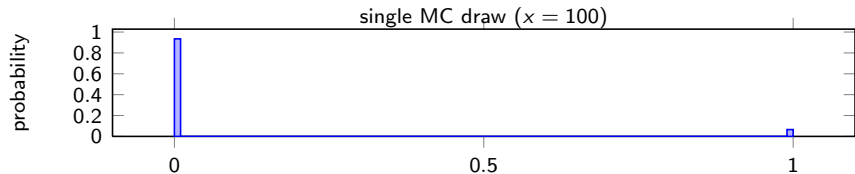
Korobov vs. CBC



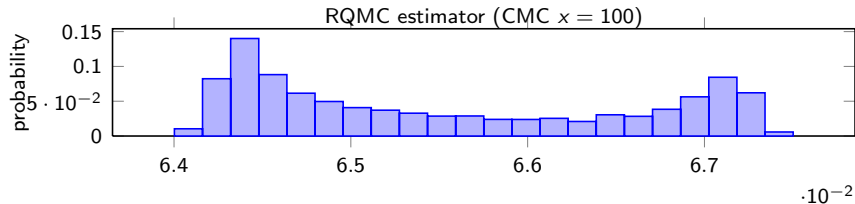
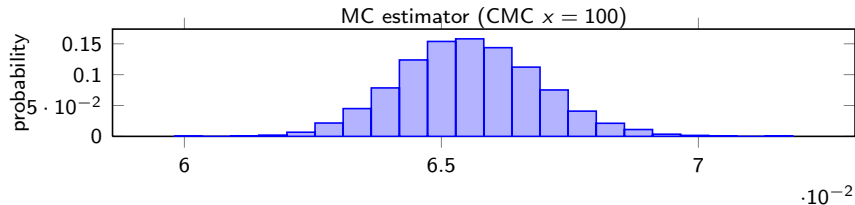
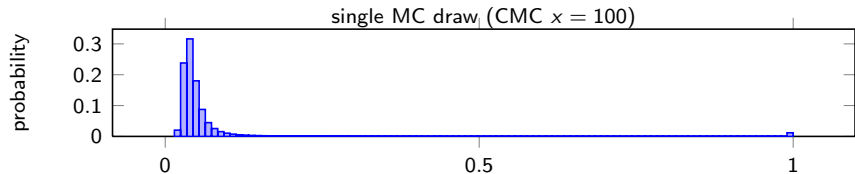
Korobov vs. CBC



Histograms



Histograms



Function of a Multinormal vector

Let $\mu = E[f(\mathbf{U})] = E[g(\mathbf{Y})]$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

Function of a Multinormal vector

Let $\mu = E[f(\mathbf{U})] = E[g(\mathbf{Y})]$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

For example, if the payoff of a financial derivative is a function of the values taken by a c -dimensional geometric Brownian motions (GMB) at d observations times $0 < t_1 < \dots < t_d = T$, then we have $s = cd$.

Function of a Multinormal vector

Let $\mu = E[f(\mathbf{U})] = E[g(\mathbf{Y})]$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

For example, if the payoff of a financial derivative is a function of the values taken by a c -dimensional geometric Brownian motions (GMB) at d observations times $0 < t_1 < \dots < t_d = T$, then we have $s = cd$.

To generate \mathbf{Y} : Decompose $\mathbf{\Sigma} = \mathbf{A}\mathbf{A}^t$, generate $\mathbf{Z} = (Z_1, \dots, Z_s) = (\Phi^{-1}(U_1), \dots, \Phi^{-1}(U_s)) \sim N(\mathbf{0}, \mathbf{I})$ and return $\mathbf{Y} = \mathbf{A}\mathbf{Z}$.

Function of a Multinormal vector

Let $\mu = E[f(\mathbf{U})] = E[g(\mathbf{Y})]$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

For example, if the payoff of a financial derivative is a function of the values taken by a c -dimensional geometric Brownian motions (GMB) at d observations times $0 < t_1 < \dots < t_d = T$, then we have $s = cd$.

To generate \mathbf{Y} : Decompose $\mathbf{\Sigma} = \mathbf{A}\mathbf{A}^t$, generate $\mathbf{Z} = (Z_1, \dots, Z_s) = (\Phi^{-1}(U_1), \dots, \Phi^{-1}(U_s)) \sim N(\mathbf{0}, \mathbf{I})$ and return $\mathbf{Y} = \mathbf{A}\mathbf{Z}$.

Choice of \mathbf{A} ?

Function of a Multinormal vector

Let $\mu = E[f(\mathbf{U})] = E[g(\mathbf{Y})]$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

For example, if the payoff of a financial derivative is a function of the values taken by a c -dimensional geometric Brownian motions (GMB) at d observations times $0 < t_1 < \dots < t_d = T$, then we have $s = cd$.

To generate \mathbf{Y} : Decompose $\mathbf{\Sigma} = \mathbf{A}\mathbf{A}^t$, generate $\mathbf{Z} = (Z_1, \dots, Z_s) = (\Phi^{-1}(U_1), \dots, \Phi^{-1}(U_s)) \sim N(\mathbf{0}, \mathbf{I})$ and return $\mathbf{Y} = \mathbf{A}\mathbf{Z}$.

Choice of \mathbf{A} ?

Cholesky factorization: \mathbf{A} is lower triangular.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Function of a Brownian motion:

Payoff depends on c -dimensional Brownian motion $\{\mathbf{X}(t), t \geq 0\}$ observed at times $0 = t_0 < t_1 < \dots < t_d$.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Function of a Brownian motion:

Payoff depends on c -dimensional Brownian motion $\{\mathbf{X}(t), t \geq 0\}$ observed at times $0 = t_0 < t_1 < \dots < t_d$.

Sequential (or random walk) method: generate $\mathbf{X}(t_1)$, then $\mathbf{X}(t_2) - \mathbf{X}(t_1)$, then $\mathbf{X}(t_3) - \mathbf{X}(t_2)$, etc.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Function of a Brownian motion:

Payoff depends on c -dimensional Brownian motion $\{\mathbf{X}(t), t \geq 0\}$ observed at times $0 = t_0 < t_1 < \dots < t_d$.

Sequential (or random walk) method: generate $\mathbf{X}(t_1)$, then $\mathbf{X}(t_2) - \mathbf{X}(t_1)$, then $\mathbf{X}(t_3) - \mathbf{X}(t_2)$, etc.

Brownian bridge (BB) sampling: Suppose $d = 2^m$.

Generate $\mathbf{X}(t_d)$, then $\mathbf{X}(t_{d/2})$ conditional on $(\mathbf{X}(0), \mathbf{X}(t_d))$,

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Function of a Brownian motion:

Payoff depends on c -dimensional Brownian motion $\{\mathbf{X}(t), t \geq 0\}$ observed at times $0 = t_0 < t_1 < \dots < t_d$.

Sequential (or random walk) method: generate $\mathbf{X}(t_1)$, then $\mathbf{X}(t_2) - \mathbf{X}(t_1)$, then $\mathbf{X}(t_3) - \mathbf{X}(t_2)$, etc.

Brownian bridge (BB) sampling: Suppose $d = 2^m$.

Generate $\mathbf{X}(t_d)$, then $\mathbf{X}(t_{d/2})$ conditional on $(\mathbf{X}(0), \mathbf{X}(t_d))$, then $\mathbf{X}(t_{d/4})$ conditional on $(\mathbf{X}(0), \mathbf{X}(t_{d/2}))$, and so on.

The first few $N(0, 1)$ r.v.'s already sketch the path trajectory.

Principal component decomposition (PCA):

$\mathbf{A} = \mathbf{P}\mathbf{D}^{1/2}$ where $\mathbf{D} = \text{diag}(\lambda_s, \dots, \lambda_1)$ (eigenvalues of $\mathbf{\Sigma}$ in decreasing order) and the columns of \mathbf{P} are the corresponding unit-length eigenvectors. With this \mathbf{A} , Z_1 accounts for the maximum amount of variance of \mathbf{Y} , then Z_2 for the maximum amount of variance conditional on Z_1 , and so on.

Function of a Brownian motion:

Payoff depends on c -dimensional Brownian motion $\{\mathbf{X}(t), t \geq 0\}$ observed at times $0 = t_0 < t_1 < \dots < t_d$.

Sequential (or random walk) method: generate $\mathbf{X}(t_1)$, then $\mathbf{X}(t_2) - \mathbf{X}(t_1)$, then $\mathbf{X}(t_3) - \mathbf{X}(t_2)$, etc.

Brownian bridge (BB) sampling: Suppose $d = 2^m$.

Generate $\mathbf{X}(t_d)$, then $\mathbf{X}(t_{d/2})$ conditional on $(\mathbf{X}(0), \mathbf{X}(t_d))$, then $\mathbf{X}(t_{d/4})$ conditional on $(\mathbf{X}(0), \mathbf{X}(t_{d/2}))$, and so on.

The first few $N(0, 1)$ r.v.'s already sketch the path trajectory.

Each of these methods corresponds to some matrix \mathbf{A} .

Choice has large impact on the ANOVA decomposition of f .

Example: Pricing an Asian option

Single asset, s observation times t_1, \dots, t_s . Want to estimate $\mathbb{E}[f(\mathbf{U})]$, where

$$f(\mathbf{U}) = e^{-rt_s} \max \left[0, \frac{1}{s} \sum_{j=1}^s S(t_j) - K \right]$$

and $\{S(t), t \geq 0\}$ is a geometric Brownian motion.

We have $f(\mathbf{U}) = g(\mathbf{Y})$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

Example: Pricing an Asian option

Single asset, s observation times t_1, \dots, t_s . Want to estimate $\mathbb{E}[f(\mathbf{U})]$, where

$$f(\mathbf{U}) = e^{-rt_s} \max \left[0, \frac{1}{s} \sum_{j=1}^s S(t_j) - K \right]$$

and $\{S(t), t \geq 0\}$ is a geometric Brownian motion.

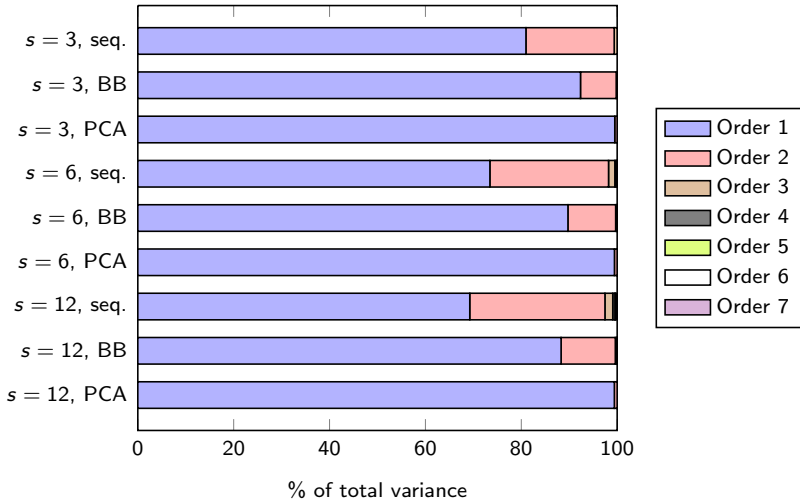
We have $f(\mathbf{U}) = g(\mathbf{Y})$ where $\mathbf{Y} = (Y_1, \dots, Y_s) \sim N(\mathbf{0}, \mathbf{\Sigma})$.

Let $S(0) = 100$, $K = 100$, $r = 0.05$, $t_s = 1$, and $t_j = jT/s$ for $1 \leq j \leq s$.

We consider $\sigma = 0.2, 0.5$ and $s = 3, 6, 12$.

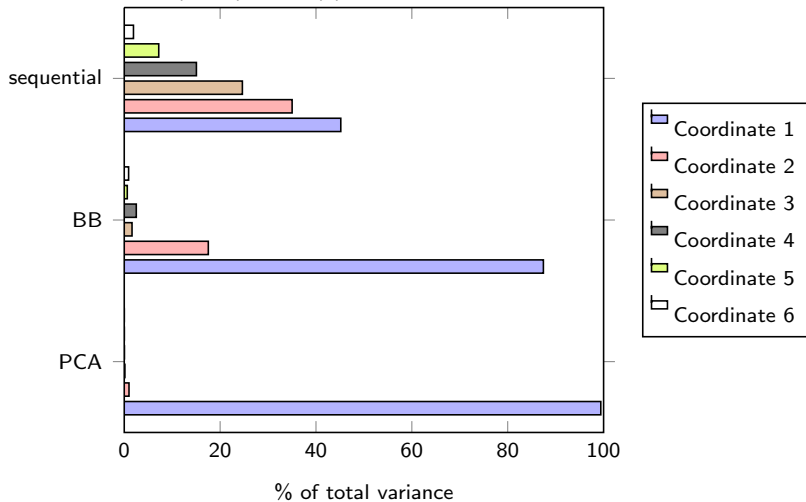
ANOVA Variances for the Asian Option

Asian Option with $S(0) = 100$, $K = 100$, $r = 0.05$, $\sigma = 0.5$

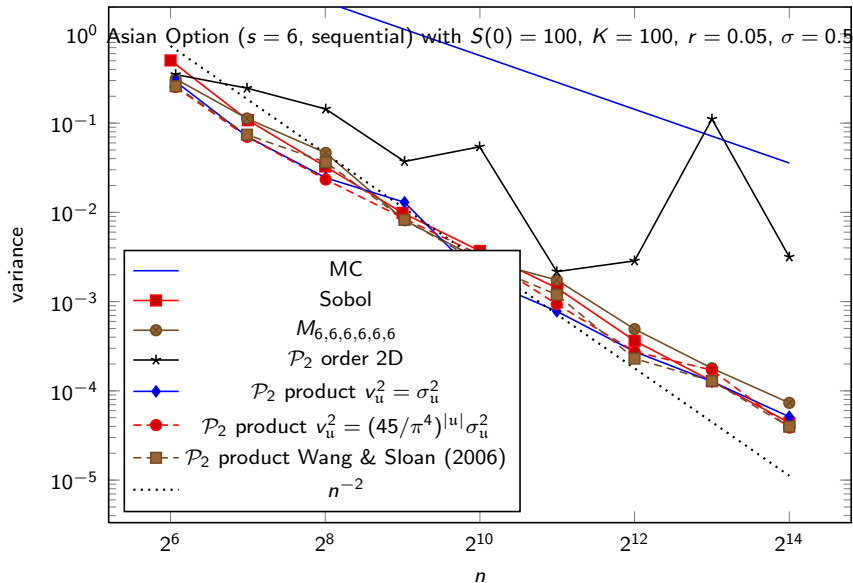


Total Variance per Coordinate for the Asian Option

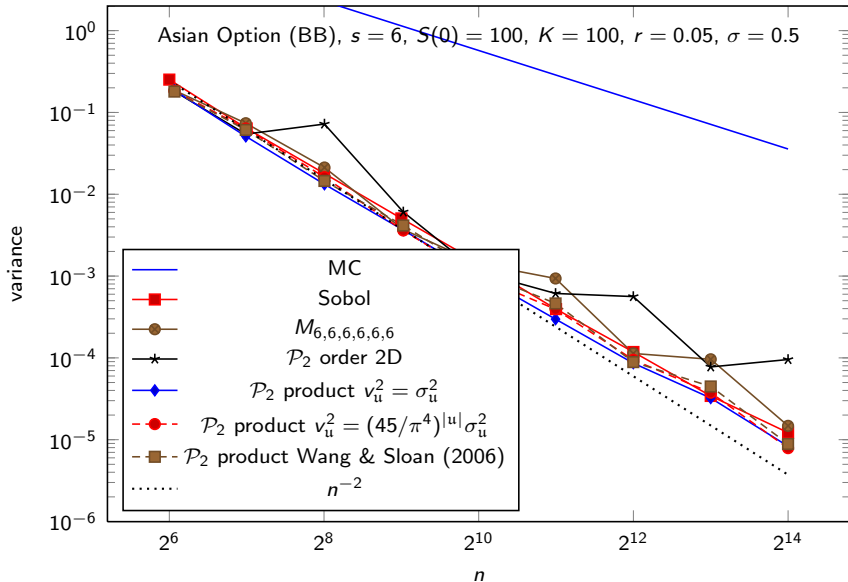
Asian Option ($s = 6$) with $S(0) = 100$, $K = 100$, $r = 0.05$, $\sigma = 0.5$



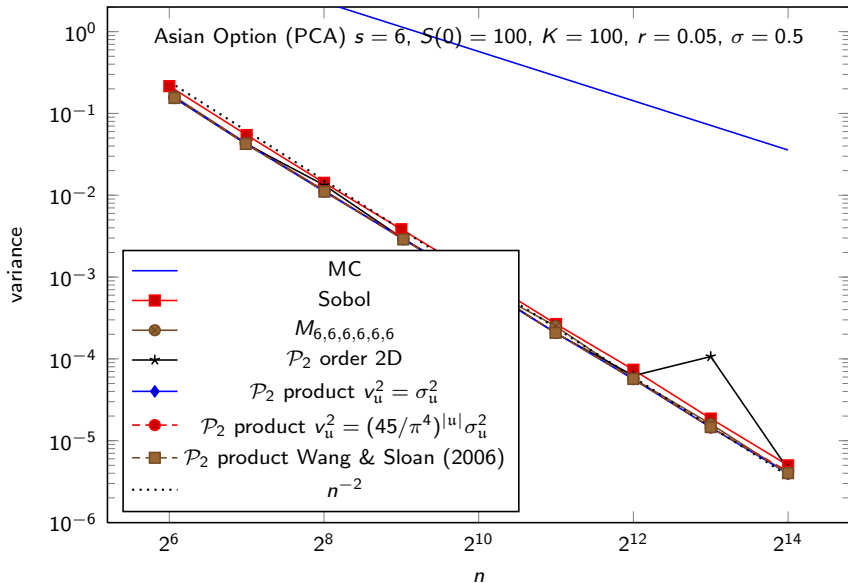
Lattices of Rank 1 with CBC



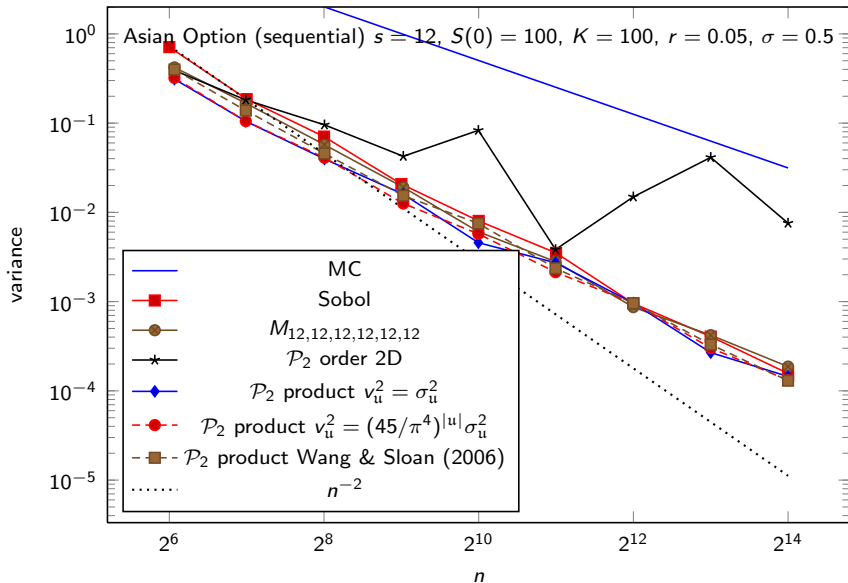
Lattices of Rank 1 with CBC



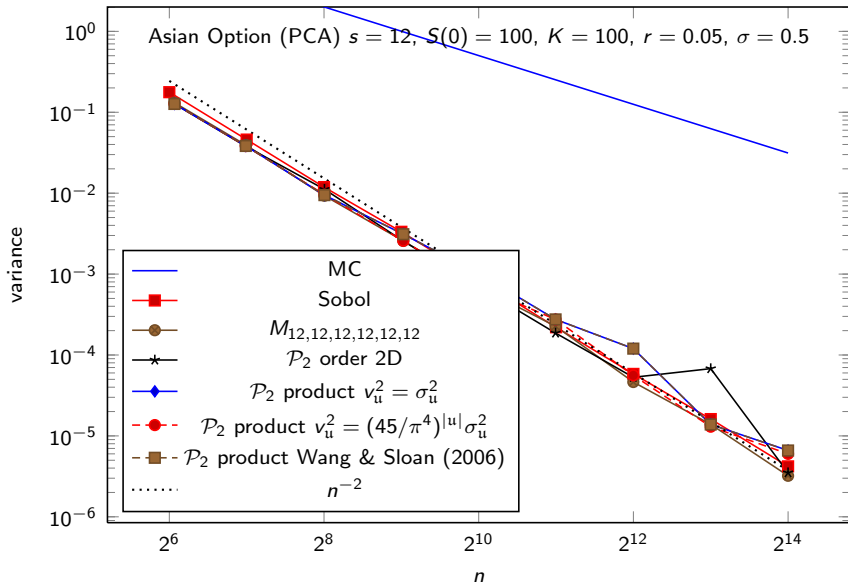
Lattices of Rank 1 with CBC



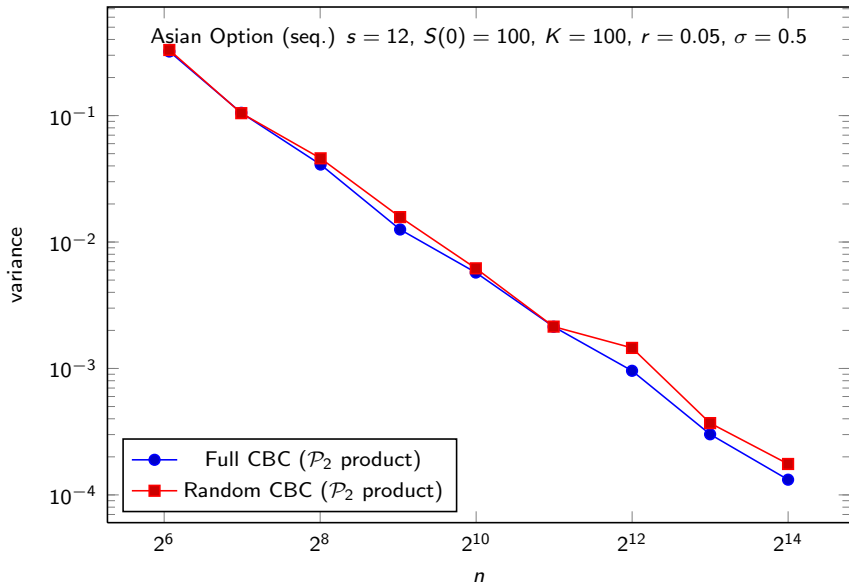
Lattices of Rank 1 with CBC



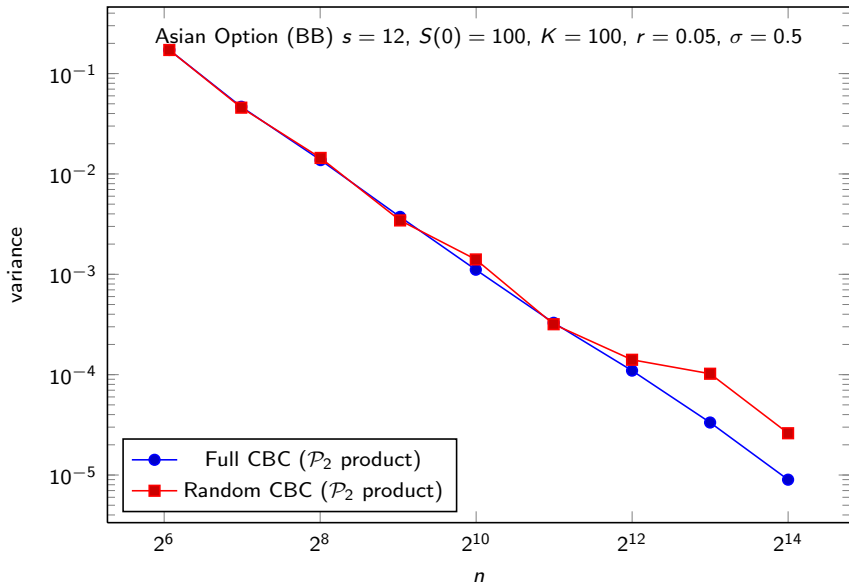
Lattices of Rank 1 with CBC



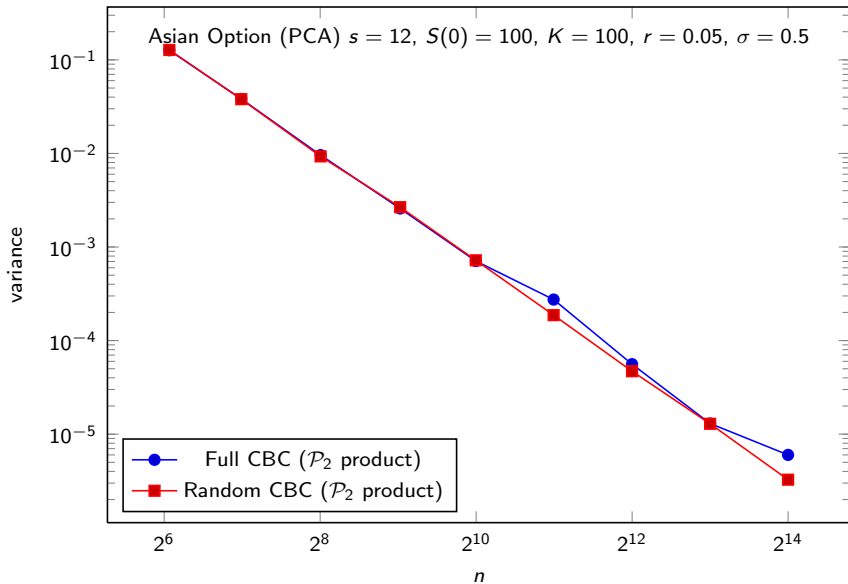
Random vs. Full CBC



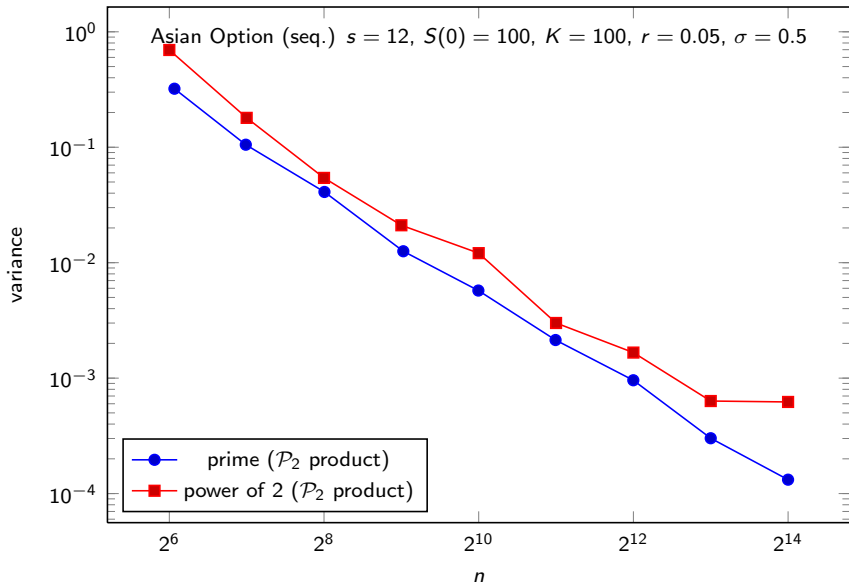
Random vs. Full CBC



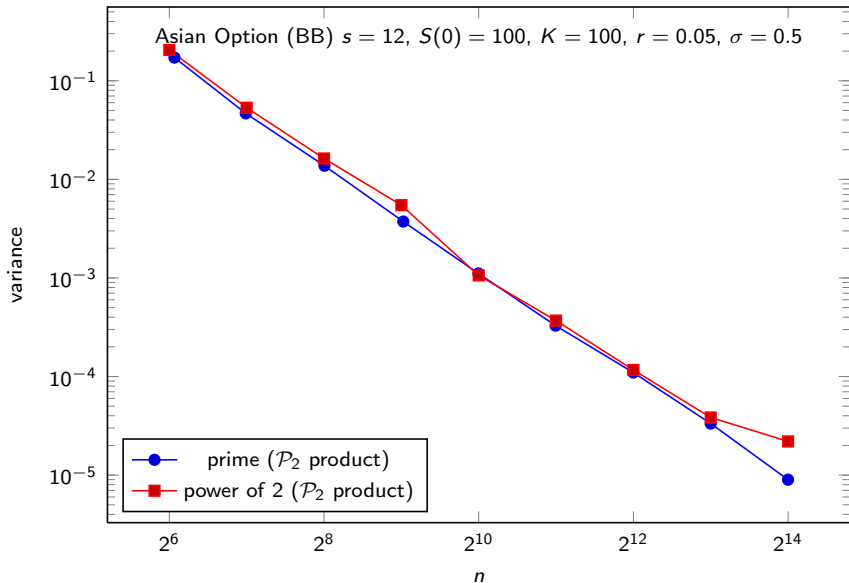
Random vs. Full CBC



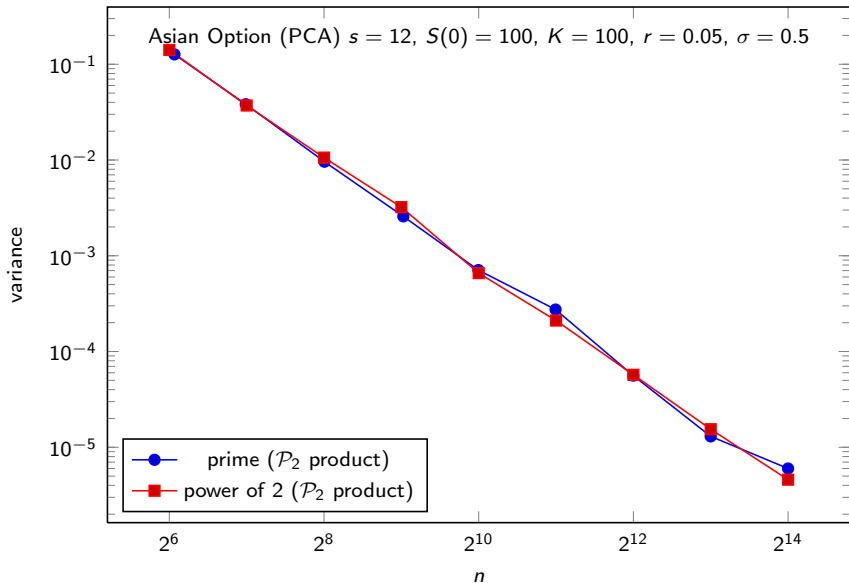
Prime vs. Power-of-2 Number of Points



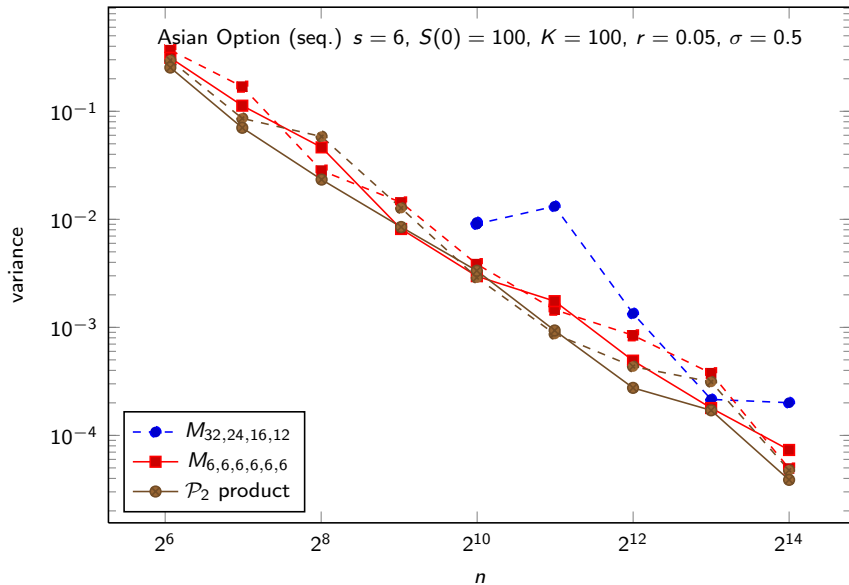
Prime vs. Power-of-2 Number of Points



Prime vs. Power-of-2 Number of Points

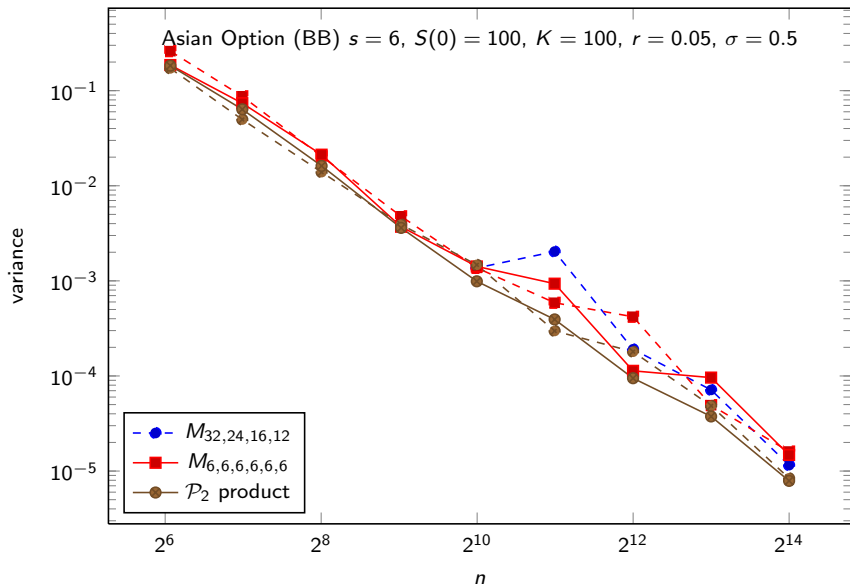


Korobov vs. CBC



Solid: CBC. Dashed: Korobov.

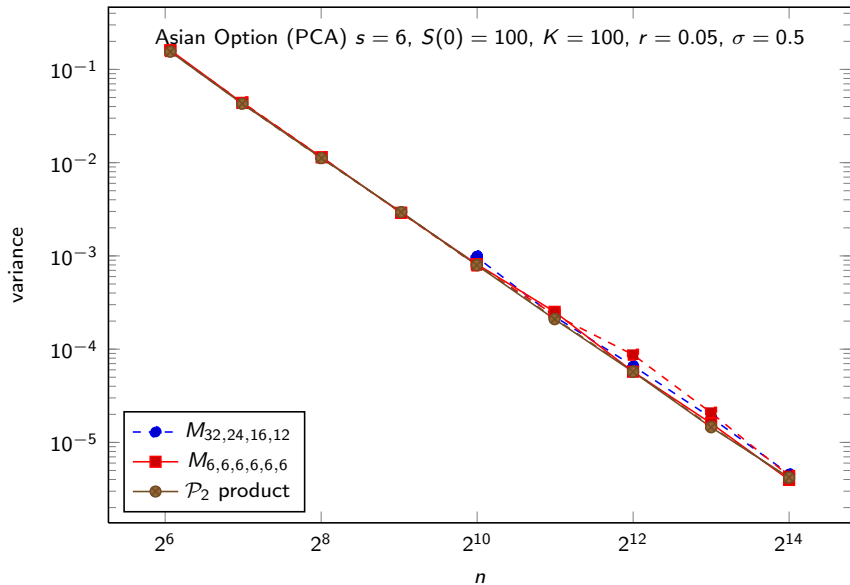
Korobov vs. CBC



Solid: CBC.

Dashed: Korobov.

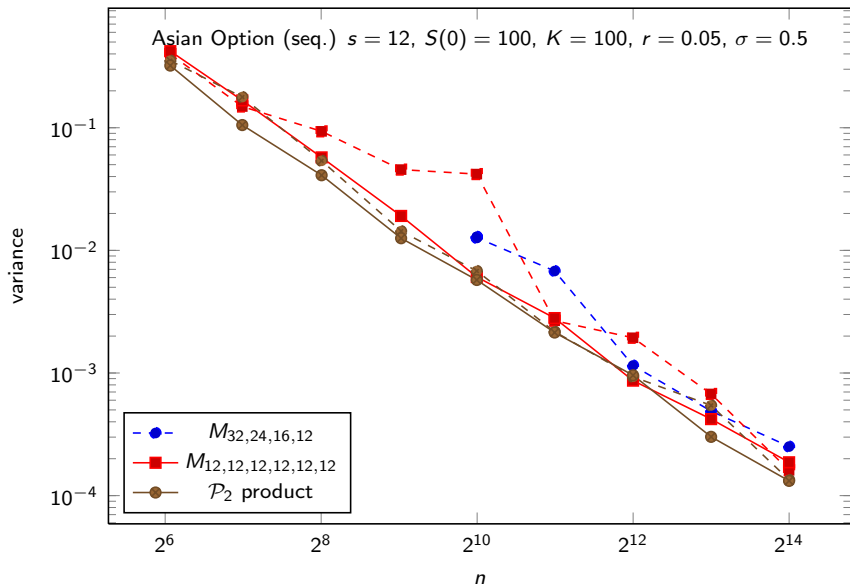
Korobov vs. CBC



Solid: CBC.

Dashed: Korobov.

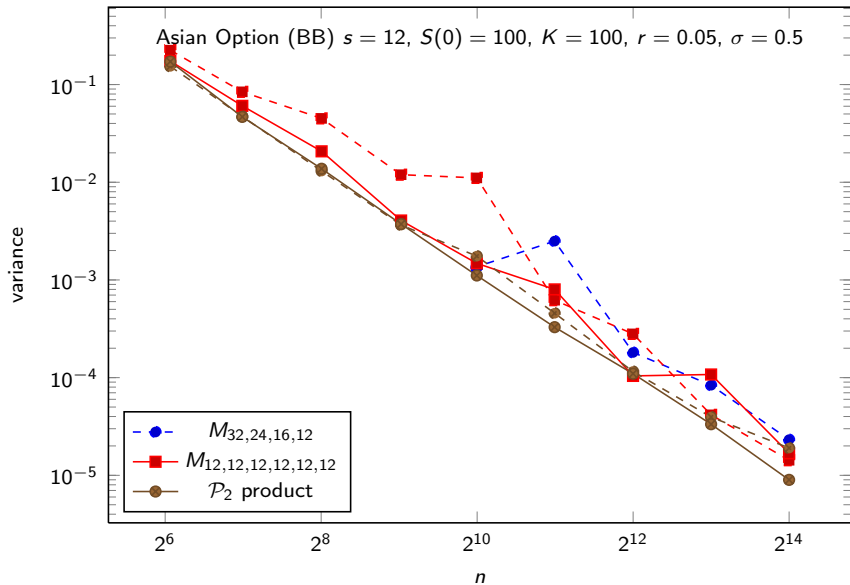
Korobov vs. CBC



Solid: CBC.

Dashed: Korobov.

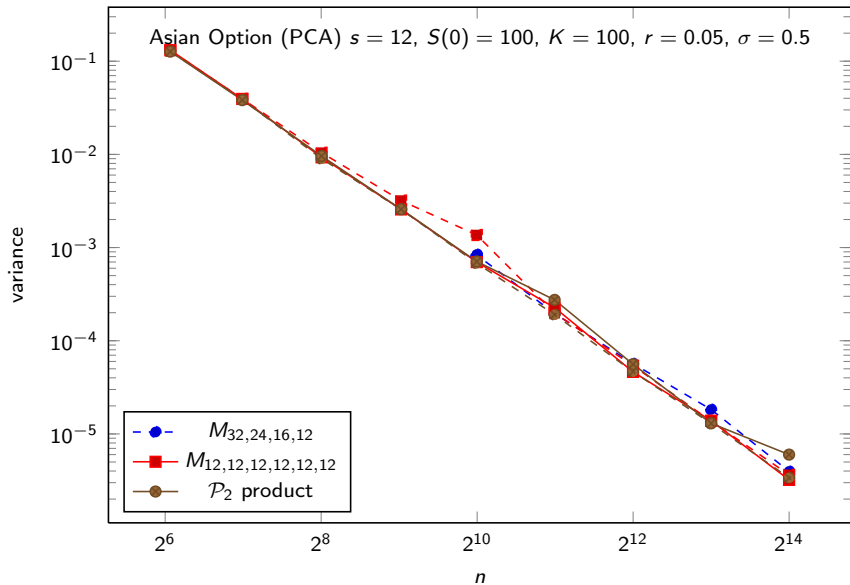
Korobov vs. CBC



Solid: CBC.

Dashed: Korobov.

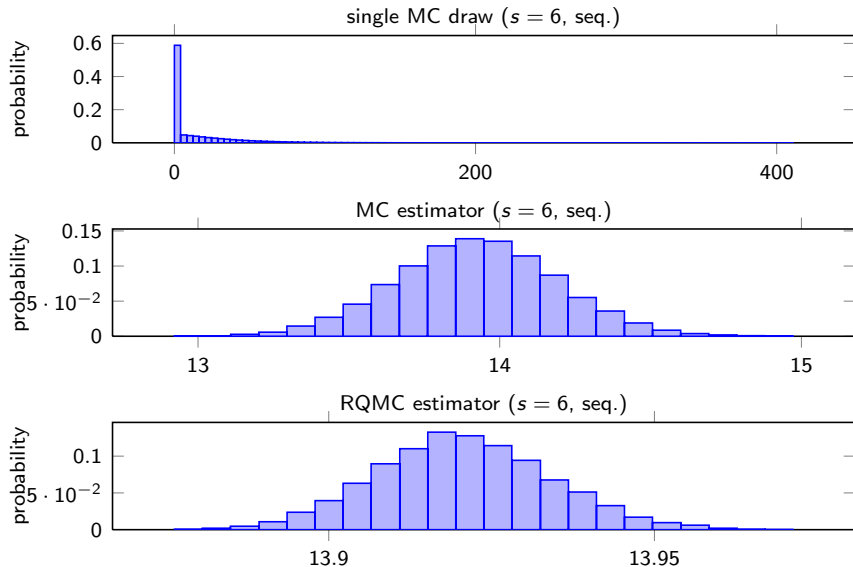
Korobov vs. CBC



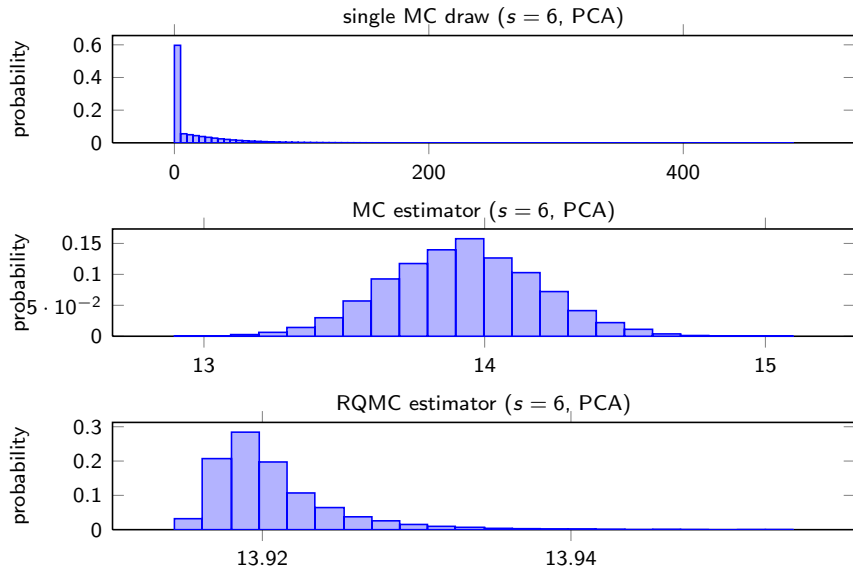
Solid: CBC.

Dashed: Korobov.

Histograms for the Asian Option in $s = 6$



Histograms for the Asian option, $s = 6$



A down-and-in Asian option with barrier B

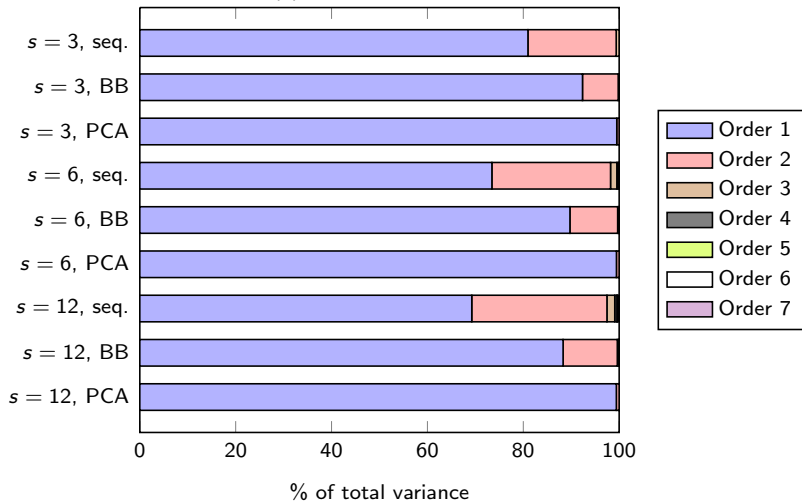
Same as for Asian option, except that payoff is zero unless

$$\min_{1 \leq j \leq s} S(t_j) \leq B$$

for a given constant $B = 80$.

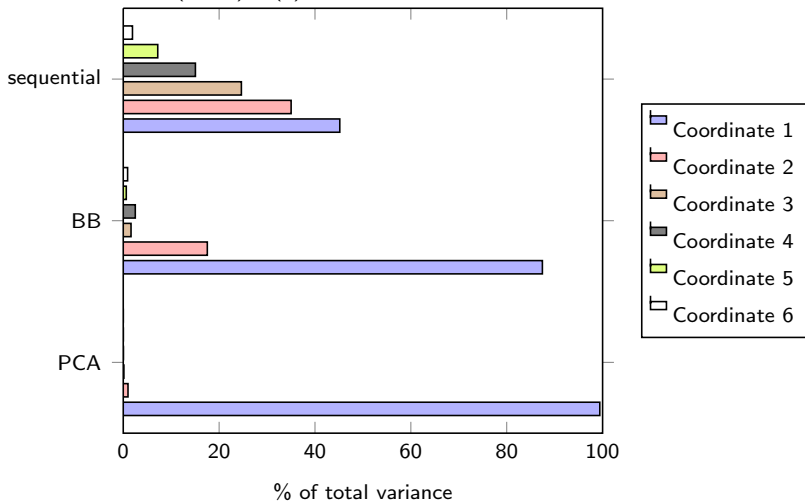
ANOVA Variances for the down-and-in Asian Option

Down-and-in with $S(0) = K = 100$, $r = 0.05$, $\sigma = 0.2$, $B = 80$

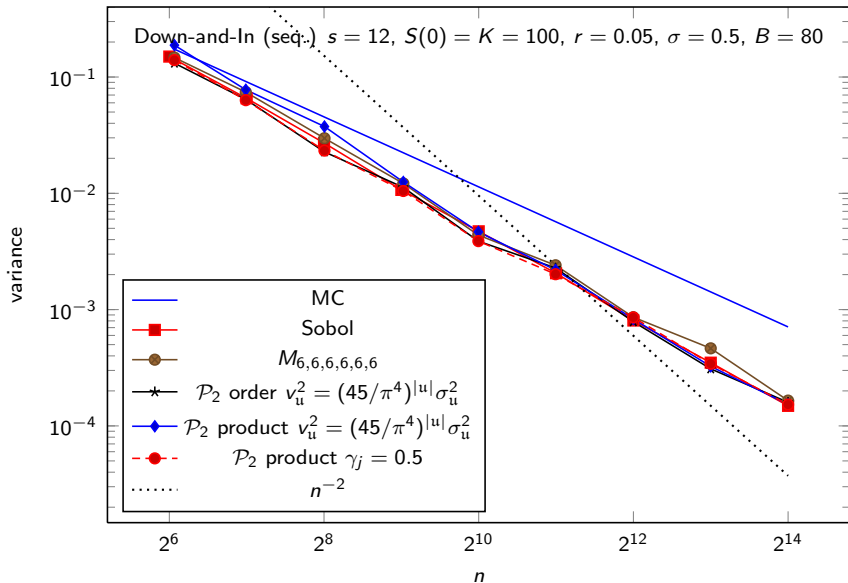


Total Variance per Coordinate for the down-and-in Asian Option

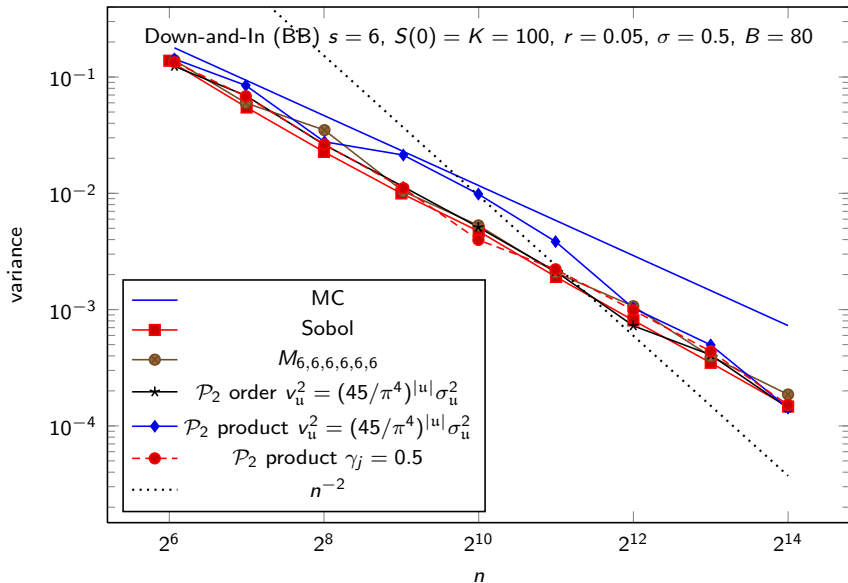
Down-and-In ($s = 6$), $S(0) = K = 100$, $r = 0.05$, $\sigma = 0.2$, $B = 80$



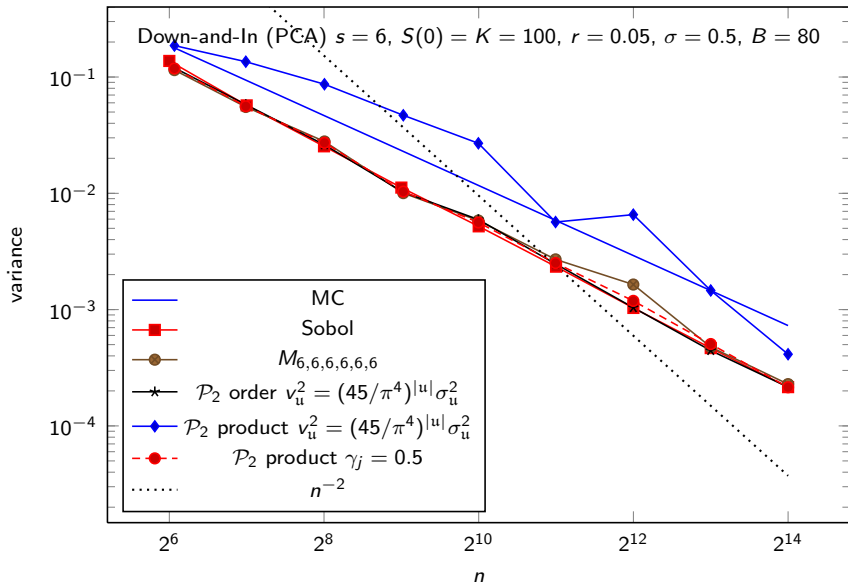
Lattices of Rank 1 with CBC



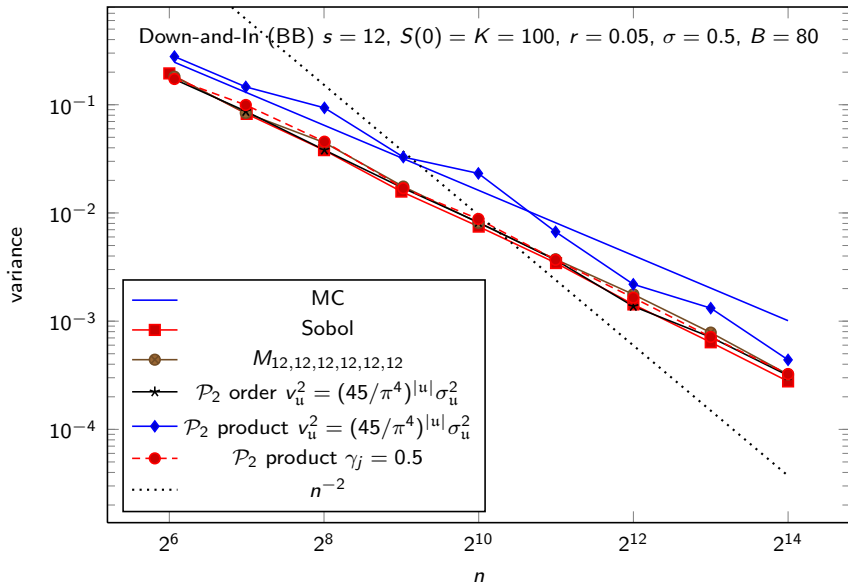
Lattices of Rank 1 with CBC



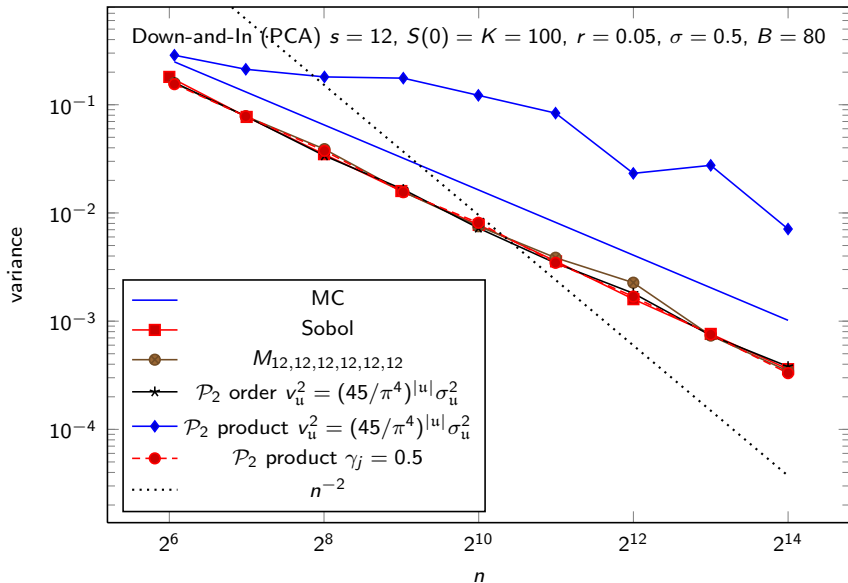
Lattices of Rank 1 with CBC



Lattices of Rank 1 with CBC

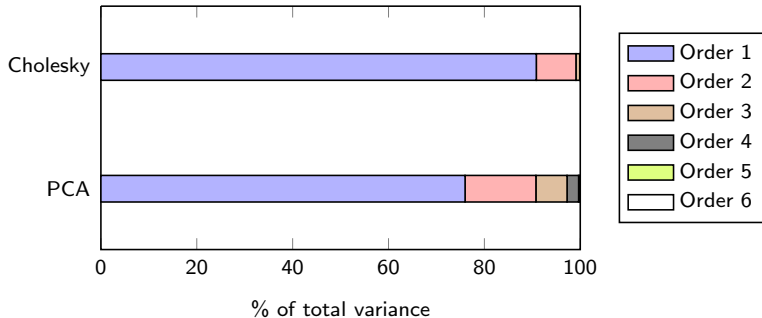


Lattices of Rank 1 with CBC



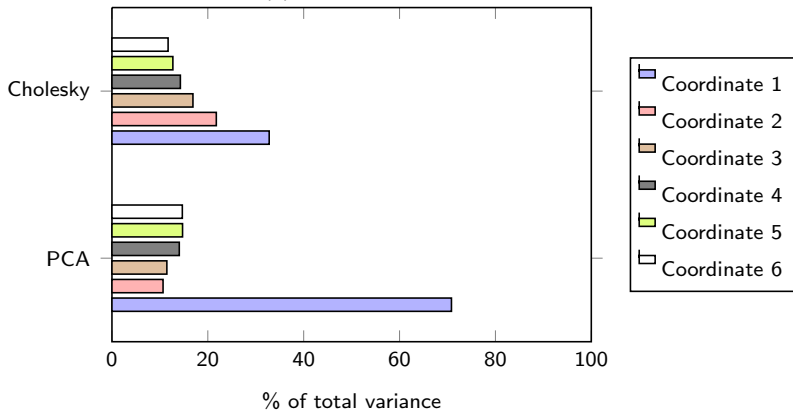
ANOVA variances for the maximum of 6 assets

Maximum of 6 assets, $S(0) = K = 100$, $r = 0.05$, $\sigma = 0.5$, $\rho = 0.3$

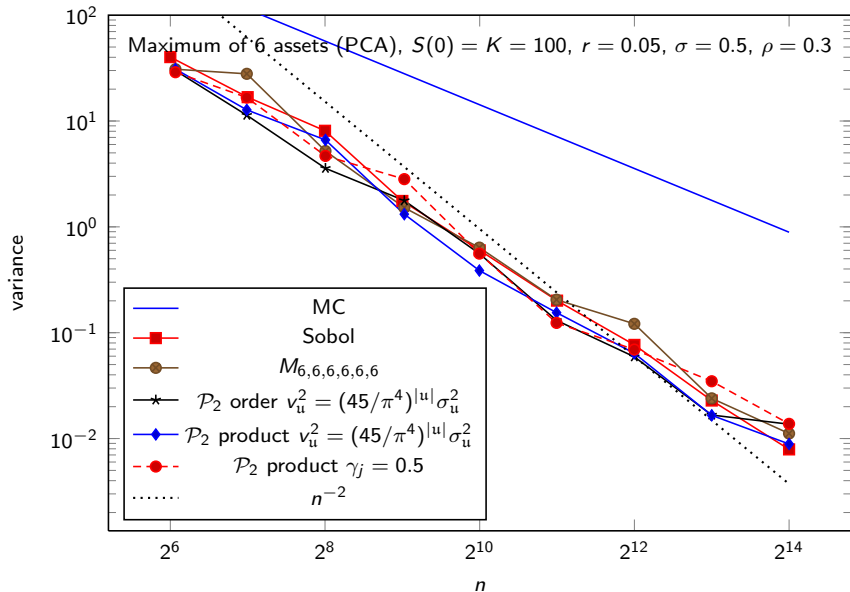


Total Variance per Coordinate for the maximum of 6 assets

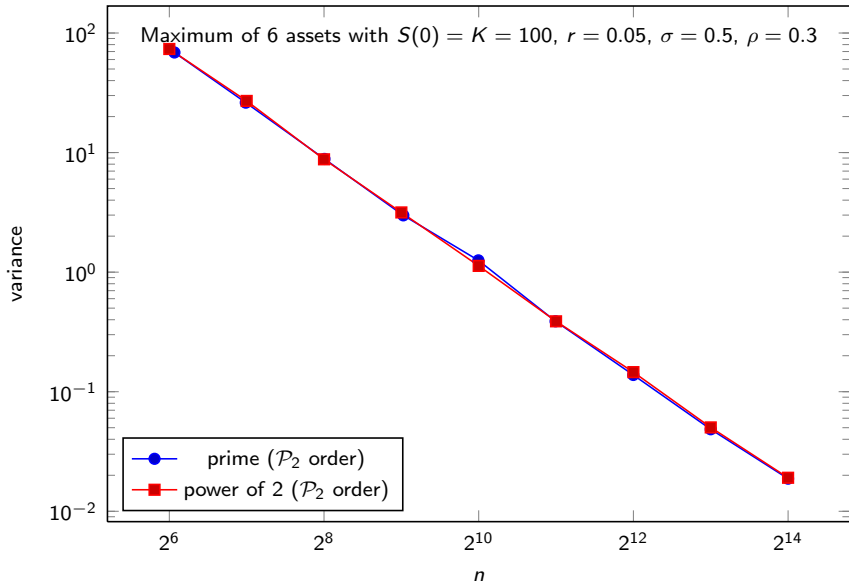
Maximum of 6 assets, $S(0) = K = 100$, $r = 0.05$, $\sigma = 0.5$, $\rho = 0.3$



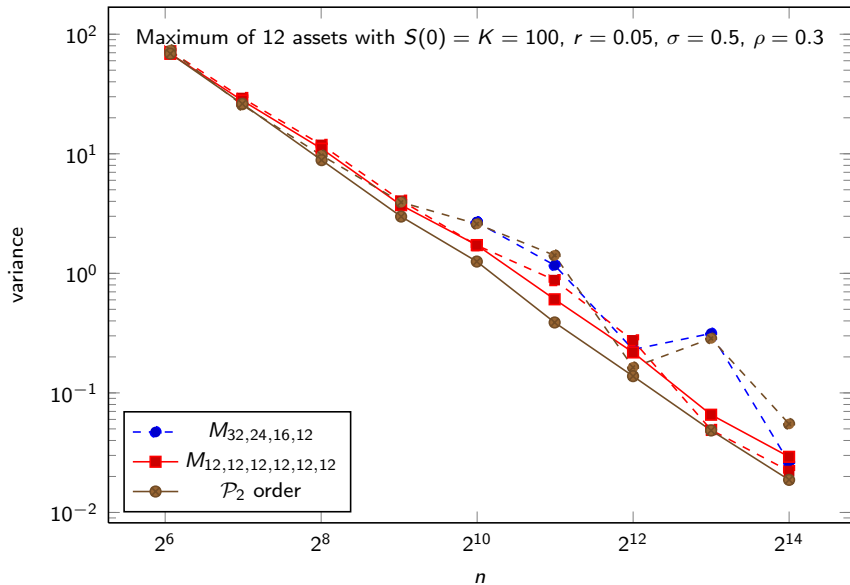
Lattices of Rank 1 with CBC



Prime vs. Power-of-2 Number of Points



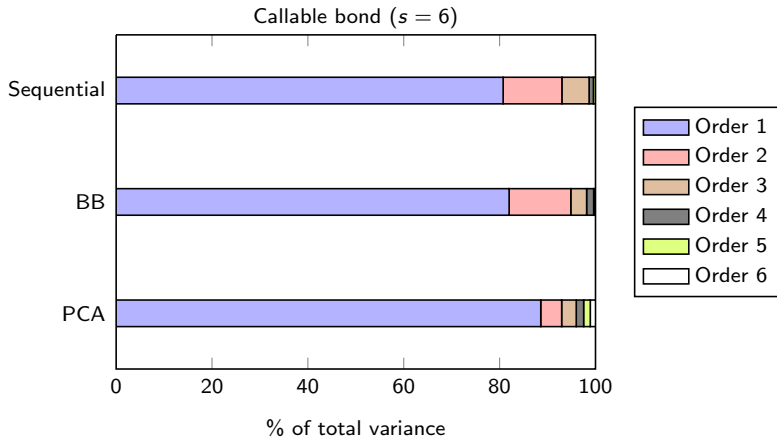
Korobov vs. CBC



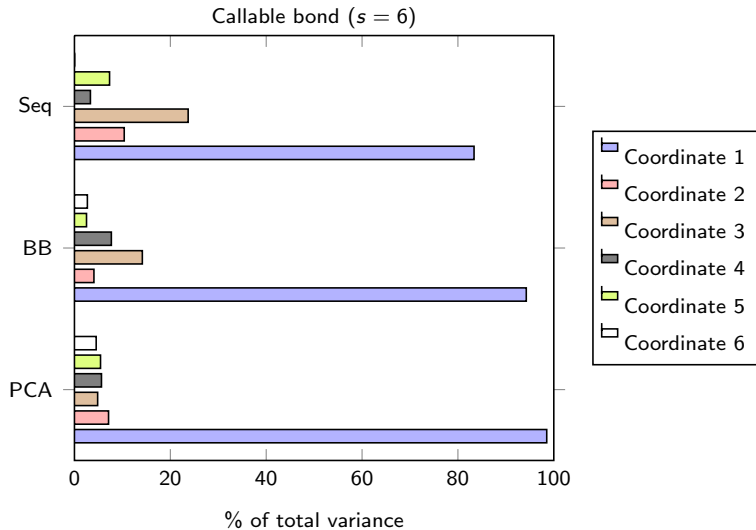
Solid: CBC. Dashed: Korobov.

Callable bond

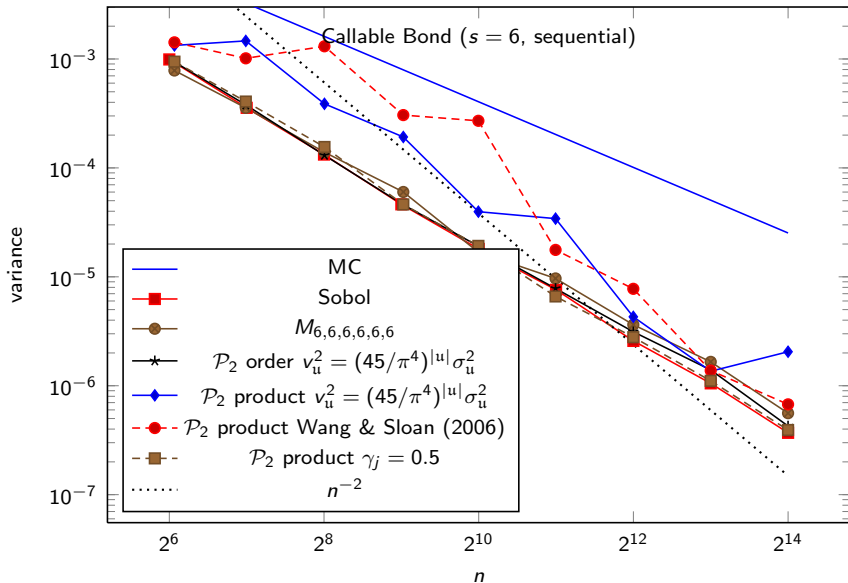
ANOVA Variances for the Callable Bond



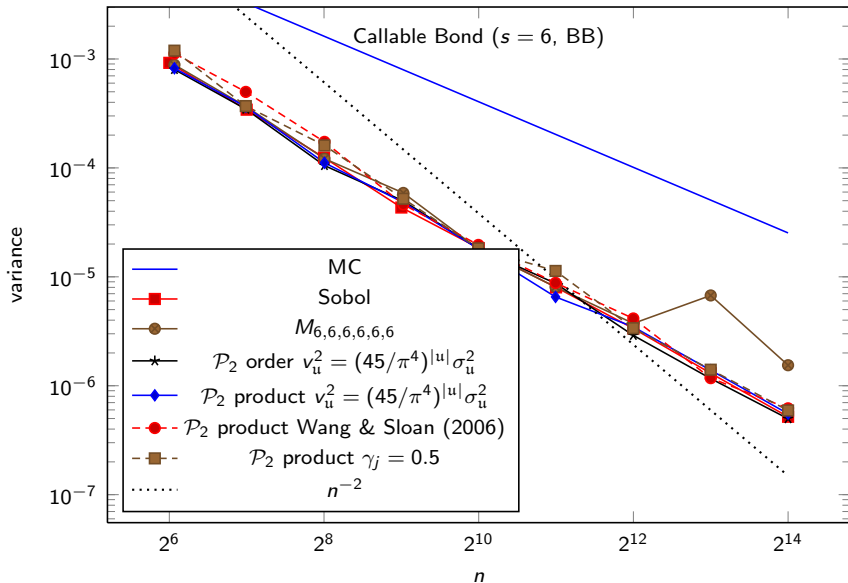
Total Variance per Coordinate for the Callable Bond



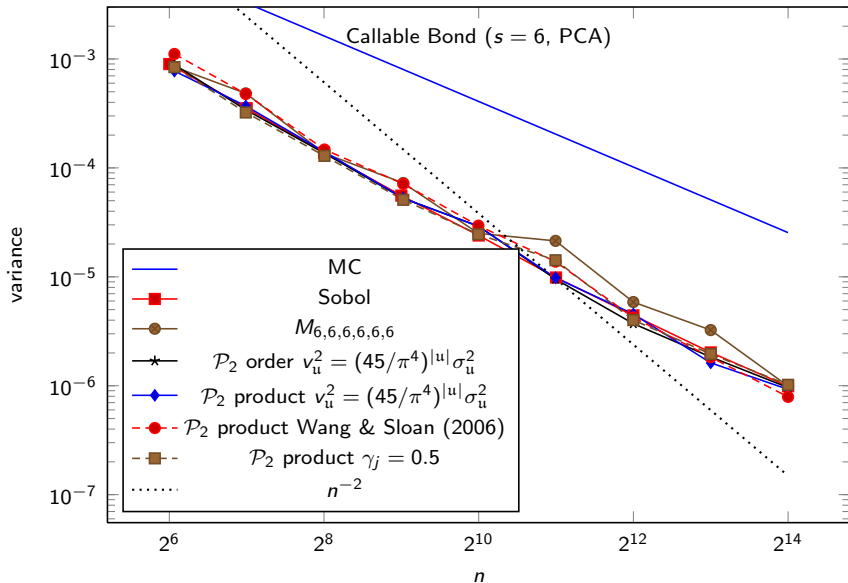
Lattices of Rank 1 with CBC



Lattices of Rank 1 with CBC



Lattices of Rank 1 with CBC



Discrete choice with multinomial mixed logit probability

Utility of alternative j for individual q is

$$U_{q,j} = \beta_q^t \mathbf{x}_{q,j} + \epsilon_{q,j} = \sum_{\ell=1}^s \beta_{q,\ell} x_{q,j,\ell} + \epsilon_{q,j}, \text{ where}$$

$\beta_q^t = (\beta_{q,1}, \dots, \beta_{q,s})$ gives the **tastes** of individual q ,

$\mathbf{x}_{q,j}^t = (x_{q,j,1}, \dots, x_{q,j,s})$ **attributes** of alternative j for individual q ,

$\epsilon_{q,j}$ noise; Gumbel of mean 0 and scale parameter $\lambda = 1$.

Individual q selects alternative with largest utility $U_{q,j}$.

Can observe the $\mathbf{x}_{q,j}$ and choices y_q , but not the rest.

Logit model: for β_q fixed, j is chosen with probability

$$L_q(j | \beta_q) = \frac{\exp[\beta_q^t \mathbf{x}_{q,j}]}{\sum_{a \in \mathcal{A}(q)} \exp[\beta_q^t \mathbf{x}_{q,a}]}$$

where $\mathcal{A}(q)$ are the available alternatives for q .

Logit model: for β_q fixed, j is chosen with probability

$$L_q(j | \beta_q) = \frac{\exp[\beta_q^t \mathbf{x}_{q,j}]}{\sum_{a \in \mathcal{A}(q)} \exp[\beta_q^t \mathbf{x}_{q,a}]}$$

where $\mathcal{A}(q)$ are the available alternatives for q .

For a random individual, suppose β_q is random with density f_{θ} , which depends on (unknown) parameter vector θ . We want to estimate θ from the data (the $\mathbf{x}_{q,j}$ and y_q).

The unconditional probability of choosing j is

$$p_q(j, \theta) = \int L_q(j | \beta) f_{\theta}(\beta) \beta.$$

It depends on $\mathcal{A}(q)$, j , and θ .

Maximum likelihood: Maximize the log of the joint probability of the sample, w.r.t. θ :

$$\ln L(\theta) = \ln \prod_{q=1}^m p_q(y_q, \theta) = \sum_{q=1}^m \ln p_q(y_q, \theta).$$

Maximum likelihood: Maximize the log of the joint probability of the sample, w.r.t. θ :

$$\ln L(\theta) = \ln \prod_{q=1}^m p_q(y_q, \theta) = \sum_{q=1}^m \ln p_q(y_q, \theta).$$

No formula for $p_q(j, \theta)$, but can use MC or RQMC, for each q and fixed θ .

Generate n realizations of β from f_θ , say $\beta_q^{(1)}(\theta), \dots, \beta_q^{(n)}(\theta)$, and estimate $p_q(y_q, \theta)$ by

$$\hat{p}_q(y_q, \theta) = \frac{1}{n} \sum_{i=1}^n L_q(j, \beta_q^{(i)}(\theta)).$$

Then we can find the maximizer $\hat{\theta}$ of $\ln \prod_{q=1}^m \hat{p}_q(y_q, \theta)$ w.r.t. θ .

Maximum likelihood: Maximize the log of the joint probability of the sample, w.r.t. θ :

$$\ln L(\theta) = \ln \prod_{q=1}^m p_q(y_q, \theta) = \sum_{q=1}^m \ln p_q(y_q, \theta).$$

No formula for $p_q(j, \theta)$, but can use MC or RQMC, for each q and fixed θ .

Generate n realizations of β from f_θ , say $\beta_q^{(1)}(\theta), \dots, \beta_q^{(n)}(\theta)$, and estimate $p_q(y_q, \theta)$ by

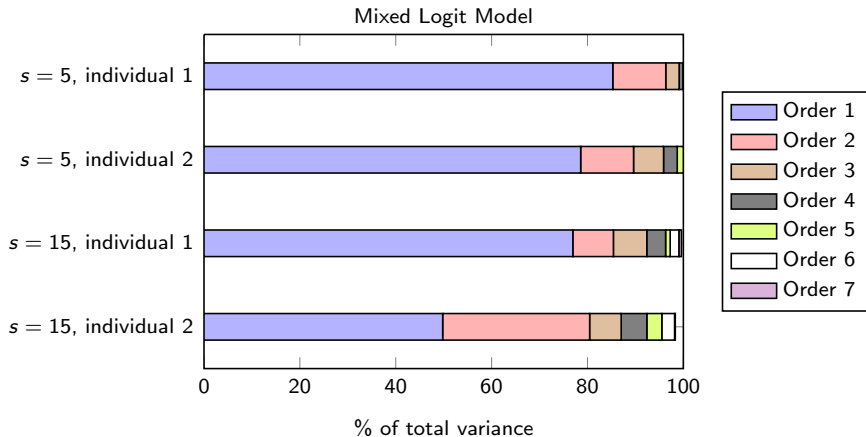
$$\hat{p}_q(y_q, \theta) = \frac{1}{n} \sum_{i=1}^n L_q(j, \beta_q^{(i)}(\theta)).$$

Then we can find the maximizer $\hat{\theta}$ of $\ln \prod_{q=1}^m \hat{p}_q(y_q, \theta)$ w.r.t. θ .

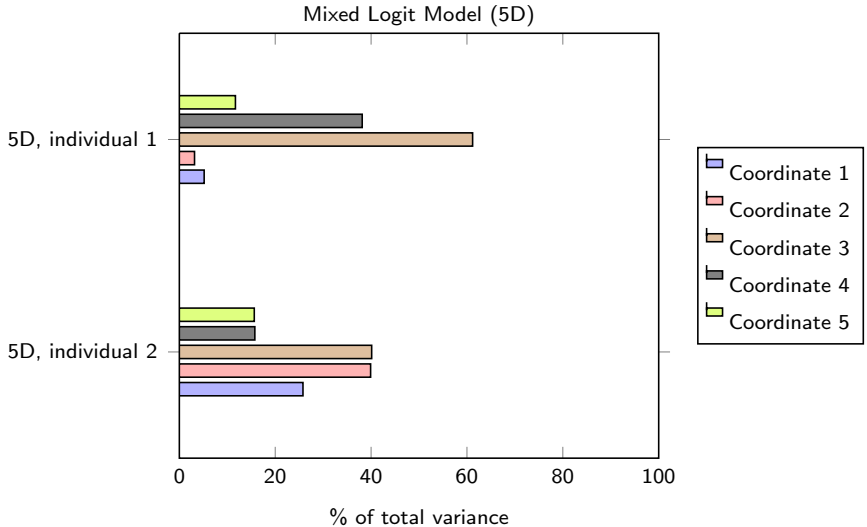
– How many alternatives? – 4 alternatives, the first two are generated from independent $N(1, 1)$ and the last two from $N(0.5, 1)$.

Suppose β_q is a vector of s independent $N(1, 1)$ random variables. We try $s = 5, 10, 15$.

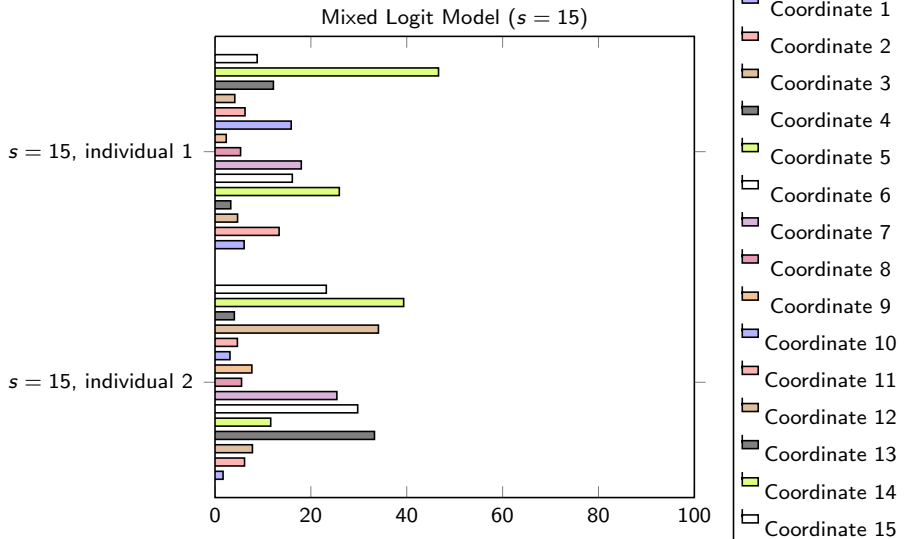
ANOVA Variances for the Mixed Logit Model



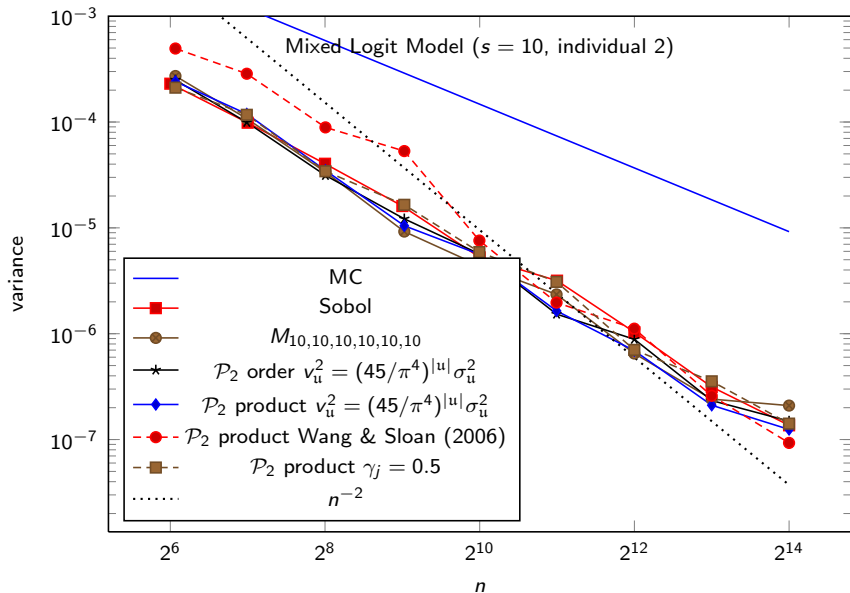
Total Variance per Coordinate for the Mixed Logit Model



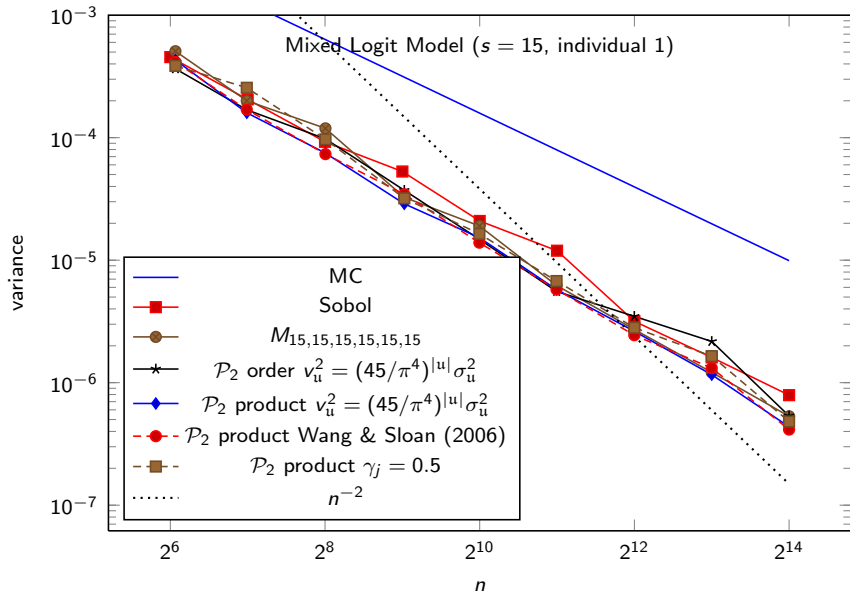
Total Variance per Coordinate for the Mixed Logit Model



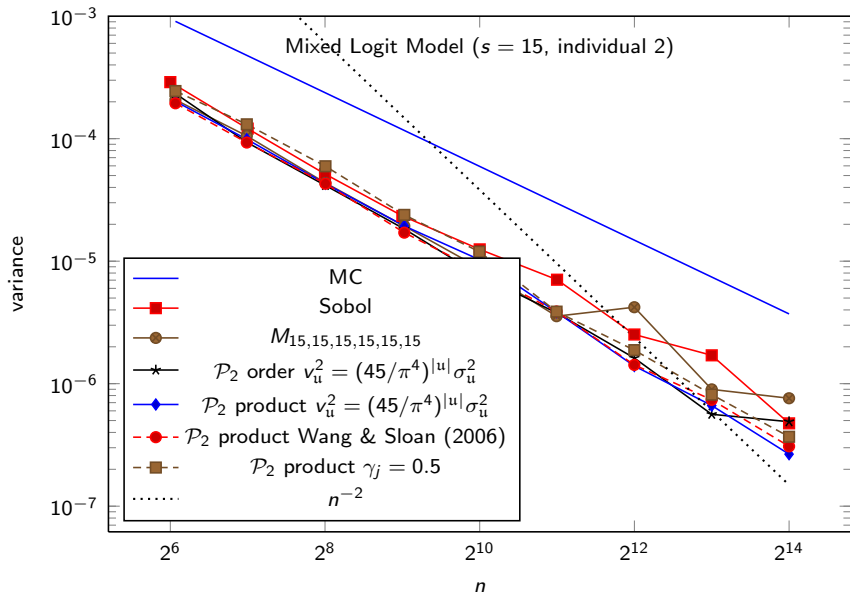
Lattices of Rank 1 with CBC



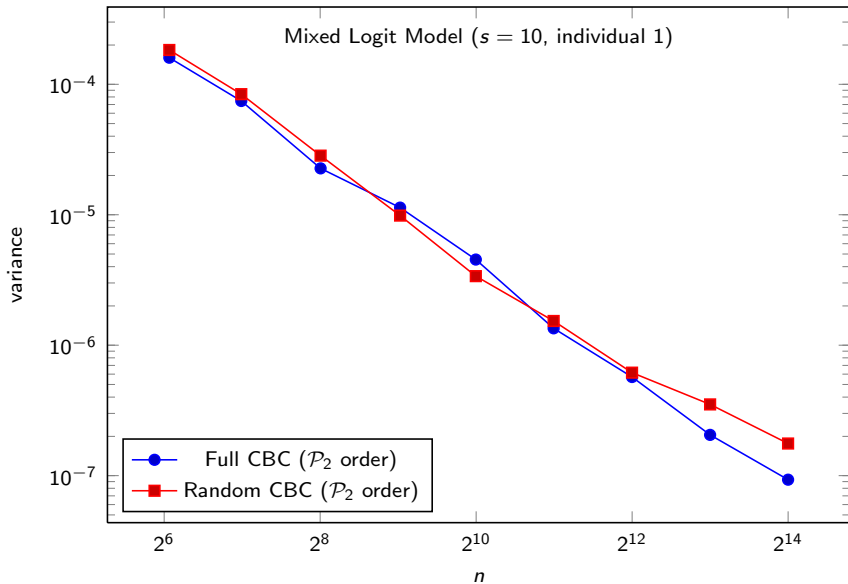
Lattices of Rank 1 with CBC



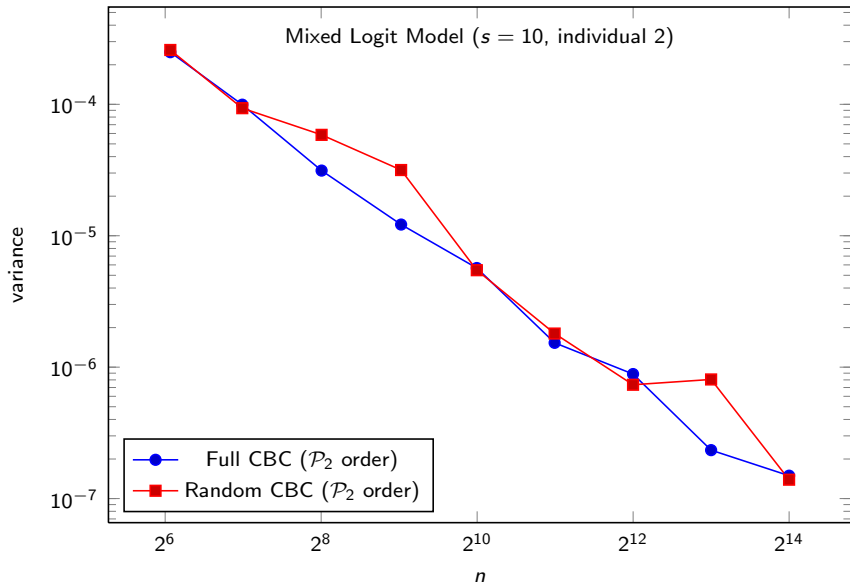
Lattices of Rank 1 with CBC



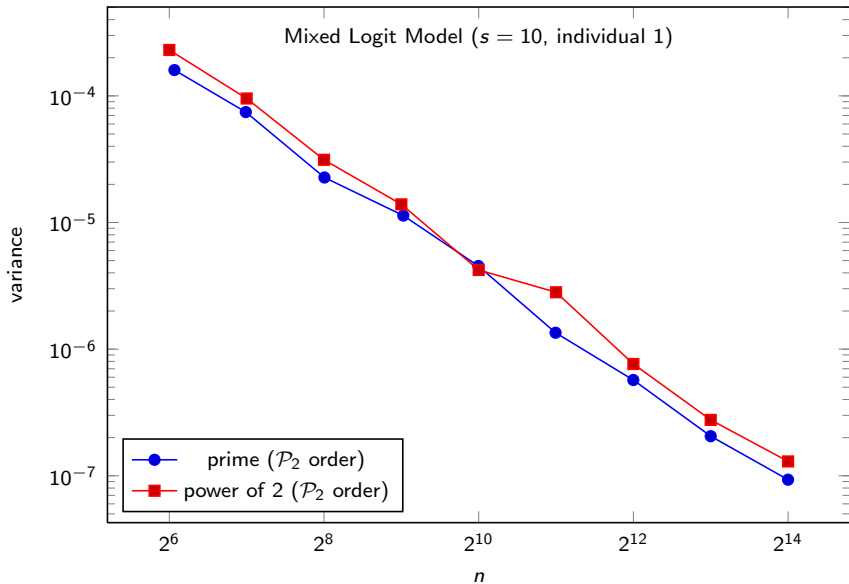
Random vs. Full CBC



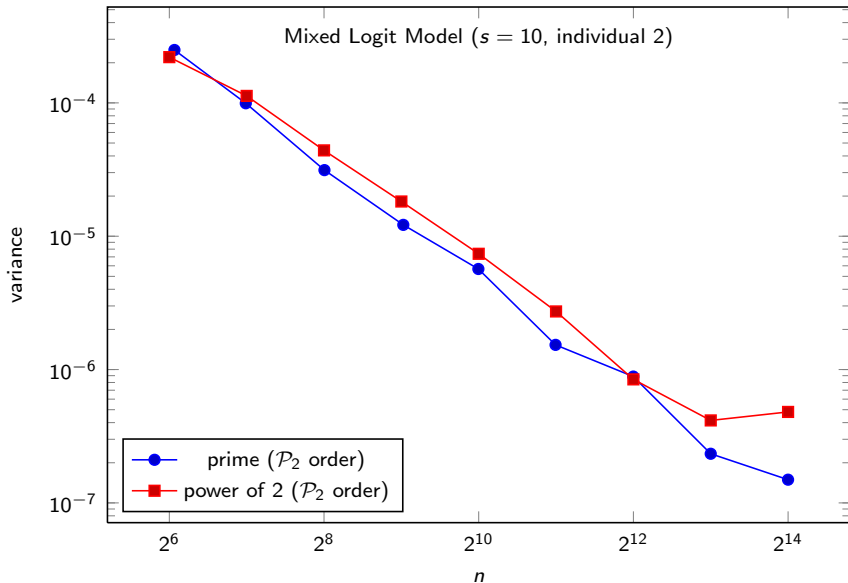
Random vs. Full CBC



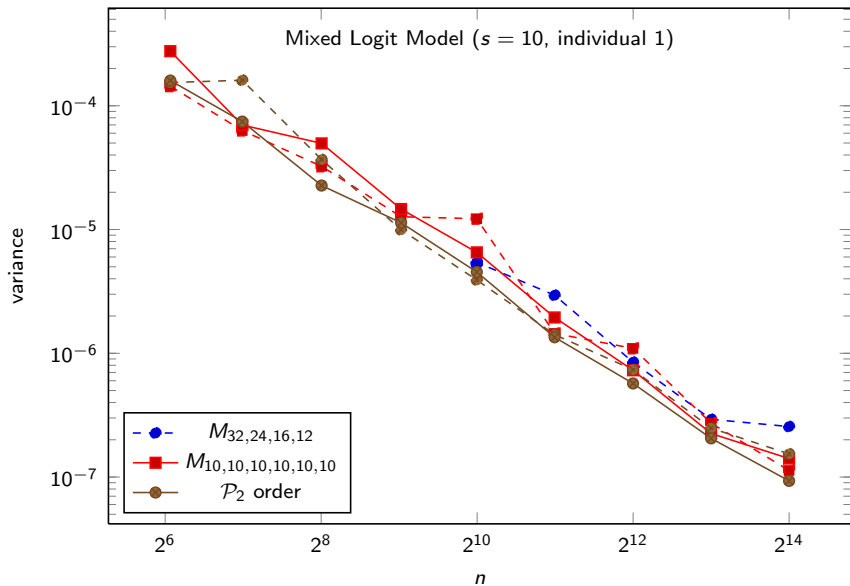
Prime vs. Power-of-2 Number of Points



Prime vs. Power-of-2 Number of Points

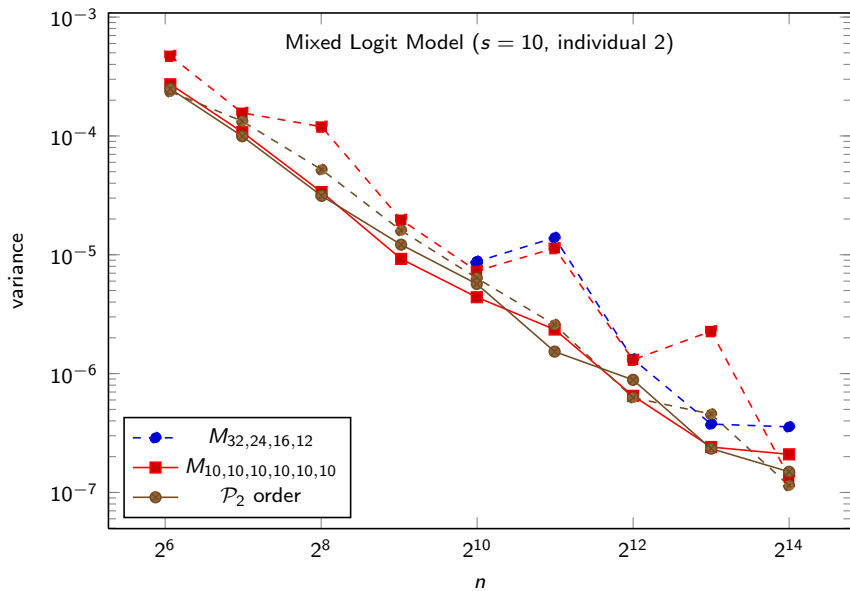


Korobov vs. CBC



Solid: CBC. Dashed: Korobov.

Korobov vs. CBC



Solid: CBC. Dashed: Korobov.

Still missing

- Compare criteria with and without baker.
- Il faudrait essayer un critere qui donne un poids different a chaque u , au lieu de supposer la forme produit ou order-dependent.