

Randomized Quasi-Monte Carlo Simulation of Multidimensional Markov Chains

Pierre L'Ecuyer, Université de Montréal

Christian Lécot, Université de Savoie, Chambéry

Adam L'Archevêque-Gaudet, Université de Montréal

Monte Carlo for Markov Chains

Setting: A Markov chain with state space $\mathcal{X} \subseteq \mathbb{R}^\ell$, evolves as

$$X_0 = x_0, \quad X_j = \varphi_j(X_{j-1}, \mathbf{U}_j), \quad j \geq 1,$$

where the \mathbf{U}_j are i.i.d. uniform r.v.'s over $(0, 1)^d$.

We want to estimate

$$\mu = \mathbb{E}[Y] \quad \text{where} \quad Y = \sum_{j=1}^{\tau} c_j(X_j)$$

and τ is a stopping time w.r.t. $\mathcal{F}\{(j, X_j), j \geq 0\}$.

Monte Carlo for Markov Chains

Setting: A Markov chain with state space $\mathcal{X} \subseteq \mathbb{R}^\ell$, evolves as

$$X_0 = x_0, \quad X_j = \varphi_j(X_{j-1}, \mathbf{U}_j), \quad j \geq 1,$$

where the \mathbf{U}_j are i.i.d. uniform r.v.'s over $(0, 1)^d$.

We want to estimate

$$\mu = \mathbb{E}[Y] \quad \text{where} \quad Y = \sum_{j=1}^{\tau} c_j(X_j)$$

and τ is a stopping time w.r.t. $\mathcal{F}\{(j, X_j), j \geq 0\}$.

Ordinary MC: For $i = 0, \dots, n-1$, generate $X_{i,j} = \varphi_j(X_{i,j-1}, \mathbf{U}_{i,j})$, $j = 1, \dots, \tau_i$, where the $\mathbf{U}_{i,j}$'s are i.i.d. $U(0, 1)^d$. Estimate μ by

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^{\tau_i} c(X_{i,j}) = \frac{1}{n} \sum_{i=1}^n Y_i.$$

Classical RQMC for Markov Chains

Put $\mathbf{V}_i = (\mathbf{U}_{i,1}, \mathbf{U}_{i,2}, \dots)$.

Estimate μ by

$$\hat{\mu}_{\text{rqmc},n} = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^{\tau_i} c(X_{i,j})$$

where $P_n = \{\mathbf{V}_0, \dots, \mathbf{V}_{n-1}\} \subset (0, 1)^s$ has the following properties:

- (a) each point \mathbf{V}_i has the **uniform distribution** over $(0, 1)^s$;
- (b) P_n has low discrepancy.

Dimension is $s = \inf\{s' : \mathbb{P}[d\tau \leq s'] = 1\}$.

For a Markov chain, the effective dimension often remains very large!

Array-RQMC for Markov Chains

[Lécot, Tuffin, L'Ecuyer 2004]

Simulate n chains in parallel. At each step, use an RQMC point set P_n to advance all the chains by one step, while inducing global negative dependence across the chains.

Intuition: The empirical distribution of $X_{0,j}, \dots, X_{n-1,j}$, should be a very accurate approximation of the theoretical distribution of X_j , for each j . The **discrepancy** between these two distributions should be as small as possible.

Array-RQMC for Markov Chains

[Lécot, Tuffin, L'Ecuyer 2004]

Simulate n chains in parallel. At each step, use an RQMC point set P_n to advance all the chains by one step, while inducing global negative dependence across the chains.

Intuition: The empirical distribution of $X_{0,j}, \dots, X_{n-1,j}$, should be a very accurate approximation of the theoretical distribution of X_j , for each j . The **discrepancy** between these two distributions should be as small as possible.

Then, we will have small error in the estimates:

$$\mathbb{E}[c_j(X_j)] \approx \frac{1}{n} \sum_{i=0}^{n-1} c_j(X_{i,j}) \quad \text{and} \quad \mu \approx \frac{1}{n} \sum_{i=0}^{n-1} Y_i.$$

Array-RQMC for Markov Chains

[Lécot, Tuffin, L'Ecuyer 2004]

Simulate n chains in parallel. At each step, use an RQMC point set P_n to advance all the chains by one step, while inducing global negative dependence across the chains.

Intuition: The empirical distribution of $X_{0,j}, \dots, X_{n-1,j}$, should be a very accurate approximation of the theoretical distribution of X_j , for each j . The **discrepancy** between these two distributions should be as small as possible.

Then, we will have small error in the estimates:

$$\mathbb{E}[c_j(X_j)] \approx \frac{1}{n} \sum_{i=0}^{n-1} c_j(X_{i,j}) \quad \text{and} \quad \mu \approx \frac{1}{n} \sum_{i=0}^{n-1} Y_i.$$

How can we preserve low-discrepancy of $X_{0,j}, \dots, X_{n-1,j}$ when j increases? Can we quantify the variance improvement?

Select some measure of **discrepancy** D for a point set in the unit cube, together with RKHS of functions, and corresponding **variation** V .

Let $D_j = D(X_{0,j}, \dots, X_{n-1,j})$, the discrepancy of the states at step j and

$$\hat{\mu}_{\text{rqmc},j,n} = \frac{1}{n} \sum_{i=1}^n c(X_{i,j})$$

Select some measure of **discrepancy** D for a point set in the unit cube, together with RKHS of functions, and corresponding **variation** V .

Let $D_j = D(X_{0,j}, \dots, X_{n-1,j})$, the discrepancy of the states at step j and

$$\hat{\mu}_{\text{rqmc},j,n} = \frac{1}{n} \sum_{i=1}^n c(X_{i,j})$$

We have

$$\text{Var}[\hat{\mu}_{\text{rqmc},j,n}] \leq \mathbb{E}[D_j^2] V^2(c_j).$$

We would like to show, by induction on j , that

$$\mathbb{E}[D_j^2] \leq \kappa_j n^{-\alpha+\epsilon},$$

for some $\alpha \geq 1$, where κ_j does not depend on n and grows only very slowly (or not at all) with j .

Array-RQMC algorithm

For simplicity, assume $X_j \sim U[0, 1]^\ell$;

Roughly speaking, we will view $\mathbb{E}[D_j^2]$ as a $(\ell + d)$ -dimensional integral and estimate it by randomized quasi-Monte Carlo.

1. Select an $(\ell + d)$ -dimensional low-discrepancy point set

$$\tilde{Q}_n = \{(\mathbf{w}_0, \tilde{\mathbf{u}}_0), \dots, (\mathbf{w}_{n-1}, \tilde{\mathbf{u}}_{n-1})\},$$

where $\mathbf{w}_i \in [0, 1]^\ell$ and $\tilde{\mathbf{u}}_i \in [0, 1]^d$.

Define a randomization of $\tilde{P}_n = \{\tilde{\mathbf{u}}_0, \dots, \tilde{\mathbf{u}}_{n-1}\}$.

Let $P_n = (\mathbf{u}_0, \dots, \mathbf{u}_{n-1})$ denote its realization and Q_n the randomized \tilde{Q}_n . This randomization must satisfy:

- (a) each \mathbf{u}_i is $U[0, 1]^d$ and
- (b) Q_n has low discrepancy (in some sense).

2. Simulate (in parallel) n copies of the chain, as follows:

For $j = 1, 2, \dots$ until all stopping times are reached

Randomize \tilde{P}_n afresh into $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$;

For $i = 0, \dots, n - 1$, do $X_{i,j} = \varphi_j(X_{i,j-1}, \mathbf{u}_i)$;

2. Simulate (in parallel) n copies of the chain, as follows:

For $j = 1, 2, \dots$ until all stopping times are reached

Randomize \tilde{P}_n afresh into $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$;

For $i = 0, \dots, n - 1$, do $X_{i,j} = \varphi_j(X_{i,j-1}, \mathbf{u}_i)$;

Map the n chains to the n points, and renumber the chains accordingly, so that $X_{i,j}$ is “close” to \mathbf{w}_i for each i
(more on this);

2. Simulate (in parallel) n copies of the chain, as follows:

For $j = 1, 2, \dots$ until all stopping times are reached

Randomize \tilde{P}_n afresh into $P_n = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$;

For $i = 0, \dots, n - 1$, do $X_{i,j} = \varphi_j(X_{i,j-1}, \mathbf{u}_i)$;

Map the n chains to the n points, and renumber the chains accordingly, so that $X_{i,j}$ is “close” to \mathbf{w}_i for each i (more on this);

3. Estimate μ by the average \bar{Y}_n of the n values of Y .

Replicate m times to estimate variance and compute confidence interval on μ .

Idea: Would like to estimate $\mathbb{E}[c_j(X_j)] = \mathbb{E}[c_j(\varphi_j(X_{j-1}, \mathbf{U}))]$ by RQMC using Q_n . But we replace \mathbf{w}_i by $X_{i,j-1}$.

Proposition (unbiasedness):

(a) The average \bar{Y}_n is an **unbiased** estimator of μ and

(b) the **empirical variance** of its m copies is an unbiased estimator of $\text{Var}[\bar{Y}_n]$.

Mapping chains to points

Multivariate sort:

Sort the states (the chains) by their first coordinate, in n_1 packets of size n/n_1 .

Then sort each packet by the second coordinate, in n_2 packets of size $n/n_1 n_2$.

...

At the last level, sort each packet of size n_ℓ by the last coordinate.

Choice of n_1, n_2, \dots

Mapping chains to points

Multivariate sort:

Sort the states (the chains) by their first coordinate, in n_1 packets of size n/n_1 .

Then sort each packet by the second coordinate, in n_2 packets of size $n/n_1 n_2$.

...

At the last level, sort each packet of size n_ℓ by the last coordinate.

Choice of n_1, n_2, \dots

Generalization:

Define a sorting function $v : \mathcal{X} \rightarrow [0, 1]^c$ and apply the multivariate sort (in c dimensions) to the transformed points $v(X_{i,j})$.

Choice of v : Two states mapped to nearby values of v should be approximately equivalent.

Mapping chains to points

Multivariate sort:

Sort the states (the chains) by their first coordinate, in n_1 packets of size n/n_1 .

Then sort each packet by the second coordinate, in n_2 packets of size $n/n_1 n_2$.

...

At the last level, sort each packet of size n_ℓ by the last coordinate.

Choice of n_1, n_2, \dots

Generalization:

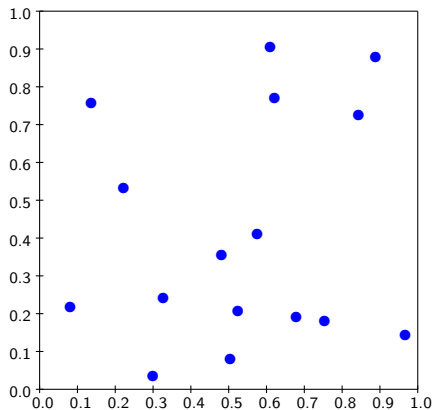
Define a sorting function $v : \mathcal{X} \rightarrow [0, 1]^c$ and apply the multivariate sort (in c dimensions) to the transformed points $v(X_{i,j})$.

Choice of v : Two states mapped to nearby values of v should be approximately equivalent.

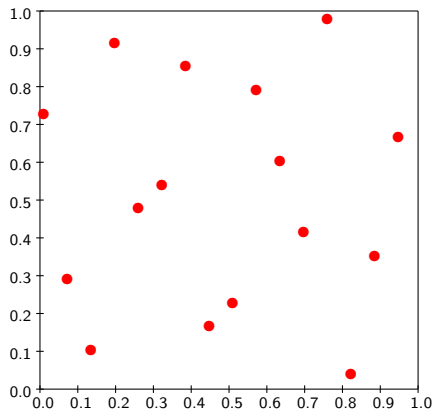
In L. Lécot, Tuffin (2004, 2008), we had $v : \mathcal{X} \rightarrow \mathbb{R}$.

A (4,4) mapping

States of the chains

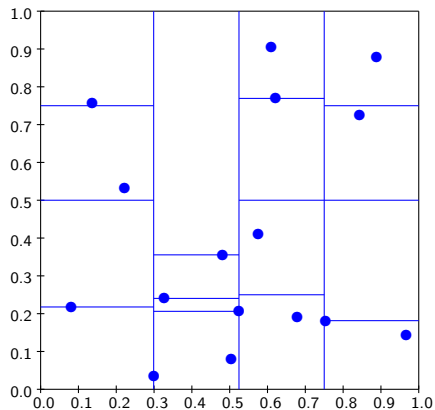


Sobol' Sequence in 2 dimensions with digital shift

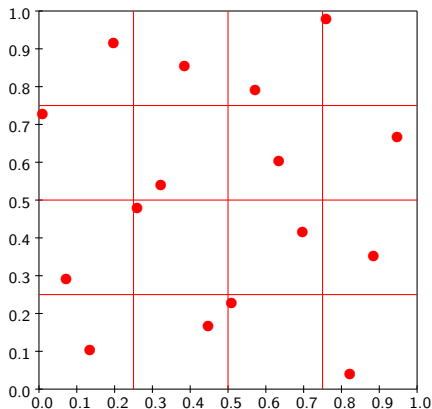


A (4,4) mapping

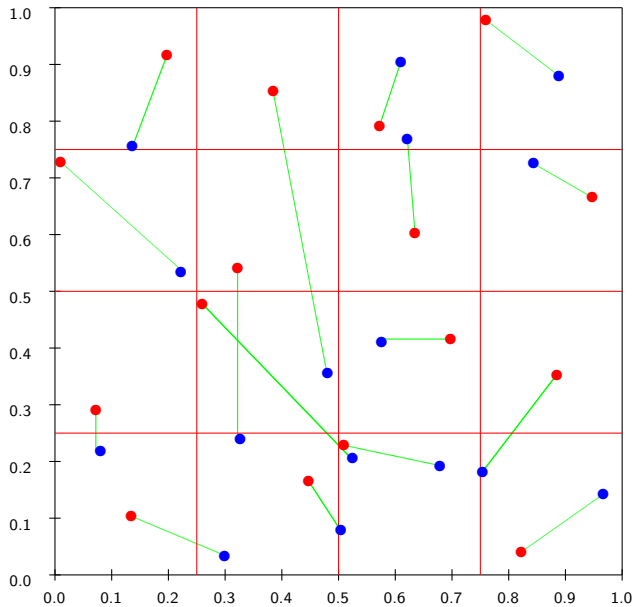
States of the chains



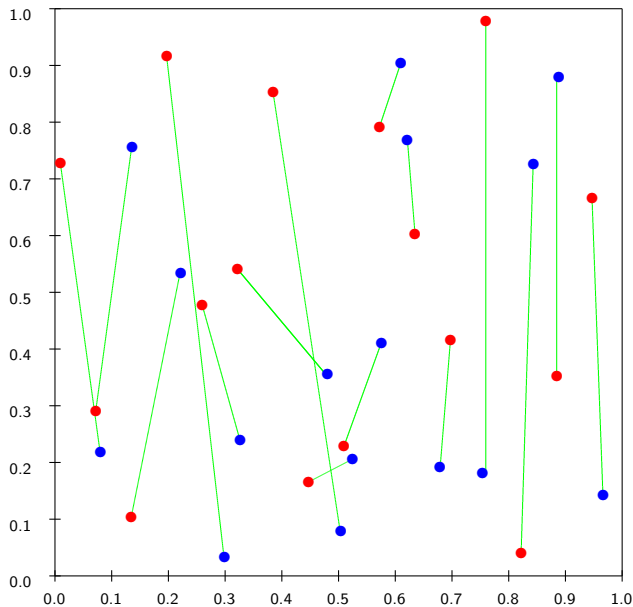
Sobol' Sequence in 2 dimensions with digital shift



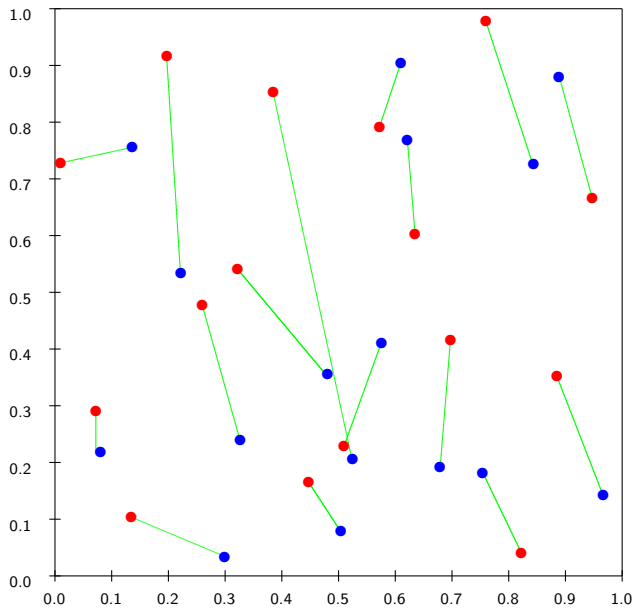
A (4,4) mapping



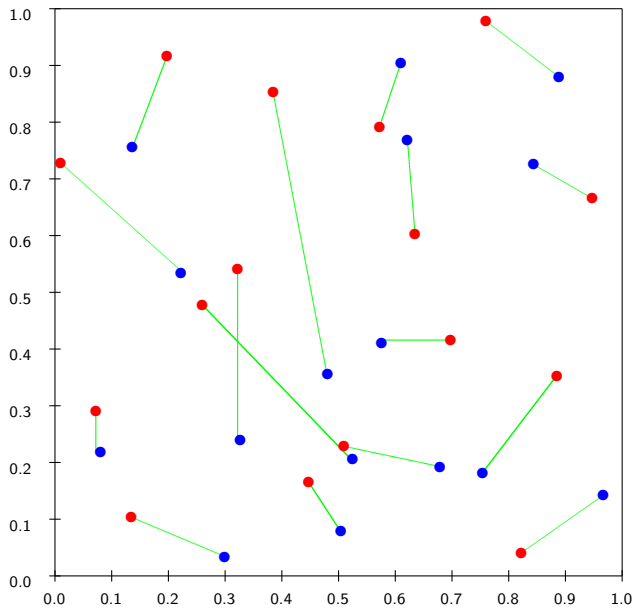
A (16,1) mapping, sorting along the first coordinate



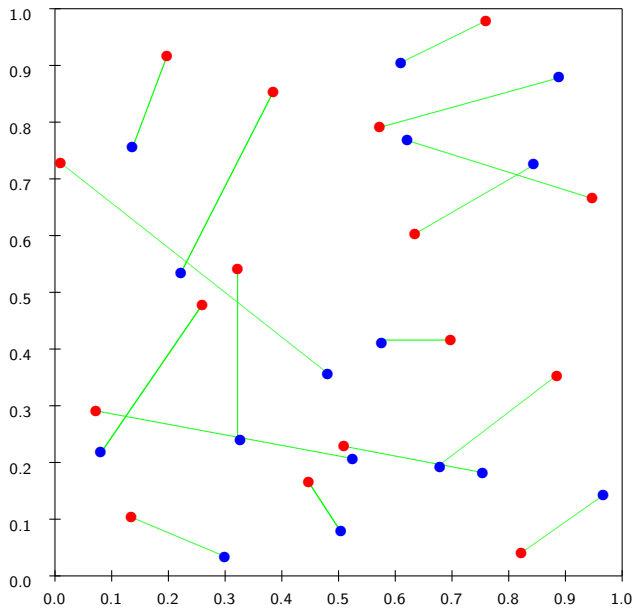
A (8,2) mapping



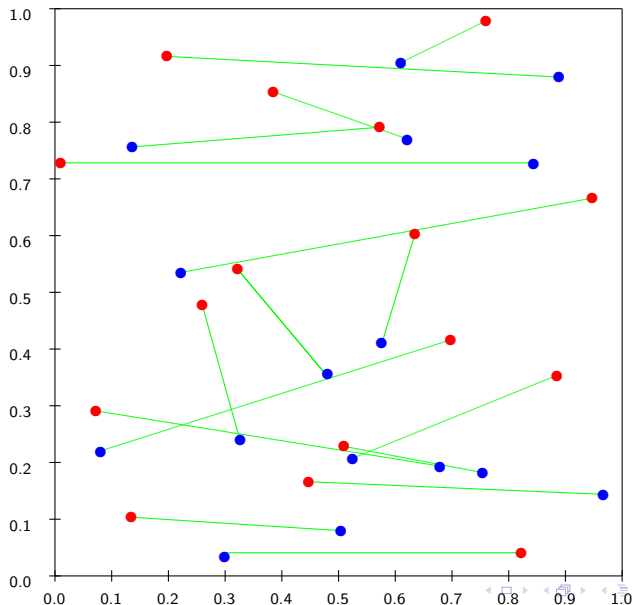
A (4,4) mapping



A (2,8) mapping



A (1,16) mapping, sorting along the second coordinate



$O(n^{-2})$ convergence for $\ell = d = 1$

Suppose D is the \mathcal{L}_2 star discrepancy, whose square is the CvM statistic

$$W^2 = \frac{1}{12n^2} + \frac{1}{n} \sum_{i=0}^{n-1} (w_i - x_i)^2$$

where $w_i = (i + 1/2)/n$. At best, $W^2 = O(n^{-2})$.

$O(n^{-2})$ convergence for $\ell = d = 1$

Suppose D is the \mathcal{L}_2 star discrepancy, whose square is the CvM statistic

$$W^2 = \frac{1}{12n^2} + \frac{1}{n} \sum_{i=0}^{n-1} (w_i - x_i)^2$$

where $w_i = (i + 1/2)/n$. At best, $W^2 = O(n^{-2})$.

Suppose

$$\mathbb{E} \left[\sup_x |\partial \varphi(x, u) / \partial x|^2 \right] \leq K.$$

$O(n^{-2})$ convergence for $\ell = d = 1$

Suppose D is the \mathcal{L}_2 star discrepancy, whose square is the CvM statistic

$$W^2 = \frac{1}{12n^2} + \frac{1}{n} \sum_{i=0}^{n-1} (w_i - x_i)^2$$

where $w_i = (i + 1/2)/n$. At best, $W^2 = O(n^{-2})$.

Suppose

$$\mathbb{E} \left[\sup_x |\partial \varphi(x, u) / \partial x|^2 \right] \leq K.$$

Induction proof: We assume that

- $\mathbb{E}[D_{j-1}^2] \leq \kappa_{j-1} n^{-2+\epsilon}$,
- the states at step $j - 1$ are sorted by increasing order,
- $X_{i,j} = \varphi(X_{i,j-1}, U_{i,j})$, $\tilde{X}_{i,j} = \varphi(w_i, U_{i,j})$,
- $\tilde{w}_0, \dots, \tilde{w}_{n-1}$ is a permutation of w_0, \dots, w_{n-1} that matches the ordering of $\tilde{X}_{0,j}, \dots, \tilde{X}_{n-1,j}$. Then,

$$\begin{aligned}
\mathbb{E}[D_j^2] &\leq \frac{1}{n} \mathbb{E} \left[\sum_{i=0}^{n-1} (\tilde{w}_i - X_{i,j})^2 \right] + \frac{1}{12n^2} \\
&\leq \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E} \left[(\tilde{w}_i - \tilde{X}_{i,j})^2 + (\tilde{X}_{i,j} - X_{i,j})^2 \right] + \frac{1}{12n^2} \\
&= \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E} \left[(\tilde{w}_i - \tilde{X}_{i,j})^2 + (\varphi(w_i, U_{i,j}) - \varphi(X_{i,j-1}, U_{i,j}))^2 \right] + \frac{1}{12n^2} \\
&\leq \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E} \left[(\tilde{w}_i - \tilde{X}_{i,j})^2 + K(w_i - X_{i,j-1})^2 \right] + \frac{1}{12n^2} \\
&\leq \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E} \left[(\tilde{w}_i - \tilde{X}_{i,j})^2 \right] + K \mathbb{E}[D_{j-1}^2] + \frac{1-K}{12n^2} \\
&\leq \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{E} \left[(\tilde{w}_i - \tilde{X}_{i,j})^2 \right] + K \kappa_{j-1} n^{-2+\epsilon} + \frac{1-K}{12n^2} \\
&\leq \mathbb{E}[\tilde{D}_j^2] + K \kappa_{j-1} n^{-2+\epsilon} - \frac{K}{12n^2}.
\end{aligned}$$

Observe that $\mathbb{E}[\tilde{D}_j^2]$ is the mean square discrepancy obtained when we integrate $\varphi(w, u)$ over $[0, 1]^2$ by an RQMC point set.

If φ is smooth enough, we can find an RQMC point set $\{(w_i, U_{i,j}), 0 \leq i < n\}$ such that $\mathbb{E}[\tilde{D}_j^2] = O(n^{-2+\epsilon})$.

Then, $\mathbb{E}[\tilde{D}_j^2] \leq \kappa n^{-2+\epsilon}$ for some κ , and therefore

$$\mathbb{E}[D_j^2] \leq (\kappa + K\kappa_{j-1})n^{-2+\epsilon} = \kappa_j n^{-2+\epsilon}$$

with

$$\kappa_j = \kappa + K\kappa_{j-1} = (1 + K + \dots + K^{j-1})\kappa = \frac{1 - K^j}{1 - K}\kappa.$$

For $j \rightarrow \infty$, this bound converges if and only if $K < 1$.

Example 1

Let $Y = \theta U + (1 - \theta)V$, where U, V indep. $U(0, 1)$ and $\theta \in [0, 1)$.
This Y has distribution function G_θ .

Our Markov chain is defined by

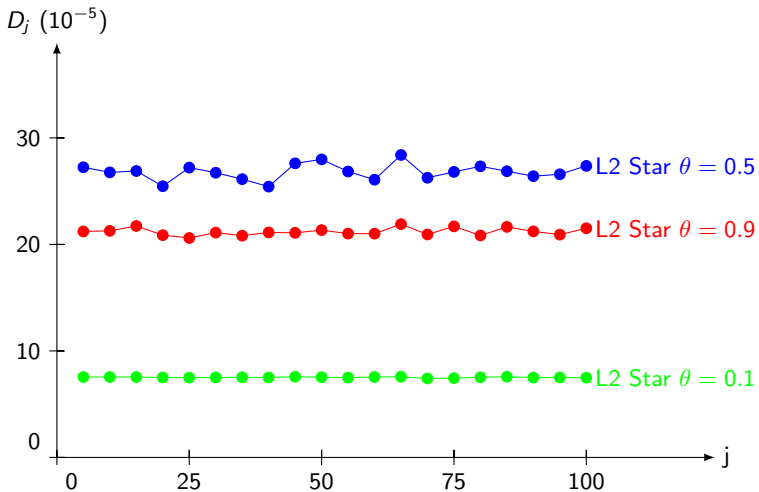
$$X_0 = U_0; \quad X_j = \varphi(X_{j-1}, U_j) = G_\theta(\theta X_{j-1} + (1 - \theta)U_j), j \geq 1$$

where $U_j \sim U(0, 1)$. Then, $X_j \sim (0, 1)$.

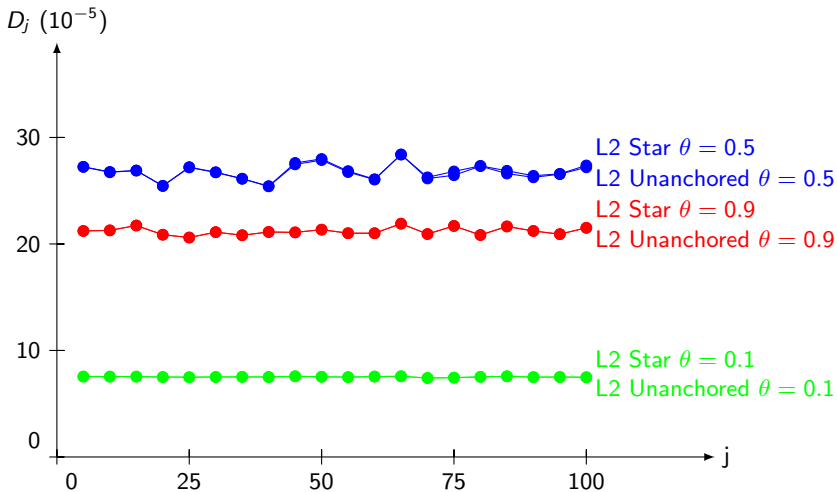
We also define $c_j(X_j) = X_j$

Condition holds with $K = 1$ if $\theta \geq 1/2$,
and $K = \theta^2/(1 - \theta)^2 < 1$ if $\theta < 1/2$.

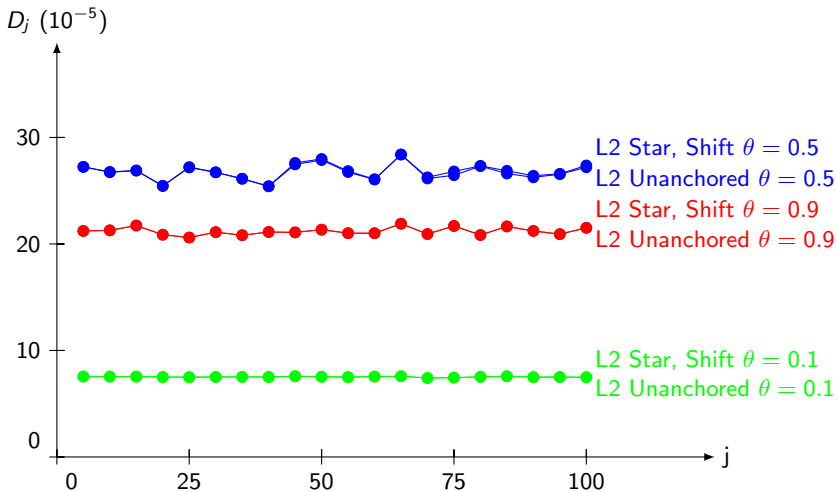
$\log D_j$ as a function of j , for $n = 4093 \approx 2^{12}$



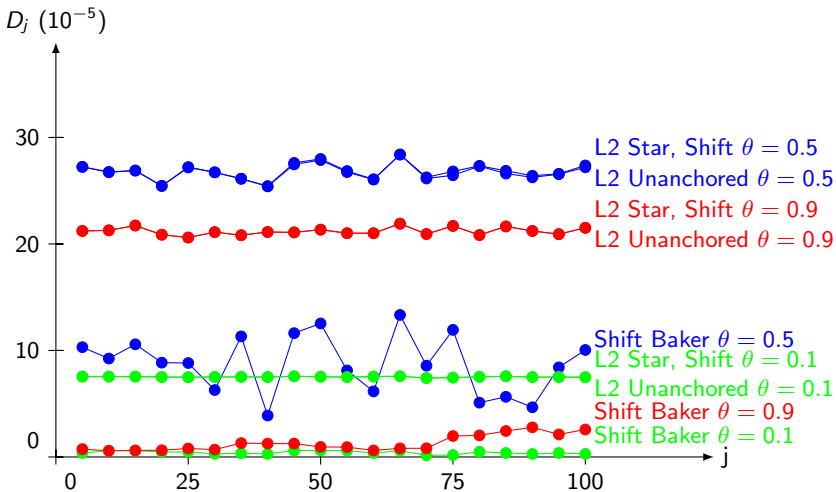
$\log D_j$ as a function of j , for $n = 4093 \approx 2^{12}$



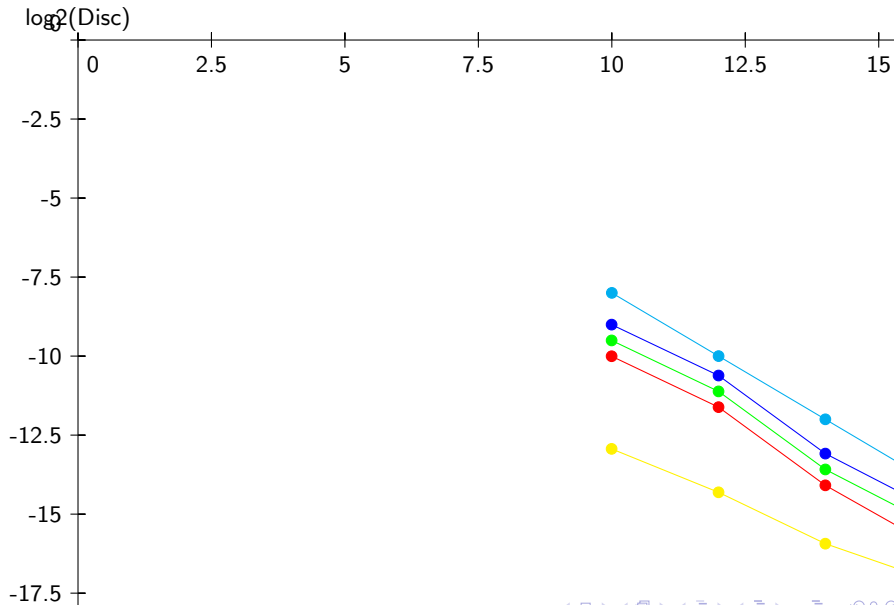
$\log D_j$ as a function of j , for $n = 4093 \approx 2^{12}$



$\log D_j$ as a function of j , for $n = 4093 \approx 2^{12}$

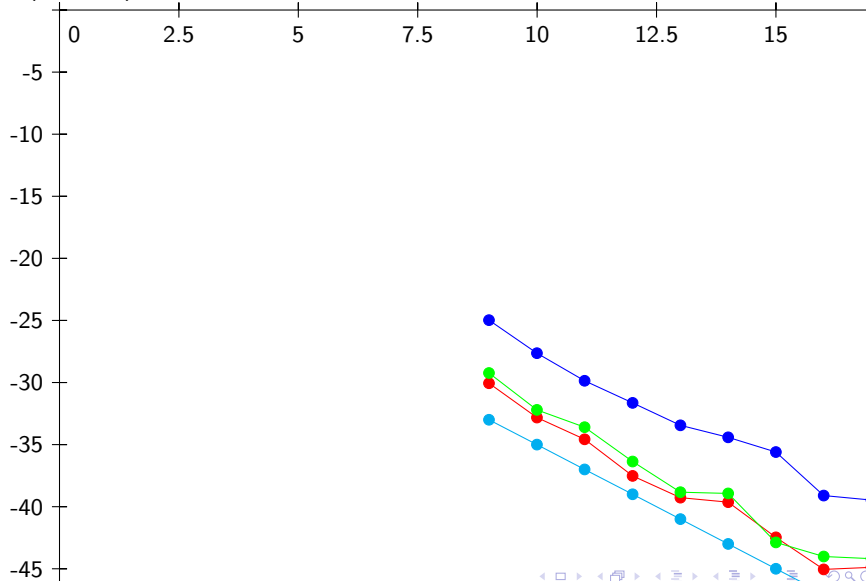


$\log_2 D_j$ as a function of $\log_2 n$



$\log_2 \text{Var}[\hat{\mu}_{\text{rqmc},j,n}]$ as a function of $\log_2 n$

$\log_2(\text{Variance})$



Example 2

Markov chain:

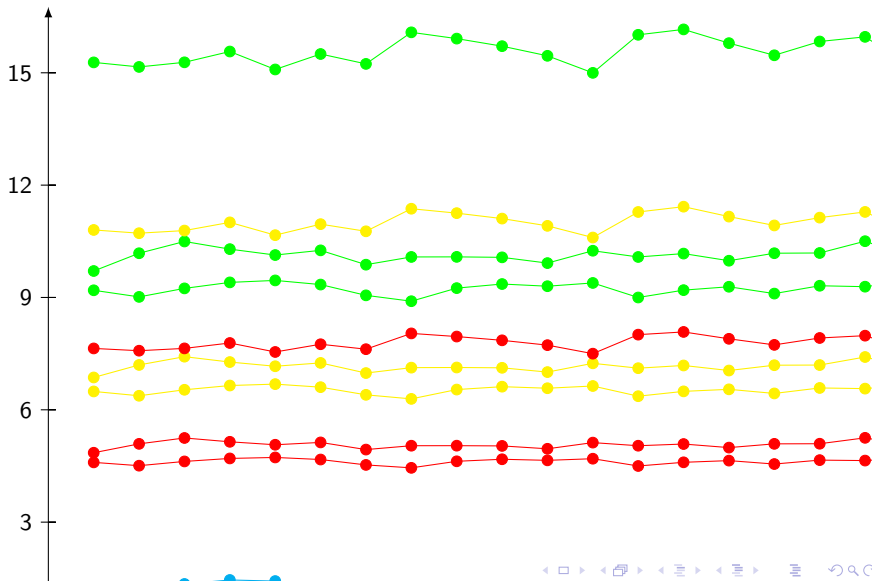
$$Y_0 = Z_0, \quad Y_j = (\rho Y_{j-1} + Z_j) / \sqrt{\rho^2 + 1}, j \geq 1,$$

where $Z_j \sim N(0, 1)$. Then, $Y_j \sim N(0, 1)$ and $X_j = \Phi(Y_j) \sim U(0, 1)$.

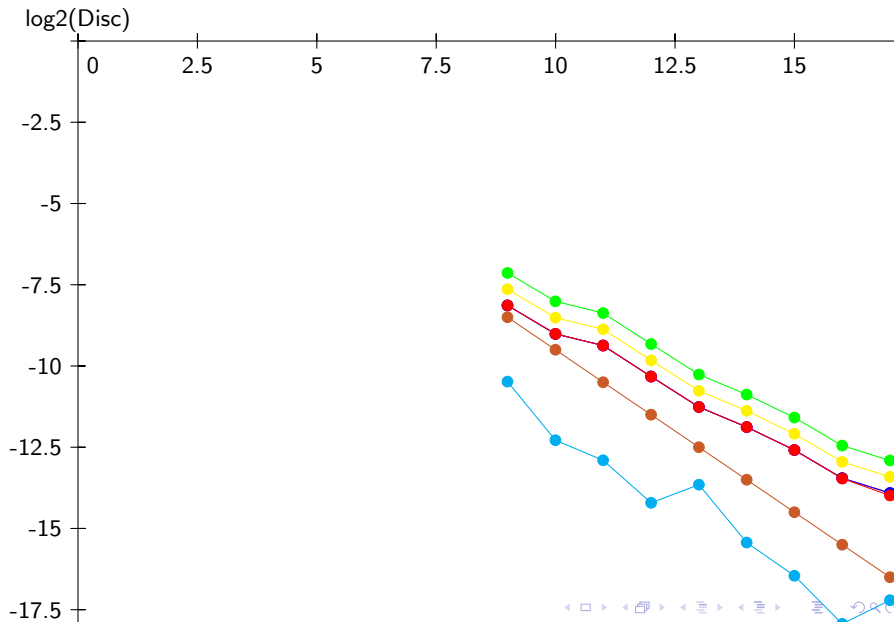
We define also, $c_j(X_j) = X_j$.

$\log_2 D_j$ as a function of j , for $n = 4093$

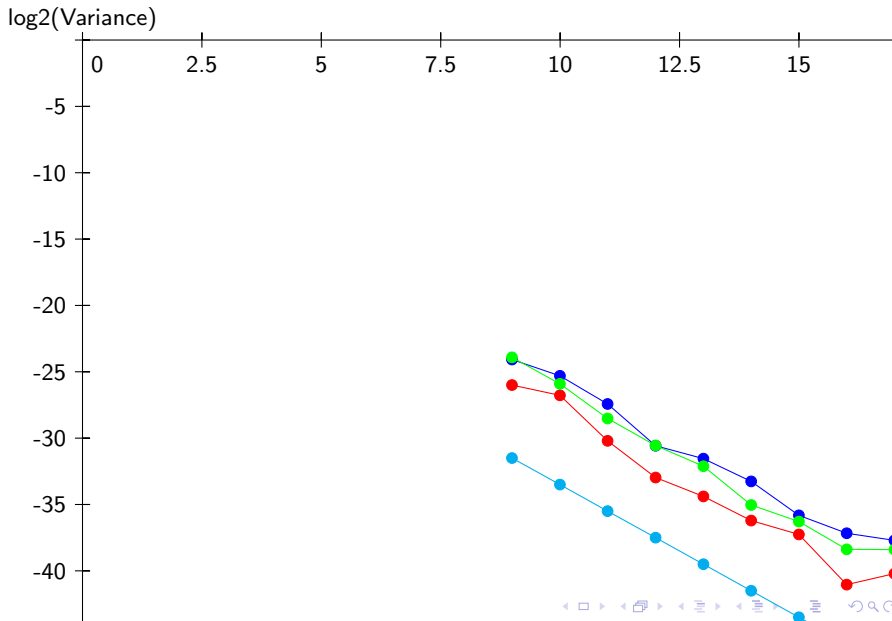
Disc (10^{-4})



$\log_2 D_j$ as a function of $\log_2 n$



$\log_2 \text{Var}[\hat{\mu}_{\text{rqmc},j,n}]$ as a function of $\log_2 n$



Example 3: Asian Option

Given observation times $t_1, t_2, \dots, t_s = T$, we define

$$S(t_j) = S(t_{j-1}) \exp[(r - \sigma^2/2)(t_j - t_{j-1}) + \sigma(t_j - t_{j-1})^{1/2} \Phi^{-1}(U_j)],$$

where $U_j \sim U[0, 1)$ and $S(t_0) = s_0$ is fixed.

State is $(S(t_j), \bar{S}_j)$, where $\bar{S}_j = \frac{1}{j} \sum_{i=1}^j S(t_i)$.

Transition:

$$(S(t_j), \bar{S}_j) = \varphi(S(t_{j-1}), \bar{S}_{j-1}, U_j) = \left(S(t_j), \frac{(j-1)\bar{S}_{j-1} + S(t_j)}{j} \right),$$

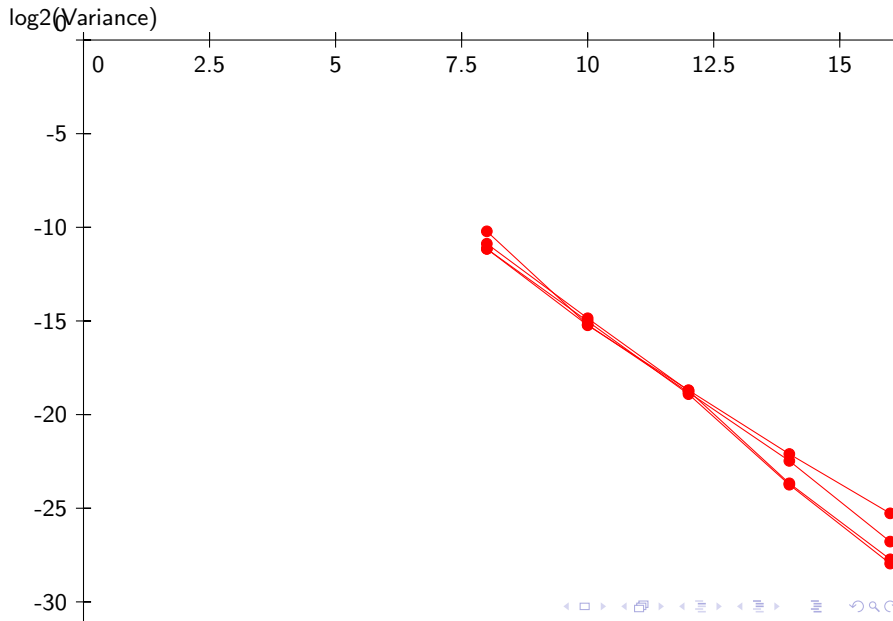
Payoff is

$$\max [0, (\bar{S}_s) - K].$$

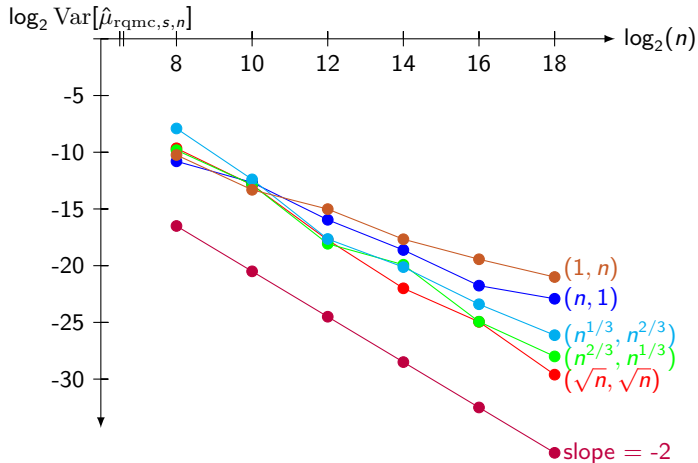
For the example, we used $S(0) = 100$, $K = 90$, $T = 240/360$,
 $t_j - t_{j-1} = 1/365$, $r = \ln 1.09$, $\sigma = 0.2$ and $s = 10, 60$.

Here, we use the two-dimensional sort at each step; we first sort in n_1 packets based on $S(t_j)$, then sort the packets based on \bar{S}_j .

$\log_2 \text{Var}[\hat{\mu}_{\text{rqmc},j,n}]$ as a function of $\log_2 n$, for $s = 10$



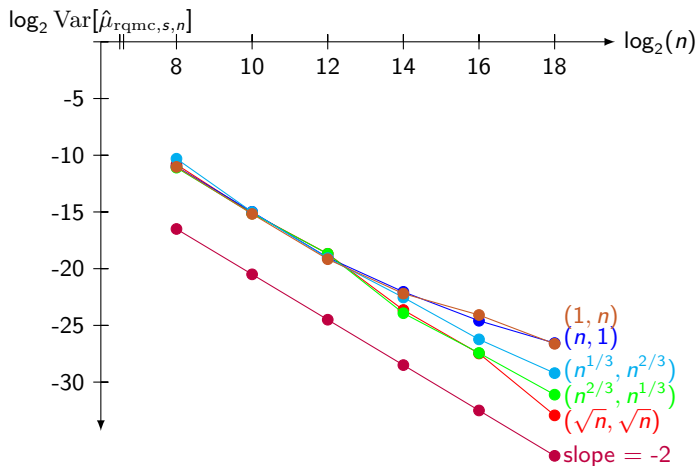
$\log_2 \text{Var}[\hat{\mu}_{\text{rqmc},j,n}]$ as a function of $\log_2 n$, for $s = 60$



Linear regression slopes

$(1, n)$	-1.29
(\sqrt{n}, \sqrt{n})	-2.00

$\log_2 \text{Var}$ as a function of $\log_2 n$, for $s = 10$, state = $(\bar{S}_j, S(t_j))$



Linear regression slopes

$(1, n)$	-1.54
(\sqrt{n}, \sqrt{n})	-2.18

Future Work

Convergence proof in the multidimensional case. Choice of discrepancy.

More experiments : larger examples, sorting strategies, etc.

Applications.