# Richardson extrapolation and higher order QMC

Takashi Goda

School of Engineering, University of Tokyo

MCM 2019

Partly joint with Josef Dick and Takehito Yoshiki

# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# Quasi-Monte Carlo

- Approximate

$$I(f) = \int_{[0,1)^s} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x},$$

where $f\colon [0,1)^s \to \mathbb{R}$ is integrable.

- Choose an $N$-element point set $P \subset [0,1)^s$ and approximate $I(f)$ by

$$Q_P(f) = \frac{1}{N}\sum_{\boldsymbol{x}\in P} f(\boldsymbol{x}).$$

This is called a quasi-Monte Carlo (QMC) integration over $P$.

# Digital nets in base 2

- Let $m, n \in \mathbb{N} := \{1, 2, \ldots\}$.
- Let $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$ ($\mathbb{F}_2 := \{0, 1\}$, the two-element field).

# Digital nets in base 2

- Let $m, n \in \mathbb{N} := \{1, 2, \ldots\}$.
- Let $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$ ($\mathbb{F}_2 := \{0, 1\}$, the two-element field).
- Denote the dyadic expansion of $0 \leq h < 2^m$ by

$$h = (\eta_{m-1} \ldots \eta_1 \eta_0)_2.$$

# Digital nets in base 2

- Let $m, n \in \mathbb{N} := \{1, 2, \ldots\}$.
- Let $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$ ($\mathbb{F}_2 := \{0, 1\}$, the two-element field).
- Denote the dyadic expansion of $0 \le h < 2^m$ by

$$h = (\eta_{m-1} \ldots \eta_1 \eta_0)_2.$$

For $1 \le j \le s$, let

$$x_j = (0.\xi_1^{(j)} \ldots \xi_n^{(j)} 00 \ldots)_2,$$

where

$$\begin{pmatrix} \xi_1^{(j)} \\ \vdots \\ \xi_n^{(j)} \end{pmatrix} = C_j \begin{pmatrix} \eta_0 \\ \vdots \\ \eta_{m-1} \end{pmatrix} \in \mathbb{F}_2^n.$$

This gives a point $\boldsymbol{x} = (x_1, \ldots, x_s) \in [0, 1)^s$.

# Digital nets in base 2

- Let $m, n \in \mathbb{N} := \{1, 2, \ldots\}$.
- Let $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$ ($\mathbb{F}_2 := \{0, 1\}$, the two-element field).
- Denote the dyadic expansion of $0 \le h < 2^m$ by

$$h = (\eta_{m-1} \ldots \eta_1 \eta_0)_2.$$

  For $1 \le j \le s$, let

$$x_j = (0.\xi_1^{(j)} \ldots \xi_n^{(j)} 00 \ldots)_2,$$

  where

$$\begin{pmatrix} \xi_1^{(j)} \\ \vdots \\ \xi_n^{(j)} \end{pmatrix} = C_j \begin{pmatrix} \eta_0 \\ \vdots \\ \eta_{m-1} \end{pmatrix} \in \mathbb{F}_2^n.$$

  This gives a point $\boldsymbol{x} = (x_1, \ldots, x_s) \in [0, 1)^s$.
- Running through all $0 \le h < 2^m$, we get a $2^m$-element point set $P$ called a *digital net* over $\mathbb{F}_2$.

# Digital sequences in base 2

- Let $C_1, \ldots, C_s \in \mathbb{F}_2^{\mathbb{N}\times\mathbb{N}}$, where every column is assumed to have only finite non-zero elements.
- Denote the dyadic expansion of $h \in \mathbb{N}_0 := \mathbb{N} \cup \{0\}$ by

$$h = (\ldots \eta_1 \eta_0)_2.$$

where all but finite number of $\eta_i$ are $0$. For $1 \le j \le s$, let

$$x_j = (0.\xi_1^{(j)} \xi_2^{(j)} \ldots)_2,$$

where

$$\begin{pmatrix} \xi_1^{(j)} \\ \xi_2^{(j)} \\ \vdots \end{pmatrix} = C_j \begin{pmatrix} \eta_0 \\ \eta_1 \\ \vdots \end{pmatrix} \in \mathbb{F}_2^{\mathbb{N}}.$$

This gives a point $\boldsymbol{x} = (x_1, \ldots, x_s) \in [0,1)^s$.

- This way we get an infinite sequence of points $\mathcal{S}$ called a *digital sequence* over $\mathbb{F}_2$.

# Some remarks

- "Goodness" (undefined) of digital nets/sequences is determined by generating matrices $C_1, \ldots, C_s$.
- For nets, $m$ determines total number of points, while $n$ determines precision of each point.
- For sequences, the assumption that every column has only finite non-zero elements is to ensure that each point has finite precision.

# Precision?

- For classical QMC where "goodness" is measured by *discrepancy*, it is usual to set $n = m$ (square matrices) for nets, or consider upper-triangular matrices for sequences.
- For higher order QMC where "goodness" is measured by the worst-case error for a class of smooth functions, we need higher precision. Typically, $n = \alpha m$ for nets ($\alpha$: dominating mixed smoothness of functions).

# Precision?

- For classical QMC where "goodness" is measured by *discrepancy*, it is usual to set $n = m$ (square matrices) for nets, or consider upper-triangular matrices for sequences.
- For higher order QMC where "goodness" is measured by the worst-case error for a class of smooth functions, we need higher precision. Typically, $n = \alpha m$ for nets ($\alpha$: dominating mixed smoothness of functions).

**Aim of this talk**

is to show that Richardson extrapolation can be used as a technique to reduce necessary precision of higher order QMC from $\alpha m$ to $m$.

# Precision?

- For classical QMC where "goodness" is measured by *discrepancy*, it is usual to set $n = m$ (square matrices) for nets, or consider upper-triangular matrices for sequences.
- For higher order QMC where "goodness" is measured by the worst-case error for a class of smooth functions, we need higher precision. Typically, $n = \alpha m$ for nets ($\alpha$: dominating mixed smoothness of functions).

**Aim of this talk**

is to show that Richardson extrapolation can be used as a technique to reduce necessary precision of higher order QMC from $\alpha m$ to $m$.

**Implication**

- Possibility to try large values of $\alpha$ w/o suffering from round-off error.
- Search space for $C_1, \ldots, C_s$ is significantly reduced.

# Outline

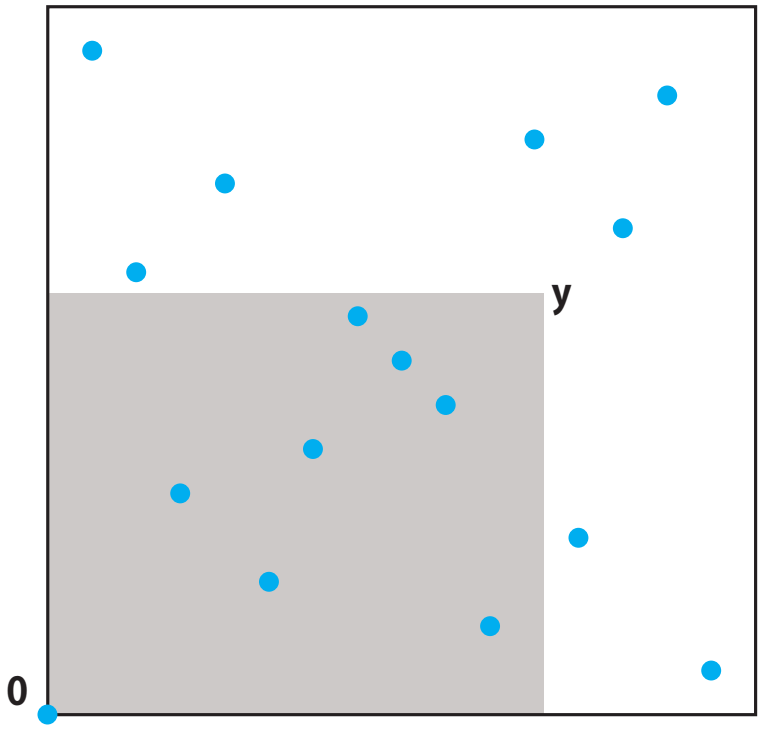
- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# Local discrepancy function

- For an $N$-element point set $P$ and $\boldsymbol{y} \in [0,1)^s$, let

$$\Delta_P(\boldsymbol{y}) := \frac{1}{N} \sum_{\boldsymbol{x} \in P} \mathbf{1}_{\boldsymbol{x} \in [\mathbf{0}, \boldsymbol{y})} - \lambda([\mathbf{0}, \boldsymbol{y})),$$

where $\lambda$ denotes the Lebesgue measure.



# Discrepancy and Koksma-Hlawka inequality

- For $1 \le p \le \infty$, the $L_p$-discrepancy of $P$ is defined by

$$L_p(P) := \left( \int_{[0,1)^s} |\Delta_P(\boldsymbol{y})|^p \, \mathrm{d}\boldsymbol{y} \right)^{1/p}.$$

When $p = \infty$, the obvious modification is needed.

# Discrepancy and Koksma-Hlawka inequality

- For $1 \leq p \leq \infty$, the $L_p$-discrepancy of $P$ is defined by

$$L_p(P) := \left( \int_{[0,1)^s} |\Delta_P(\boldsymbol{y})|^p \, \mathrm{d}\boldsymbol{y} \right)^{1/p}.$$

  When $p = \infty$, the obvious modification is needed.

- The integration error is bounded by

$$|Q_P(f) - I(f)| \leq V_{\mathrm{HK}}(f) L_\infty(P)$$

  where $V_{\mathrm{HK}}(f)$ is the Hardy-Krause variation of $f$.

# Discrepancy and Koksma-Hlawka inequality

- For $1 \le p \le \infty$, the $L_p$-discrepancy of $P$ is defined by

$$L_p(P) := \left( \int_{[0,1)^s} |\Delta_P(\boldsymbol{y})|^p \, \mathrm{d}\boldsymbol{y} \right)^{1/p}.$$

  When $p = \infty$, the obvious modification is needed.

- The integration error is bounded by

$$|Q_P(f) - I(f)| \le V_{\mathrm{HK}}(f) L_\infty(P)$$

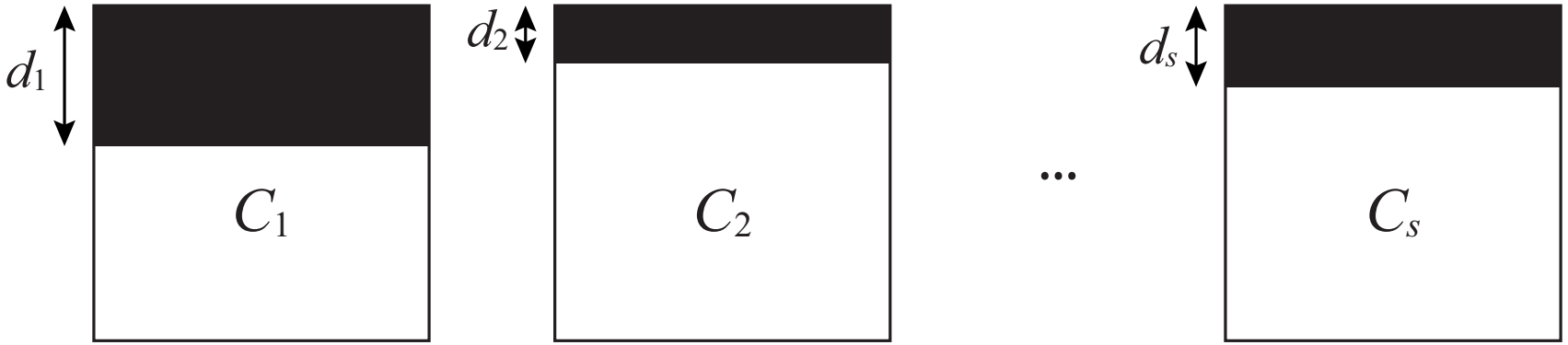
  where $V_{\mathrm{HK}}(f)$ is the Hardy-Krause variation of $f$.

**Main focus on classical QMC**

How to construct digital nets/sequences with small $L_\infty(P)$.

# Quality measure: $t$-value

- Let $t$ be an integer such that, for any choice $d_1, \ldots, d_s \geq 0$ with $d_1 + \cdots + d_s = m - t$,
  the first $d_1$ rows of $C_1$
  $\vdots$
  the first $d_s$ rows of $C_s$
  are linearly independent over $\mathbb{F}_2$. $P$ is called a *digital* $(t, m, s)$*-net*.



# Quality measure: $t$-value

- An equi-distribution property of digital $(t, m, s)$-nets: every dyadic elementary box of the form

$$\prod_{j=1}^{s} \left[ \frac{a_j}{2^{c_j}}, \frac{a_j + 1}{2^{c_j}} \right) \quad \text{with} \quad \begin{cases} c_1, \ldots, c_s \geq 0, \\ c_1 + \cdots + c_s = m - t, \\ 0 \leq a_j < 2^{c_j} \end{cases}$$

contains exactly $2^t$ points.

# Quality measure: $t$-value

- An equi-distribution property of digital $(t, m, s)$-nets: every dyadic elementary box of the form

$$\prod_{j=1}^{s} \left[ \frac{a_j}{2^{c_j}}, \frac{a_j + 1}{2^{c_j}} \right) \quad \text{with} \quad \begin{cases} c_1, \ldots, c_s \geq 0, \\ c_1 + \cdots + c_s = m - t, \\ 0 \leq a_j < 2^{c_j} \end{cases}$$

  contains exactly $2^t$ points.

- Let $t$ be an integer such that, for any $m \geq t$, the first $2^m$ points of $\mathcal{S}$ are a digital $(t, m, s)$-net. $\mathcal{S}$ is called a *digital* $(t, s)$*-sequence*.

# $t$-value and star-discrepancy

- If $P$ is a digital $(t, m, s)$-net over $\mathbb{F}_2$,

$$L_\infty(P) \leq C_{s,t} \frac{(\log N)^{s-1}}{N} \quad \text{with} \quad N = 2^m.$$

- If $P$ is the first $N$ points of a digital $(t, s)$-sequence over $\mathbb{F}_2$,

$$L_\infty(P) \leq D_{s,t} \frac{(\log N)^s}{N},$$

for all $N \geq 2$.

# Explicit construction

- For $s = 1$, $C_1$ can be the identity matrix, which generates the famous digital $(0, 1)$-sequence called *van der Corput sequence*.

# Explicit construction

- For $s = 1$, $C_1$ can be the identity matrix, which generates the famous digital $(0, 1)$-sequence called *van der Corput sequence*.

- For $s = 2$, $C_1$ and $C_2$ can be

$$C_1 = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}, C_2 = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ 0 & \cdots & 1 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 1 & \cdots & 0 & 0 \end{pmatrix},$$

  which generates a digital $(0, m, 2)$-net, known as the *Hammersley* point set. Not extensible in $m$.

# Explicit construction

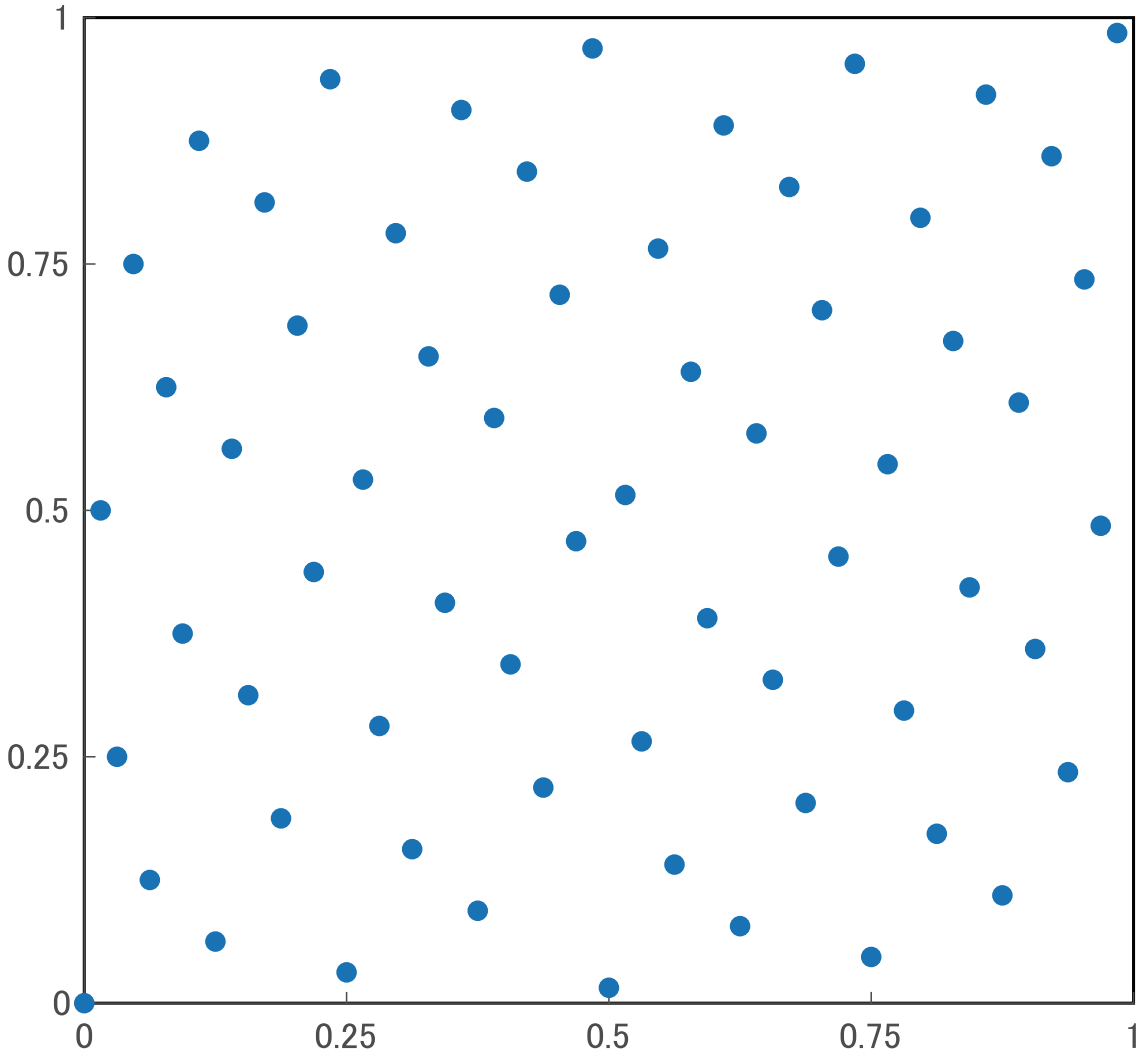



Figure: Hammersley point set for $m = 6$

# Explicit construction

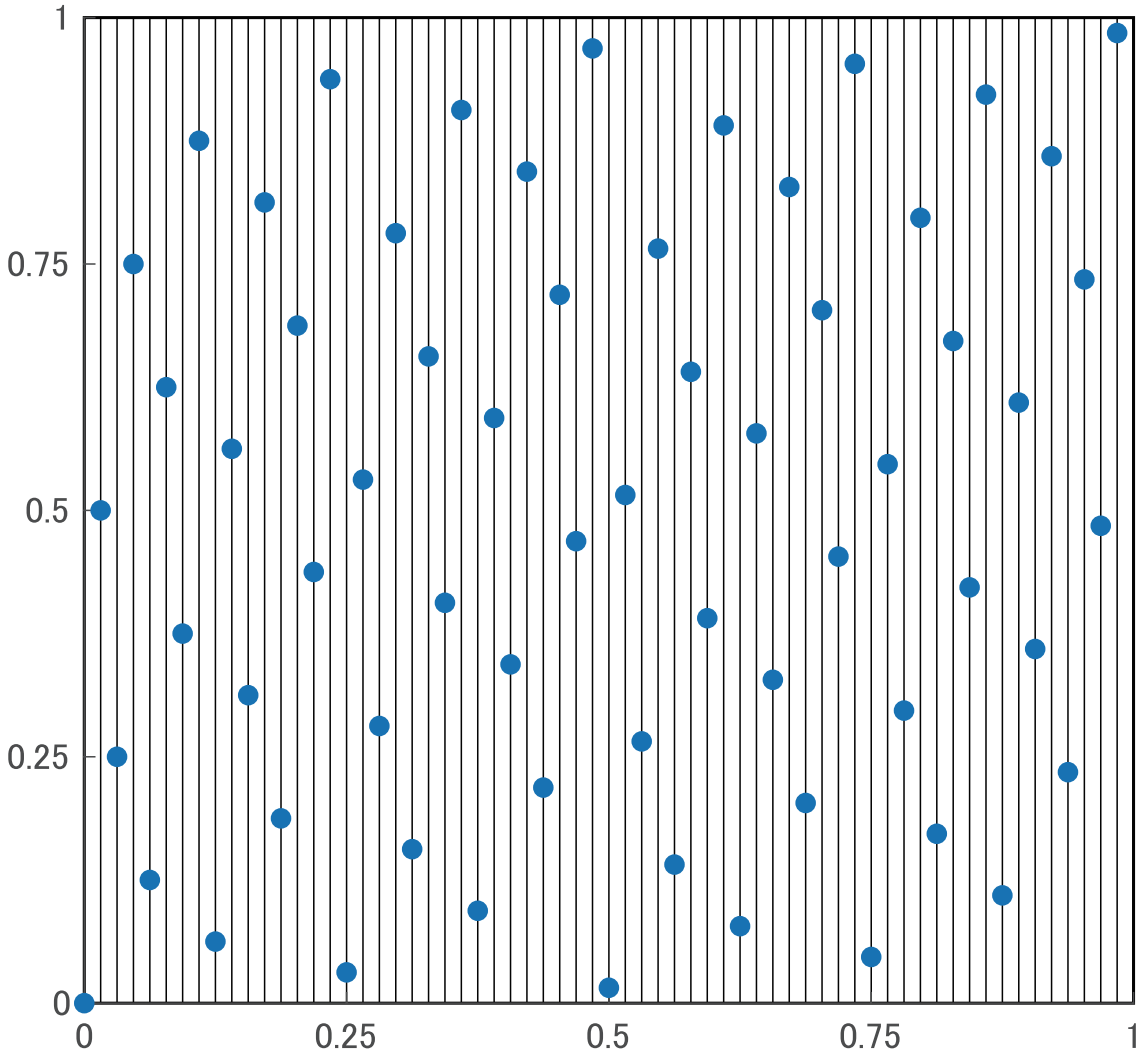



Figure: Hammersley point set for $m = 6$

# Explicit construction

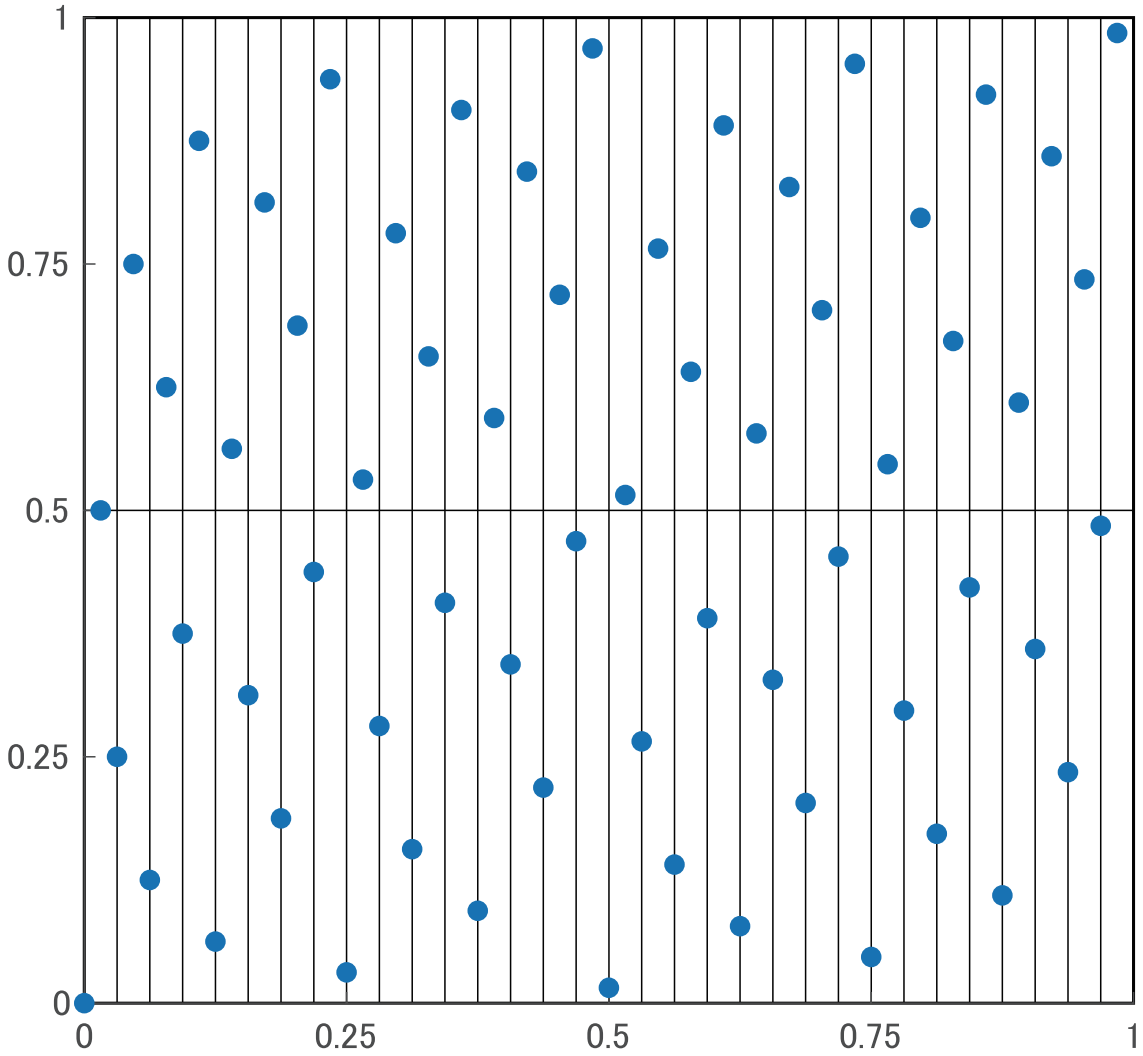



Figure: Hammersley point set for $m = 6$

# Explicit construction

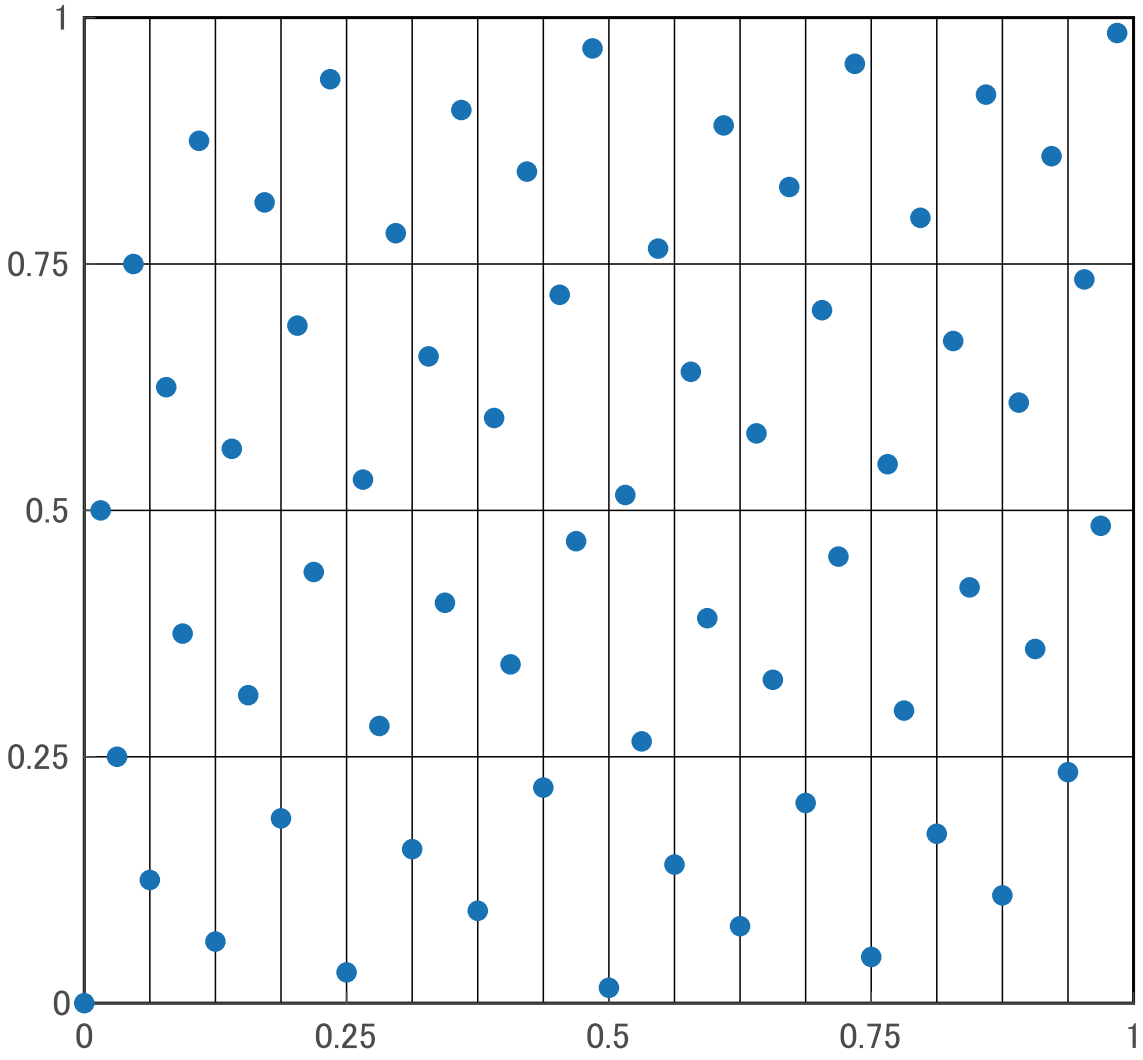



Figure: Hammersley point set for $m = 6$

# Explicit construction

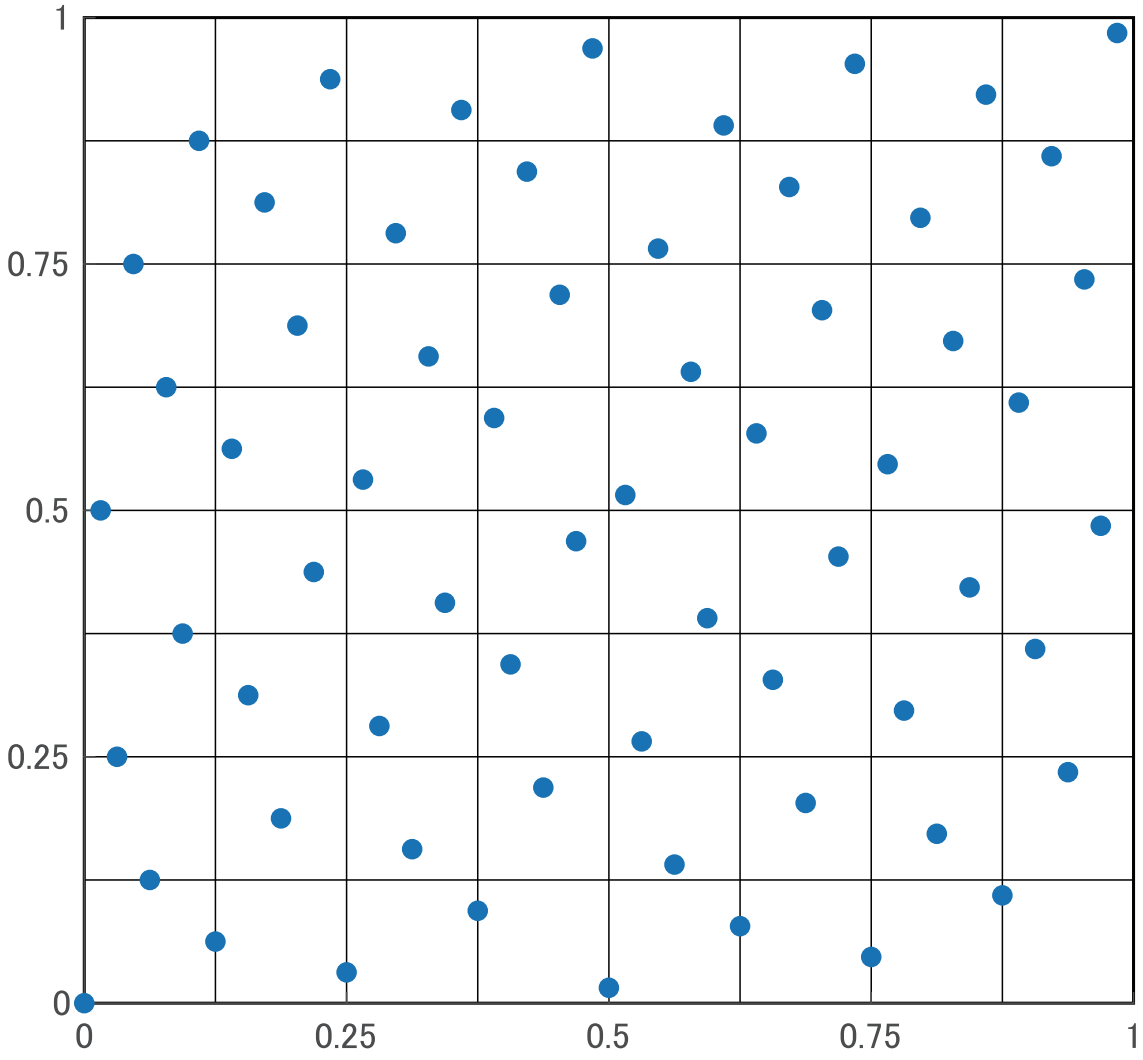



Figure: Hammersley point set for $m = 6$

# Explicit construction

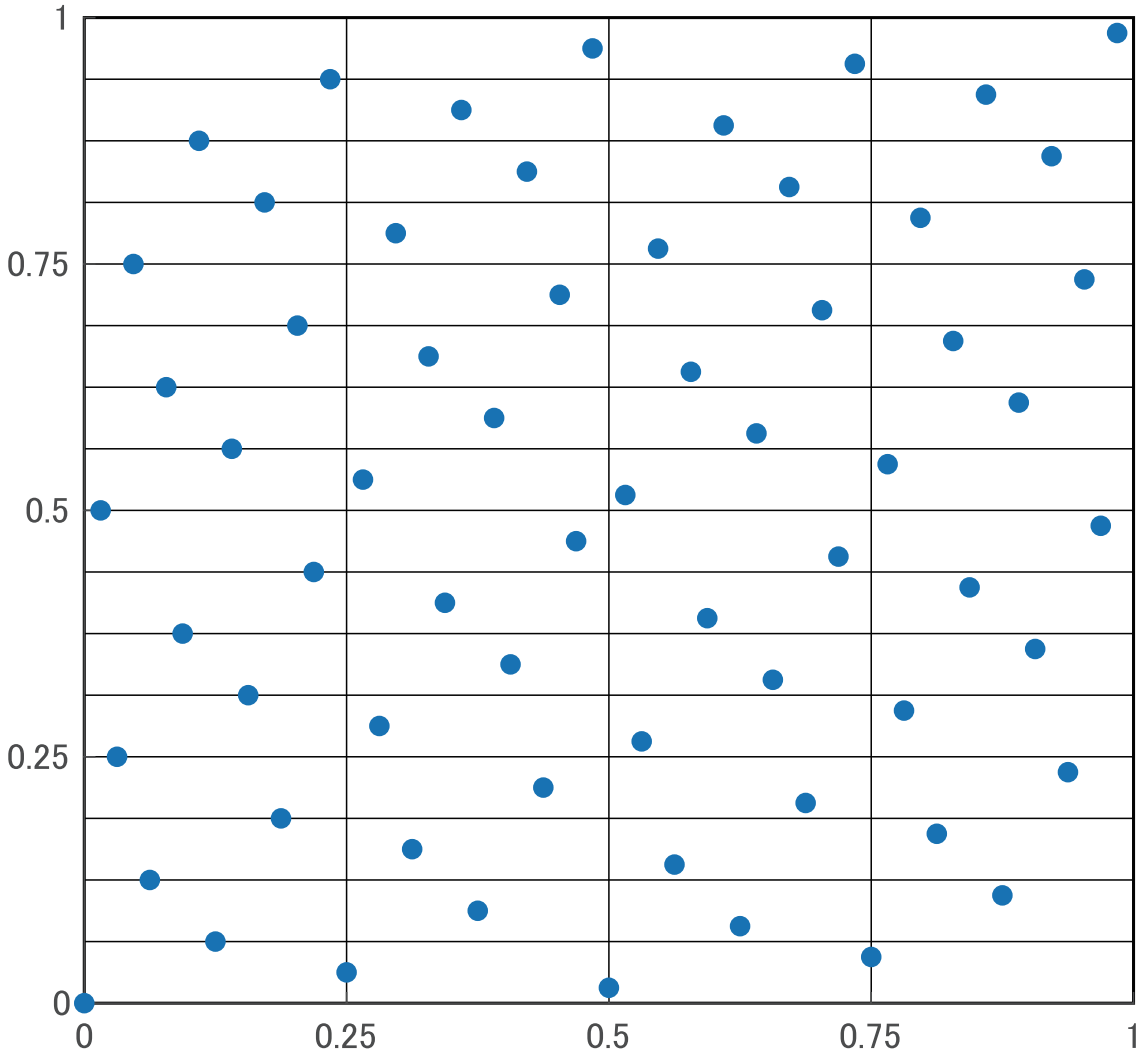



Figure: Hammersley point set for $m = 6$

# Explicit construction

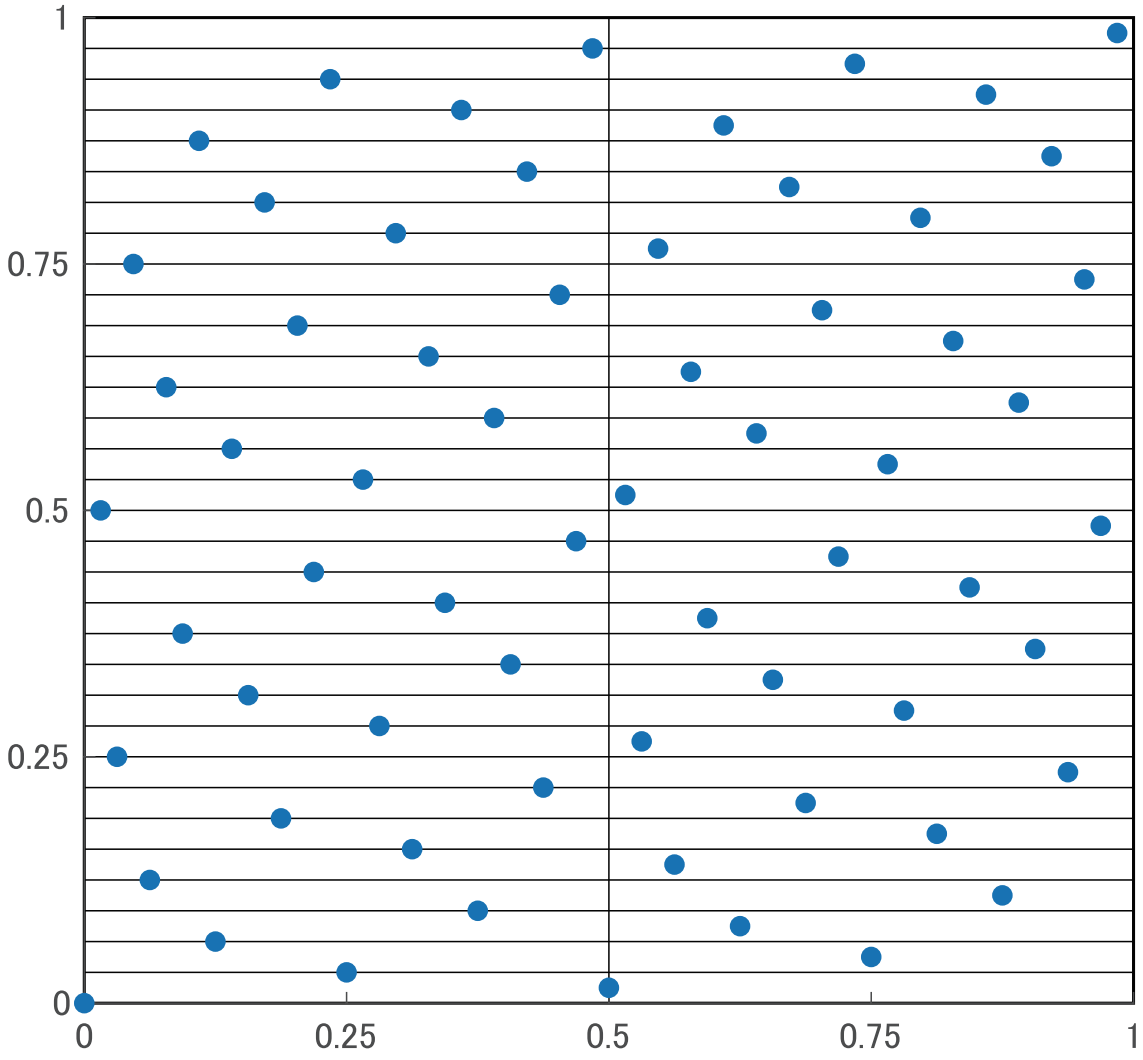



Figure: Hammersley point set for $m = 6$

# Explicit construction

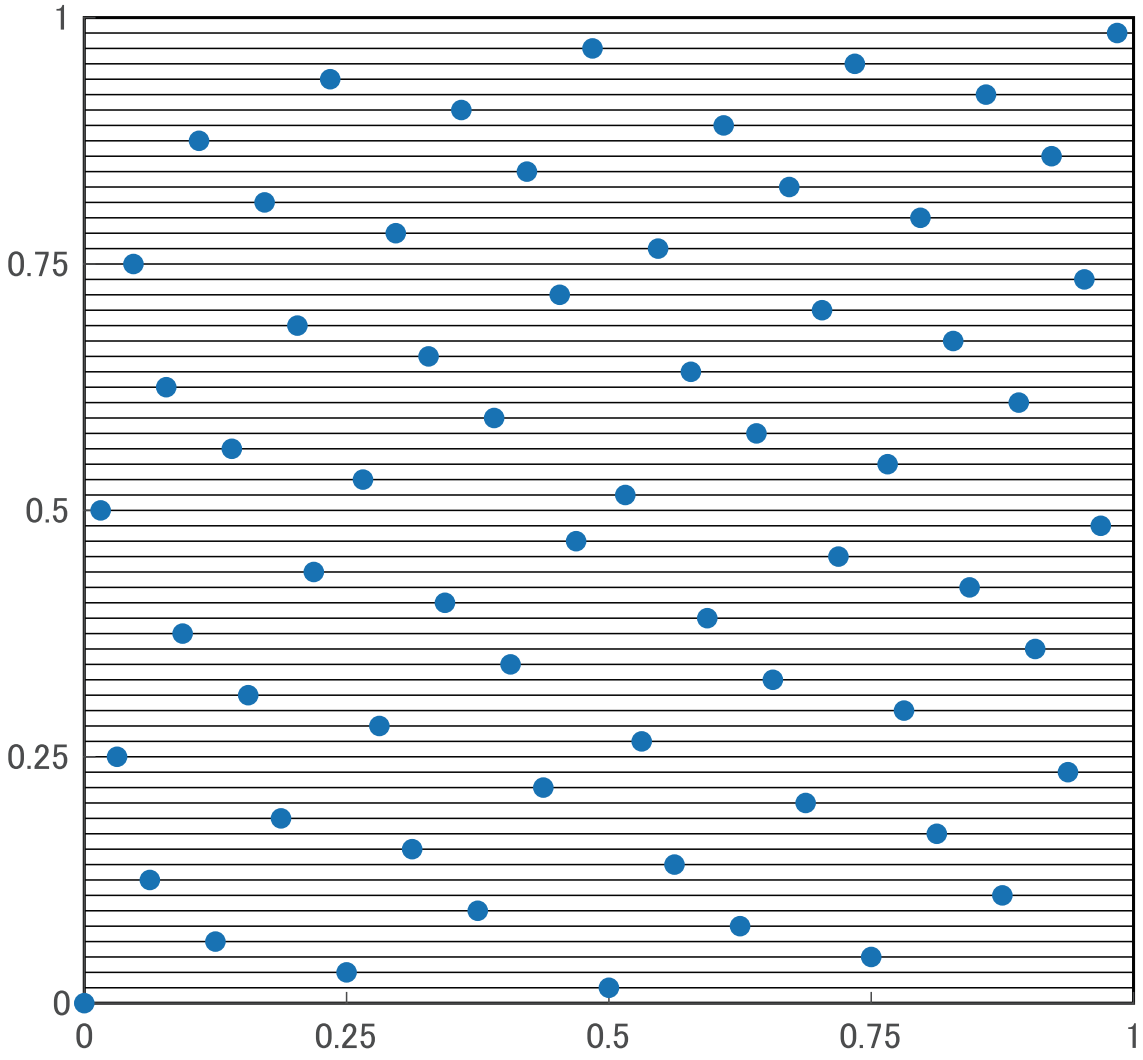



Figure: Hammersley point set for $m = 6$

# Explicit construction

- Let $p_1, p_2, \ldots \in \mathbb{F}_2[x]$ be a sequence of distinct primitive/irreducible polynomials over $\mathbb{F}_2$ with $e_1 \leq e_2 \leq \cdots$ where $e_j = \deg(p_j)$.

# Explicit construction

- Let $p_1, p_2, \ldots \in \mathbb{F}_2[x]$ be a sequence of distinct primitive/irreducible polynomials over $\mathbb{F}_2$ with $e_1 \leq e_2 \leq \cdots$ where $e_j = \deg(p_j)$.
- For each $j$, $C_j = (c_{k,l}^{(j)})$ is given by the coefficients of the following Laurent series:

$$\frac{x^{e_j-1}}{p_j(x)} = \frac{c_{1,1}^{(j)}}{x} + \frac{c_{1,2}^{(j)}}{x^2} + \cdots$$

$$\vdots$$

$$\frac{1}{p_j(x)} = \frac{c_{e_j,1}^{(j)}}{x} + \frac{c_{e_j,2}^{(j)}}{x^2} + \cdots$$

$$\frac{x^{e_j-1}}{(p_j(x))^2} = \frac{c_{e_j+1,1}^{(j)}}{x} + \frac{c_{e_j+1,2}^{(j)}}{x^2} + \cdots$$

$$\vdots$$

(Sobol' 1967; Niederreiter 1988; Tezuka 1993)

# Explicit construction

- All matrices $C_j \in \mathbb{F}_2^{\mathbb{N}\times\mathbb{N}}$ are upper triangular and generate a digital $(t, s)$-sequence with $t = (e_1 - 1) + \cdots + (e_s - 1)$.
  (I used a C implementation for this sequence in at least more than 58636 dimensions due to Tomohiko Hironaka.)

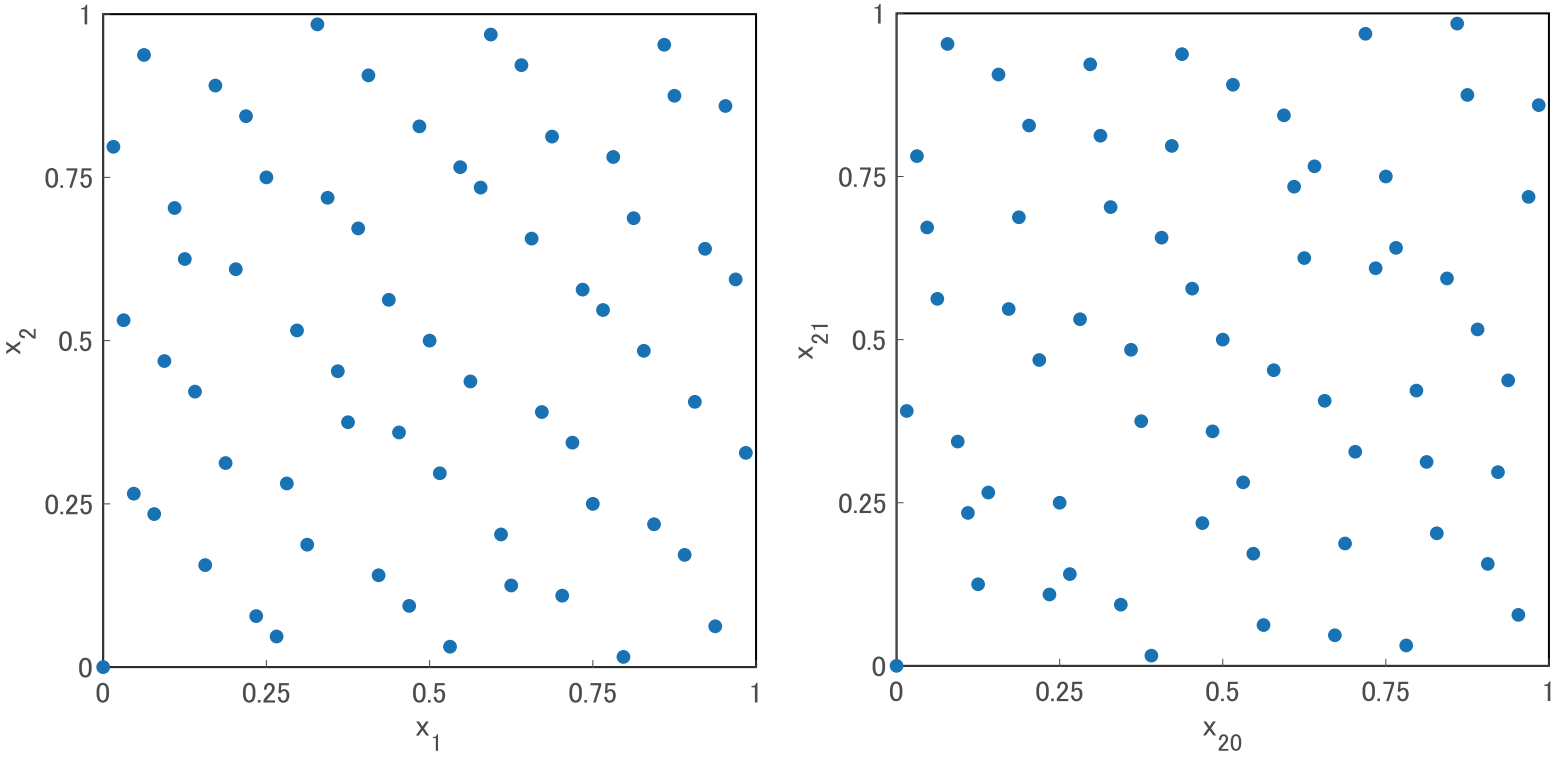



Figure: 2D projections of the first $2^6$ points of the Niederreiter sequence.

# Polynomial lattice point sets (Niederreiter, 1992)

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$
- Let $\boldsymbol{q} = (q_1, \ldots, q_s) \in (\mathbb{F}_2[x])^s$ with $\deg(q_j) < m$.

# Polynomial lattice point sets (Niederreiter, 1992)

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$
- Let $\boldsymbol{q} = (q_1, \ldots, q_s) \in (\mathbb{F}_2[x])^s$ with $\deg(q_j) < m$. For each $j$, $C_j$ is given by the square Hankel matrix

$$C_j = \begin{pmatrix} a_1^{(j)} & a_2^{(j)} & \cdots & a_m^{(j)} \\ a_2^{(j)} & a_3^{(j)} & \cdots & a_{m+1}^{(j)} \\ \vdots & \vdots & \ddots & \vdots \\ a_m^{(j)} & a_{m+1}^{(j)} & \cdots & a_{2m-1}^{(j)} \end{pmatrix} \in \mathbb{F}_2^{m \times m}$$

where $a_1^{(j)}, a_2^{(j)}, \ldots$ are the coefficients of the Laurent series

$$\frac{q_j(x)}{p(x)} = \frac{a_1^{(j)}}{x} + \frac{a_2^{(j)}}{x^2} + \cdots .$$

# Polynomial lattice point sets (Niederreiter, 1992)

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$
- Let $\boldsymbol{q} = (q_1, \ldots, q_s) \in (\mathbb{F}_2[x])^s$ with $\deg(q_j) < m$. For each $j$, $C_j$ is given by the square Hankel matrix

$$C_j = \begin{pmatrix} a_1^{(j)} & a_2^{(j)} & \cdots & a_m^{(j)} \\ a_2^{(j)} & a_3^{(j)} & \cdots & a_{m+1}^{(j)} \\ \vdots & \vdots & \ddots & \vdots \\ a_m^{(j)} & a_{m+1}^{(j)} & \cdots & a_{2m-1}^{(j)} \end{pmatrix} \in \mathbb{F}_2^{m \times m}$$

where $a_1^{(j)}, a_2^{(j)}, \ldots$ are the coefficients of the Laurent series

$$\frac{q_j(x)}{p(x)} = \frac{a_1^{(j)}}{x} + \frac{a_2^{(j)}}{x^2} + \cdots .$$

- The resulting digital net is called a *polynomial lattice point set* $P(p, \boldsymbol{q})$. Usually the vector $\boldsymbol{q}$ is constructed by a (fast) computer search algorithm (Nuyens & Cools, 2006;

# Polynomial lattice point sets (Niederreiter, 1992)

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$
- Let $\boldsymbol{q} = (q_1, \ldots, q_s) \in (\mathbb{F}_2[x])^s$ with $\deg(q_j) < m$. For each $j$, $C_j$ is given by the square Hankel matrix

$$C_j = \begin{pmatrix} a_1^{(j)} & a_2^{(j)} & \cdots & a_m^{(j)} \\ a_2^{(j)} & a_3^{(j)} & \cdots & a_{m+1}^{(j)} \\ \vdots & \vdots & \ddots & \vdots \\ a_m^{(j)} & a_{m+1}^{(j)} & \cdots & a_{2m-1}^{(j)} \end{pmatrix} \in \mathbb{F}_2^{m \times m}$$

where $a_1^{(j)}, a_2^{(j)}, \ldots$ are the coefficients of the Laurent series

$$\frac{q_j(x)}{p(x)} = \frac{a_1^{(j)}}{x} + \frac{a_2^{(j)}}{x^2} + \cdots .$$

- The resulting digital net is called a *polynomial lattice point set* $P(p, \boldsymbol{q})$. Usually the vector $\boldsymbol{q}$ is constructed by a (fast) computer search algorithm (Nuyens & Cools, 2006; P. Kritzer, this afternoon!).

# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# What if $f$ is smooth

- In some applications, such as PDEs with random coefficients, $f$ can be smooth. A proper design of QMC point sets enables higher order convergence of the integration error than $O(1/N)$ as expected from the KH inequality.
- So far, QMC point sets achieving higher order convergence for non-periodic smooth functions are

1. Higher order digital nets/sequences (Dick, 2008; ...), and
2. Tent-transformed lattice point sets (Hickernell, 2002; ...).

# What if $f$ is smooth

- In some applications, such as PDEs with random coefficients, $f$ can be smooth. A proper design of QMC point sets enables higher order convergence of the integration error than $O(1/N)$ as expected from the KH inequality.
- So far, QMC point sets achieving higher order convergence for non-periodic smooth functions are

1. Higher order digital nets/sequences (Dick, 2008; ...), and
2. Tent-transformed lattice point sets (Hickernell, 2002; ...).

- In this talk, I will focus on **higher order digital nets/sequences**. The contents of this section are mostly developed by Dick (2008). Please refer to a recent review by G. & Suzuki (arXiv:1903.12353).

# Quality measure: high order $t$-value

- Let $\alpha \in \mathbb{N}$.
- Let $t$ be an integer such that, for any choice $1 \le d_{j,v_j} < \cdots < d_{j,1} \le \alpha m$, $0 \le v_j \le \alpha m$, $1 \le j \le s$, with

$$\sum_{j=1}^{s} \sum_{i=1}^{\min(v_j,\alpha)} d_{j,i} = \alpha m - t,$$

the $d_{1,v_1}, \ldots, d_{1,1}$-th rows of $C_1$

$\vdots$

the $d_{s,v_s}, \ldots, d_{s,1}$-th rows of $C_s$

are linearly independent over $\mathbb{F}_2$. $P$ is called an *order* $\alpha$ *digital* $(t, m, s)$*-net*.

# Quality measure: high order $t$-value

- Order $\alpha$ digital $(t, m, s)$-nets hold equi-distribution properties: union of dyadic elementary boxes contains the fair number of points (shown later visually).

# Quality measure: high order $t$-value

- Order $\alpha$ digital $(t, m, s)$-nets hold equi-distribution properties: union of dyadic elementary boxes contains the fair number of points (shown later visually).

- Let $t$ be an integer such that, for any $\alpha m \geq t$, the first $2^m$ points of $\mathcal{S}$ are an order $\alpha$ digital $(t, m, s)$-net. $\mathcal{S}$ is called an *order $\alpha$ digital $(t, s)$-sequence*.

# Explicit construction

- Define $\mathcal{D}_\alpha : [0,1)^\alpha \to [0,1)$ by

$$\begin{cases} x_1 = (0.\xi_1^{(1)}\xi_2^{(1)}\xi_3^{(1)}\ldots)_2 \\ x_2 = (0.\xi_1^{(2)}\xi_2^{(2)}\xi_3^{(2)}\ldots)_2 \\ \quad\vdots \\ x_\alpha = (0.\xi_1^{(\alpha)}\xi_2^{(\alpha)}\xi_3^{(\alpha)}\ldots)_2 \end{cases} \mapsto (0.\underbrace{\xi_1^{(1)}\xi_1^{(2)}\ldots\xi_1^{(\alpha)}}_{\alpha}\underbrace{\xi_2^{(1)}\xi_2^{(2)}\ldots\xi_2^{(\alpha)}}_{\alpha}\ldots)_2.$$

# Explicit construction

- Define $\mathcal{D}_\alpha : [0,1)^\alpha \to [0,1)$ by

$$\begin{cases} x_1 = (0.\xi_1^{(1)}\xi_2^{(1)}\xi_3^{(1)}\ldots)_2 \\ x_2 = (0.\xi_1^{(2)}\xi_2^{(2)}\xi_3^{(2)}\ldots)_2 \\ \quad\vdots \\ x_\alpha = (0.\xi_1^{(\alpha)}\xi_2^{(\alpha)}\xi_3^{(\alpha)}\ldots)_2 \end{cases} \mapsto (0.\underbrace{\xi_1^{(1)}\xi_1^{(2)}\ldots\xi_1^{(\alpha)}}_{\alpha}\underbrace{\xi_2^{(1)}\xi_2^{(2)}\ldots\xi_2^{(\alpha)}}_{\alpha}\ldots)_2.$$

- For $\boldsymbol{x} \in [0,1)^{\alpha s}$, let

$$\mathcal{D}_\alpha(\boldsymbol{x}) = (\mathcal{D}_\alpha(x_1,\ldots,x_\alpha), \mathcal{D}_\alpha(x_{\alpha+1},\ldots,x_{2\alpha}), \ldots, \mathcal{D}_\alpha(x_{\alpha(s-1)+1},\ldots,x_{\alpha s})) \in [0,1)^s.$$

# Explicit construction

- Define $\mathcal{D}_\alpha : [0,1)^\alpha \to [0,1)$ by

$$\begin{cases} x_1 = (0.\xi_1^{(1)}\xi_2^{(1)}\xi_3^{(1)}\ldots)_2 \\ x_2 = (0.\xi_1^{(2)}\xi_2^{(2)}\xi_3^{(2)}\ldots)_2 \\ \quad\vdots \\ x_\alpha = (0.\xi_1^{(\alpha)}\xi_2^{(\alpha)}\xi_3^{(\alpha)}\ldots)_2 \end{cases} \mapsto (0.\underbrace{\xi_1^{(1)}\xi_1^{(2)}\ldots\xi_1^{(\alpha)}}_{\alpha}\underbrace{\xi_2^{(1)}\xi_2^{(2)}\ldots\xi_2^{(\alpha)}}_{\alpha}\ldots)_2.$$

- For $\boldsymbol{x} \in [0,1)^{\alpha s}$, let

$$\mathcal{D}_\alpha(\boldsymbol{x}) = (\mathcal{D}_\alpha(x_1,\ldots,x_\alpha), \mathcal{D}_\alpha(x_{\alpha+1},\ldots,x_{2\alpha}), \ldots, \mathcal{D}_\alpha(x_{\alpha(s-1)+1},\ldots,x_{\alpha s})) \in [0,1)^s.$$

For a digital $(t,m,\alpha s)$-net $P$, we write

$$\mathcal{D}_\alpha(P) = \{\mathcal{D}_\alpha(\boldsymbol{x}) \mid \boldsymbol{x} \in P\}.$$

# Another look at construction

- Let $P$ be a digital $(t, m, \alpha s)$-net with $C_1, \ldots, C_{\alpha s} \in \mathbb{F}_2^{m \times m}$. We write

$$C_1 = \begin{pmatrix} \boldsymbol{c}_1^{(1)} \\ \vdots \\ \boldsymbol{c}_m^{(1)} \end{pmatrix}, C_2 = \begin{pmatrix} \boldsymbol{c}_1^{(2)} \\ \vdots \\ \boldsymbol{c}_m^{(2)} \end{pmatrix}, \ldots, C_\alpha = \begin{pmatrix} \boldsymbol{c}_1^{(\alpha)} \\ \vdots \\ \boldsymbol{c}_m^{(\alpha)} \end{pmatrix}, \ldots.$$

# Another look at construction

- Let $P$ be a digital $(t, m, \alpha s)$-net with $C_1, \ldots, C_{\alpha s} \in \mathbb{F}_2^{m \times m}$. We write

$$C_1 = \begin{pmatrix} \boldsymbol{c}_1^{(1)} \\ \vdots \\ \boldsymbol{c}_m^{(1)} \end{pmatrix}, C_2 = \begin{pmatrix} \boldsymbol{c}_1^{(2)} \\ \vdots \\ \boldsymbol{c}_m^{(2)} \end{pmatrix}, \ldots, C_\alpha = \begin{pmatrix} \boldsymbol{c}_1^{(\alpha)} \\ \vdots \\ \boldsymbol{c}_m^{(\alpha)} \end{pmatrix}, \ldots.$$

- $\mathcal{D}_\alpha(P)$ is a digital net with $D_1, \ldots, D_s \in \mathbb{F}_2^{\alpha m \times m}$ where

$$D_1 = \begin{pmatrix} \boldsymbol{c}_1^{(1)} \\ \boldsymbol{c}_1^{(2)} \\ \vdots \\ \boldsymbol{c}_1^{(\alpha)} \\ \vdots \\ \boldsymbol{c}_m^{(1)} \\ \boldsymbol{c}_m^{(2)} \\ \vdots \\ \boldsymbol{c}_m^{(\alpha)} \end{pmatrix}, D_2 = \begin{pmatrix} \boldsymbol{c}_1^{(\alpha+1)} \\ \boldsymbol{c}_1^{(\alpha+2)} \\ \vdots \\ \boldsymbol{c}_1^{(2\alpha)} \\ \vdots \\ \boldsymbol{c}_m^{(\alpha+1)} \\ \boldsymbol{c}_m^{(\alpha+2)} \\ \vdots \\ \boldsymbol{c}_m^{(2\alpha)} \end{pmatrix}, \ldots.$$

# High order $t$-value

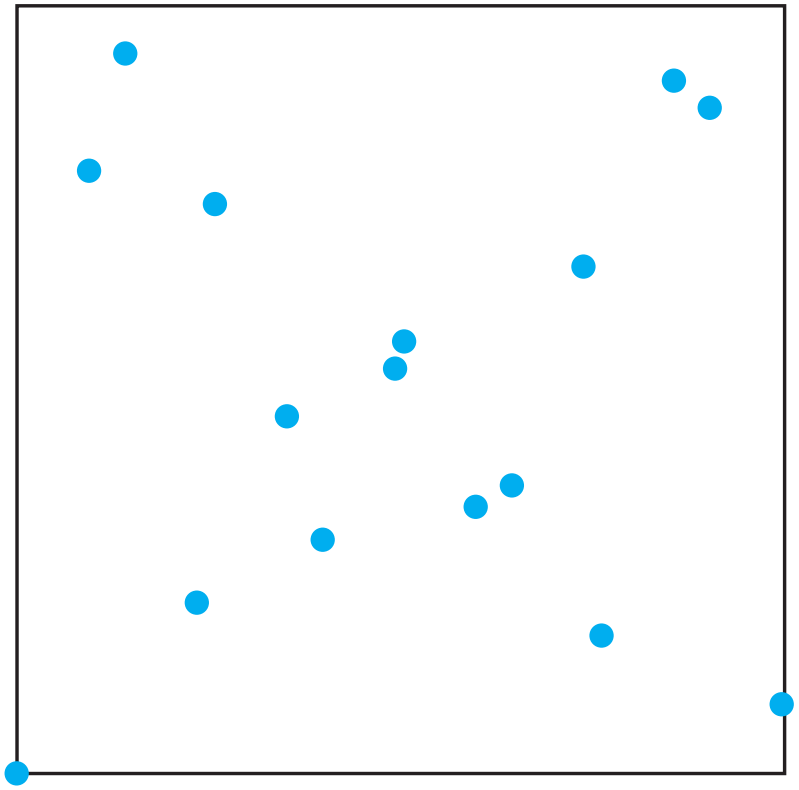
- For a digital $(t, m, \alpha s)$-net $P$, $\mathcal{D}_\alpha(P)$ is an order $\alpha$ digital $(t', m, s)$-net with

$$t' \leq \alpha \min \left\{ m, t + \left\lfloor \frac{s(\alpha - 1)}{2} \right\rfloor \right\}.$$

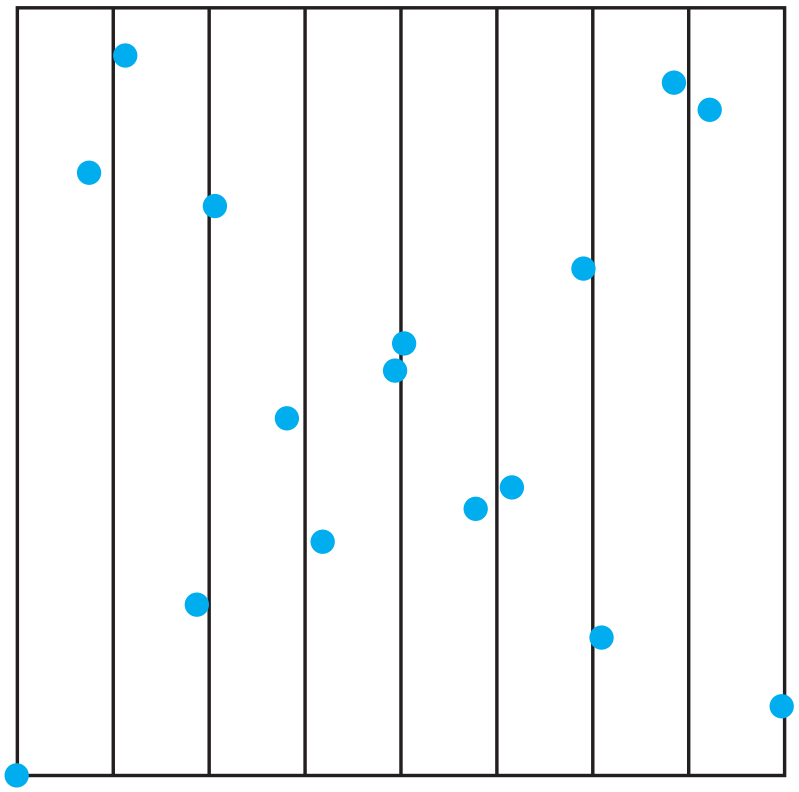
- For a digital $(t, \alpha s)$-sequences $\mathcal{S}$, $\mathcal{D}_\alpha(\mathcal{S})$ is an order $\alpha$ digital $(t', s)$-sequences with

$$t' \leq \alpha t + \frac{s\alpha(\alpha - 1)}{2}.$$

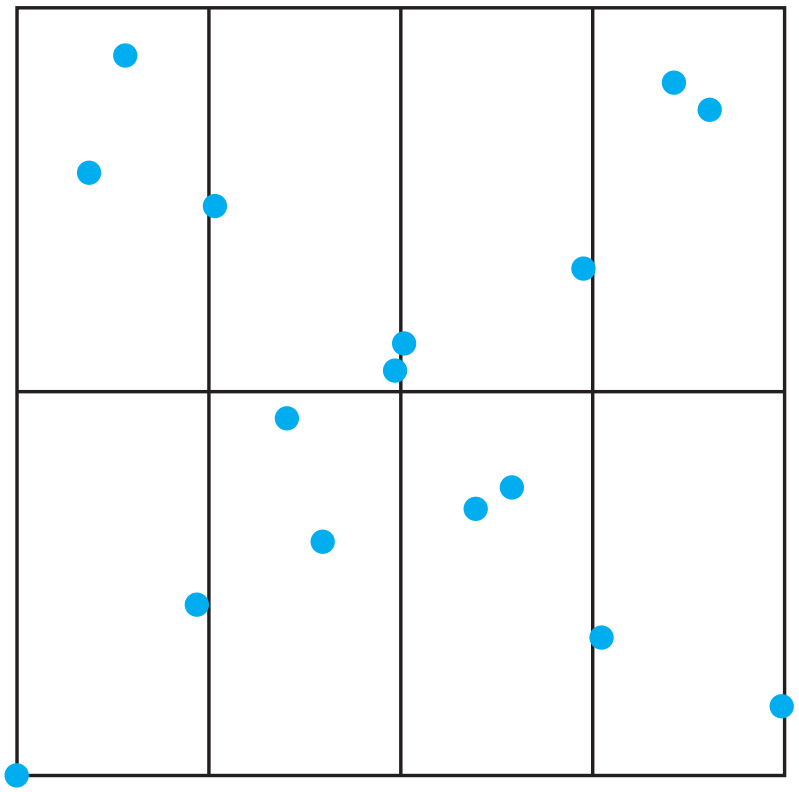
# Interlaced Sobol' point sets with $\alpha = 2$



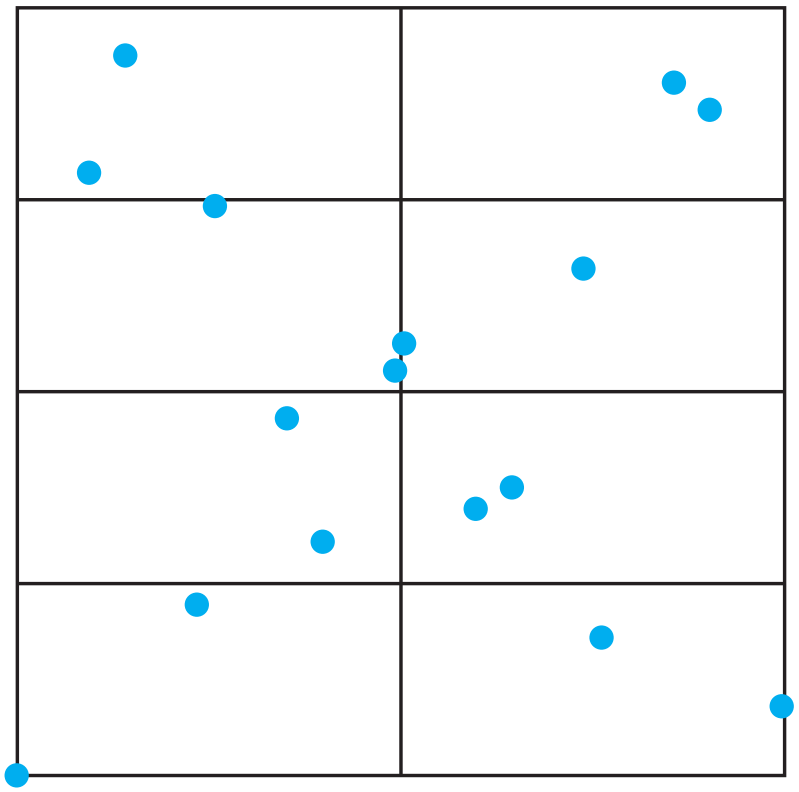
# Interlaced Sobol' point sets with $\alpha = 2$



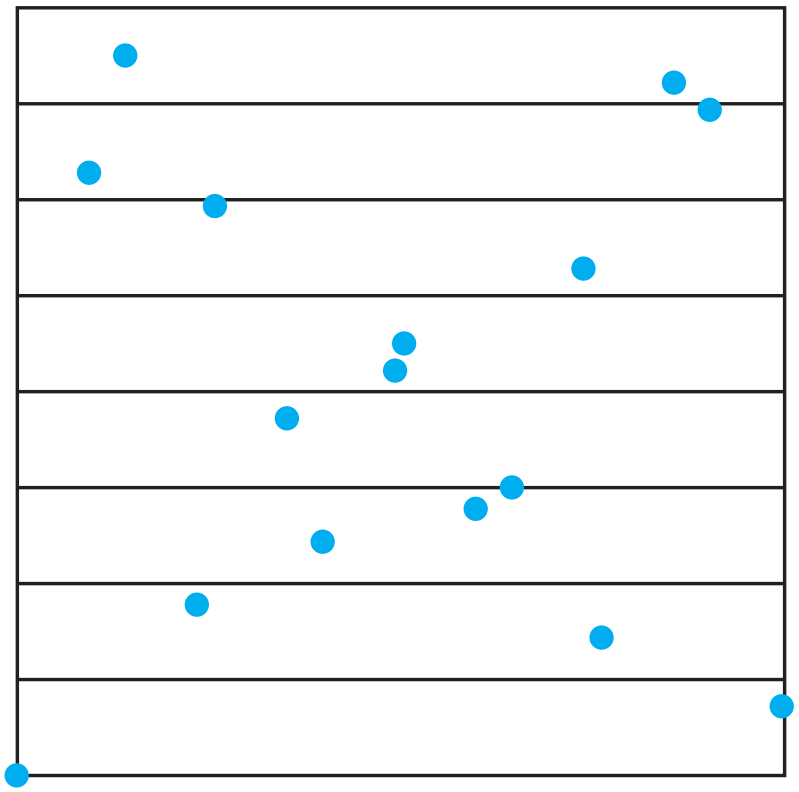
# Interlaced Sobol' point sets with $\alpha = 2$



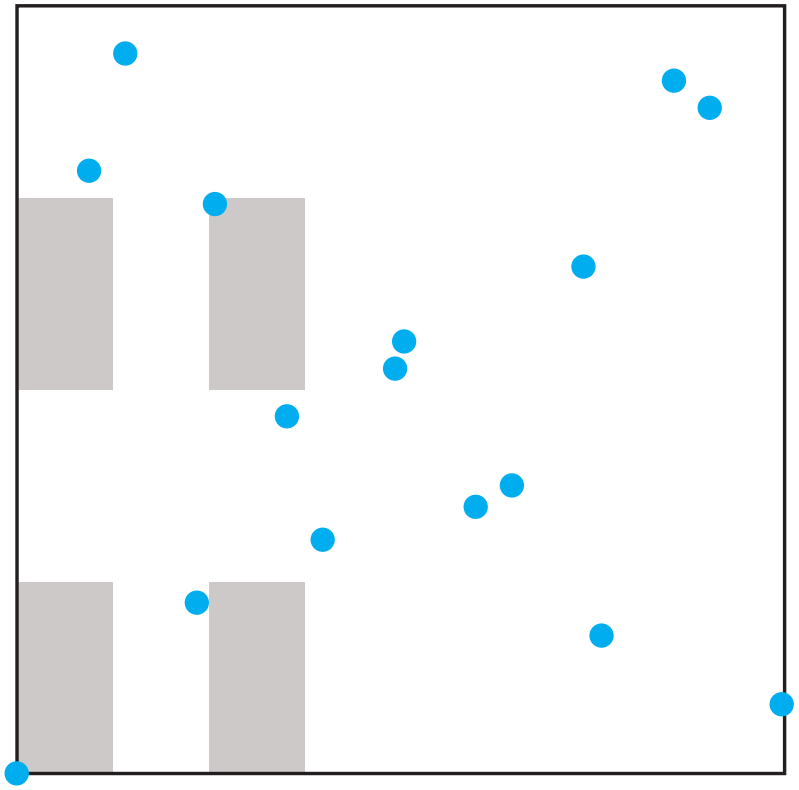
# Interlaced Sobol' point sets with $\alpha = 2$



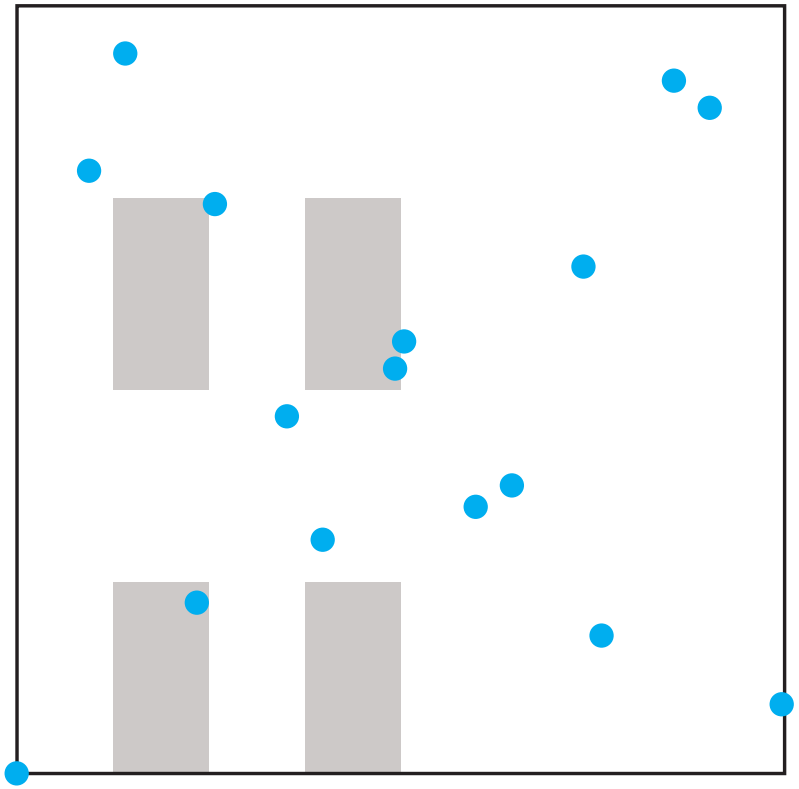
# Interlaced Sobol' point sets with $\alpha = 2$



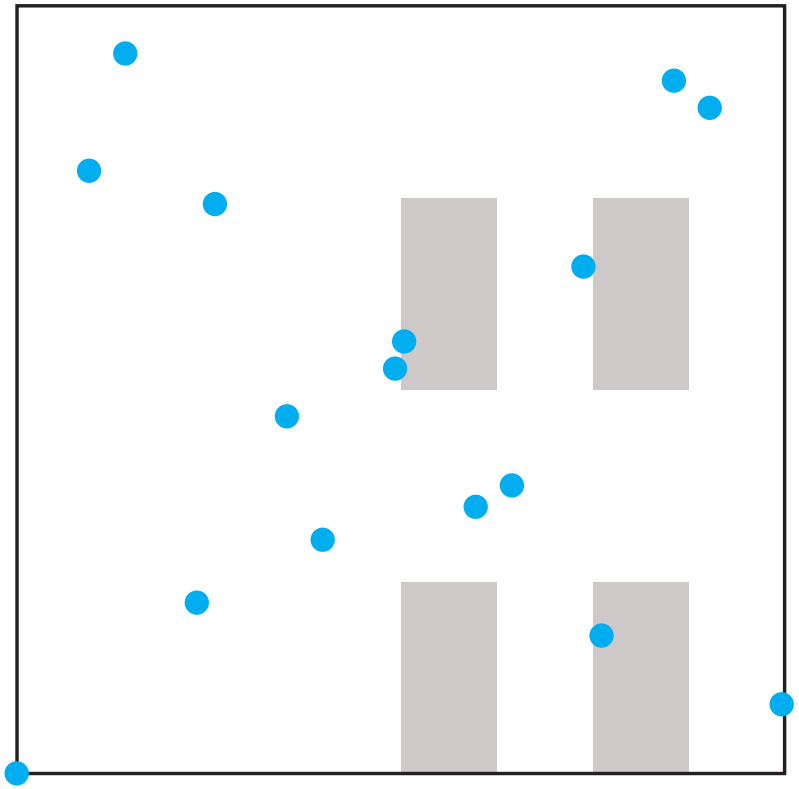
# Interlaced Sobol' point sets with $\alpha = 2$



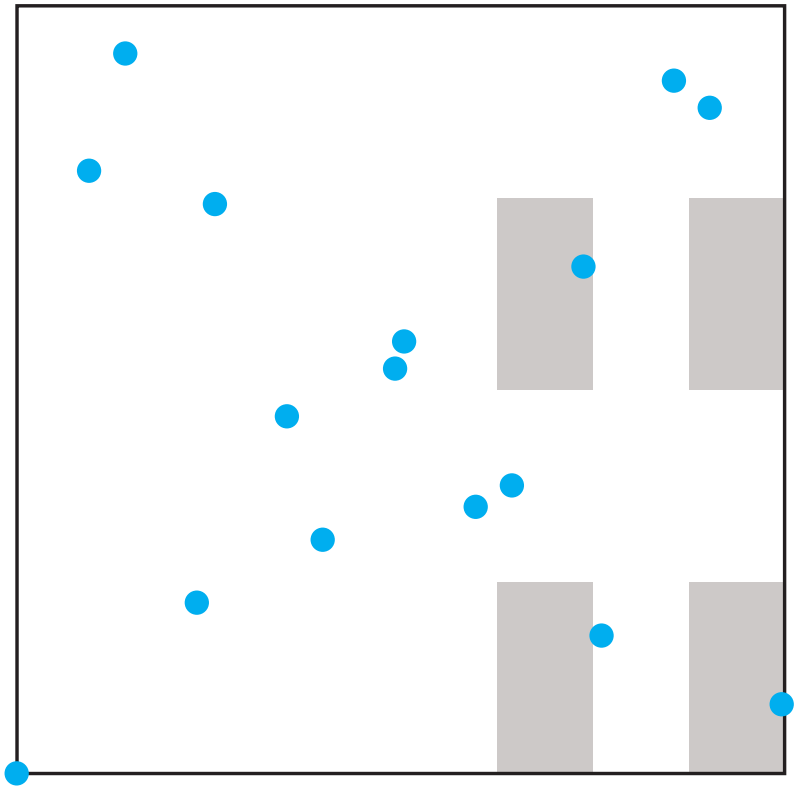
# Interlaced Sobol' point sets with $\alpha = 2$



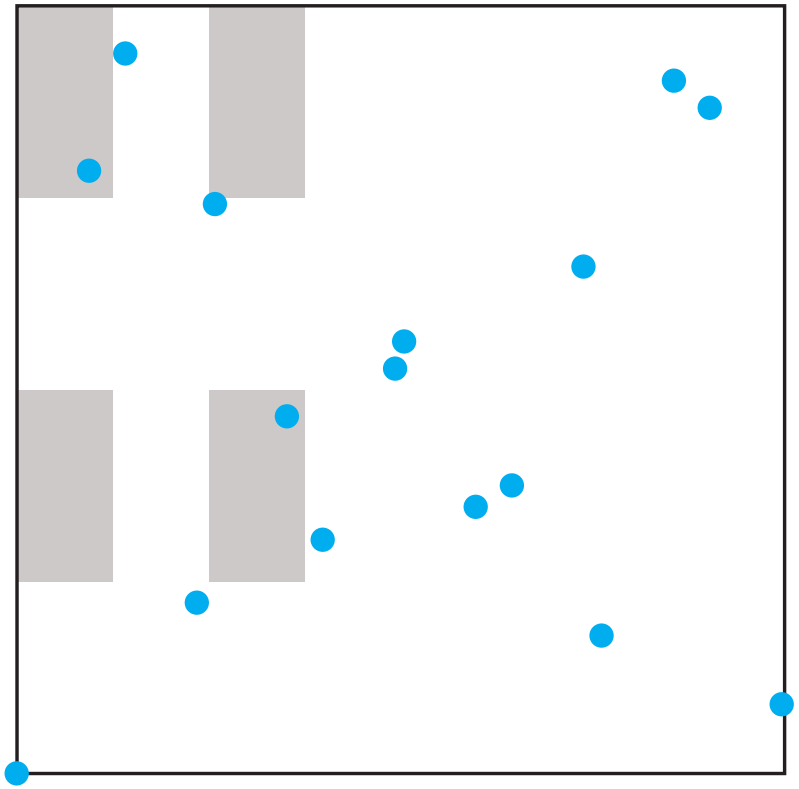
# Interlaced Sobol' point sets with $\alpha = 2$



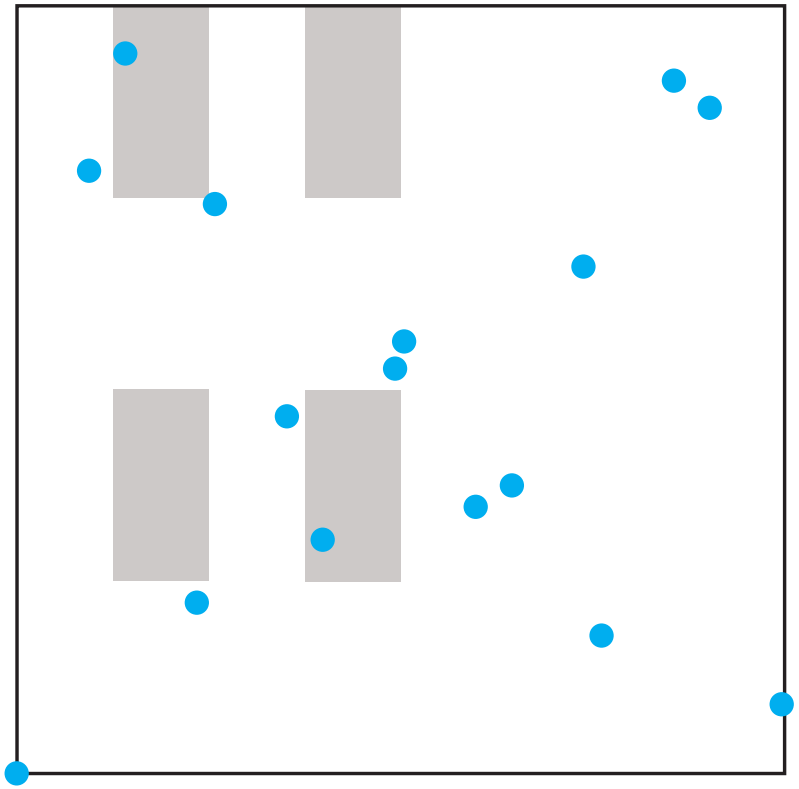
# Interlaced Sobol' point sets with $\alpha = 2$



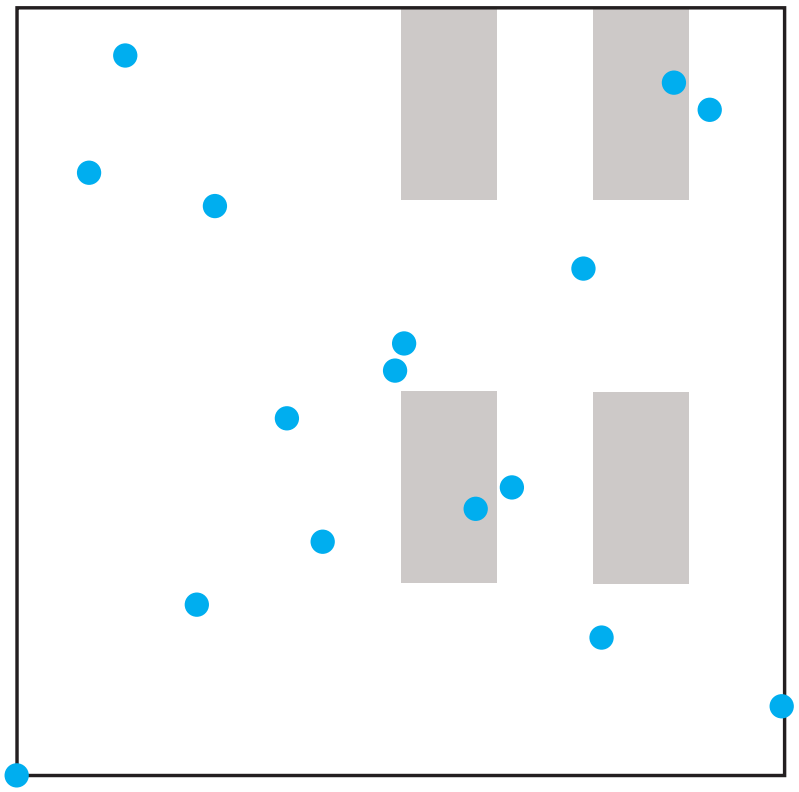
# Interlaced Sobol' point sets with $\alpha = 2$



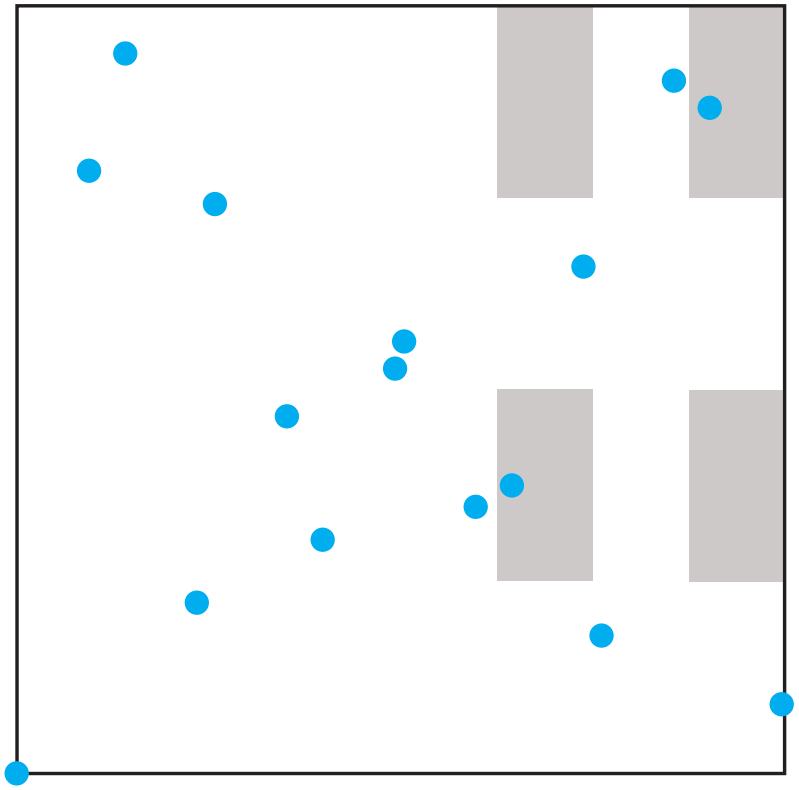
# Interlaced Sobol' point sets with $\alpha = 2$



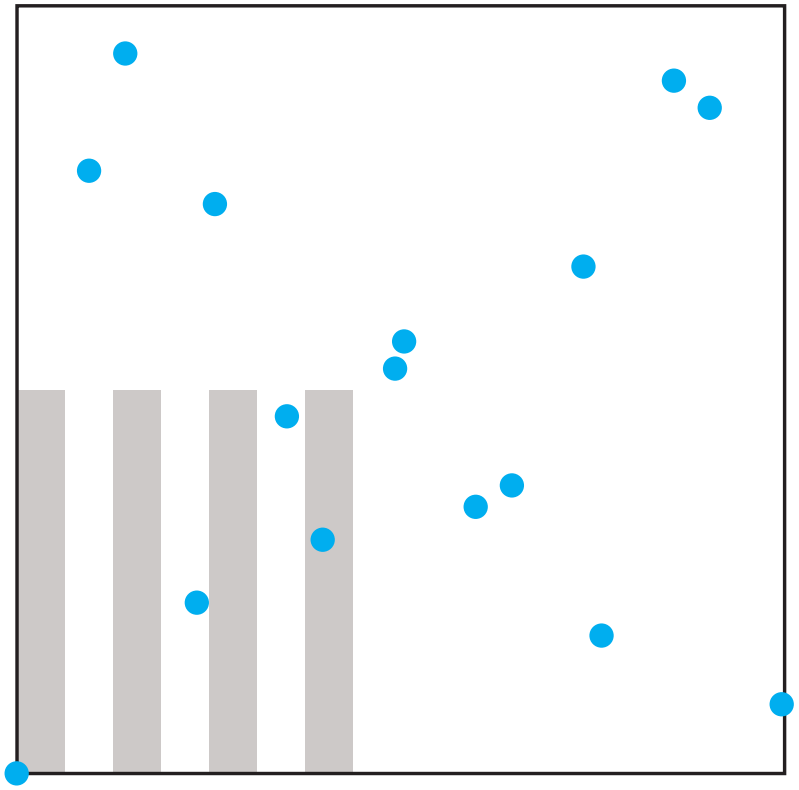
# Interlaced Sobol' point sets with $\alpha = 2$



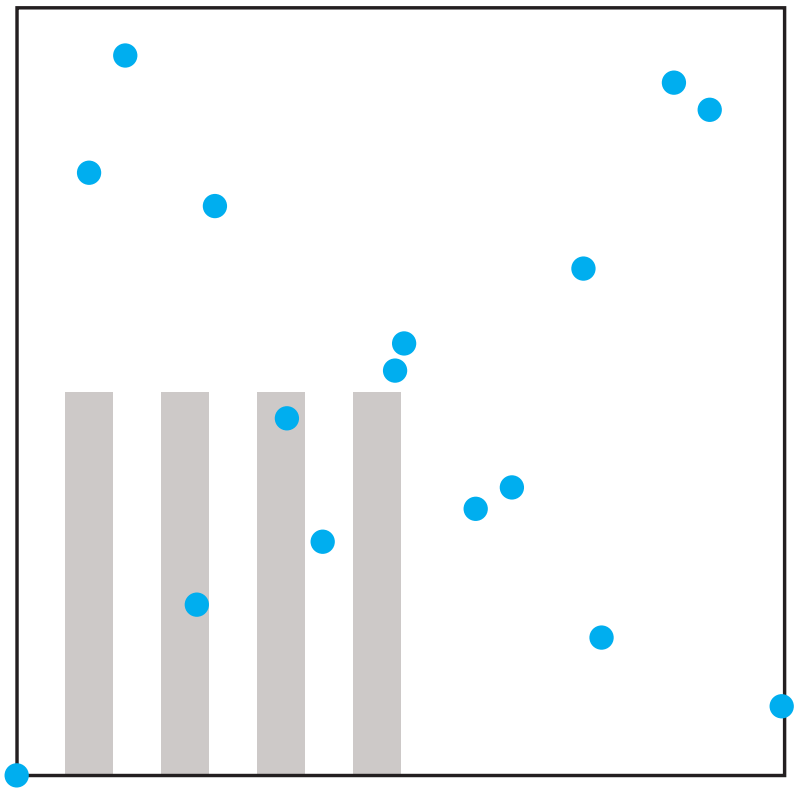
# Interlaced Sobol' point sets with $\alpha = 2$



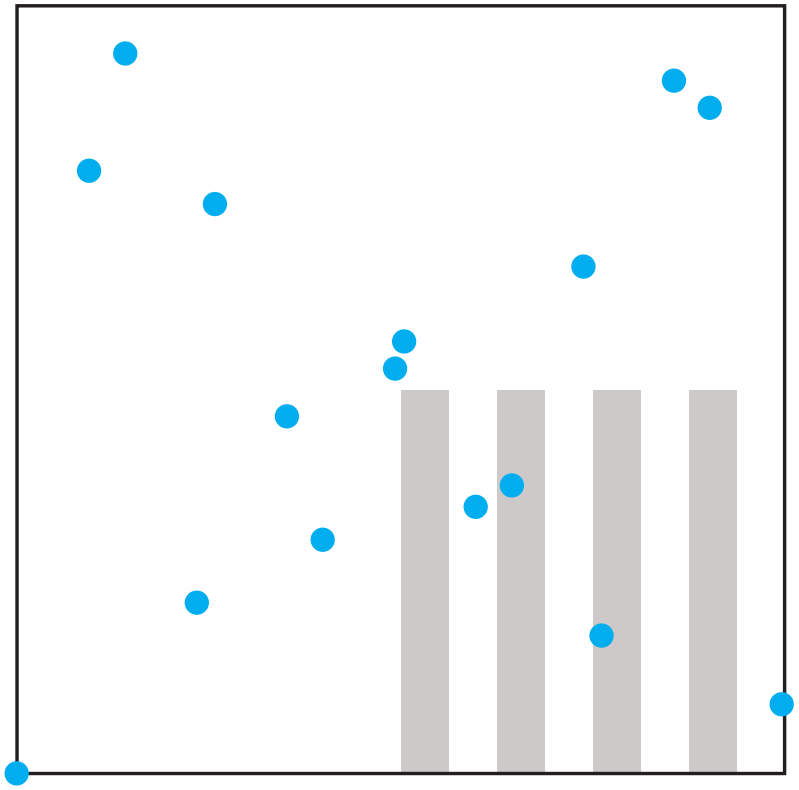
# Interlaced Sobol' point sets with $\alpha = 2$



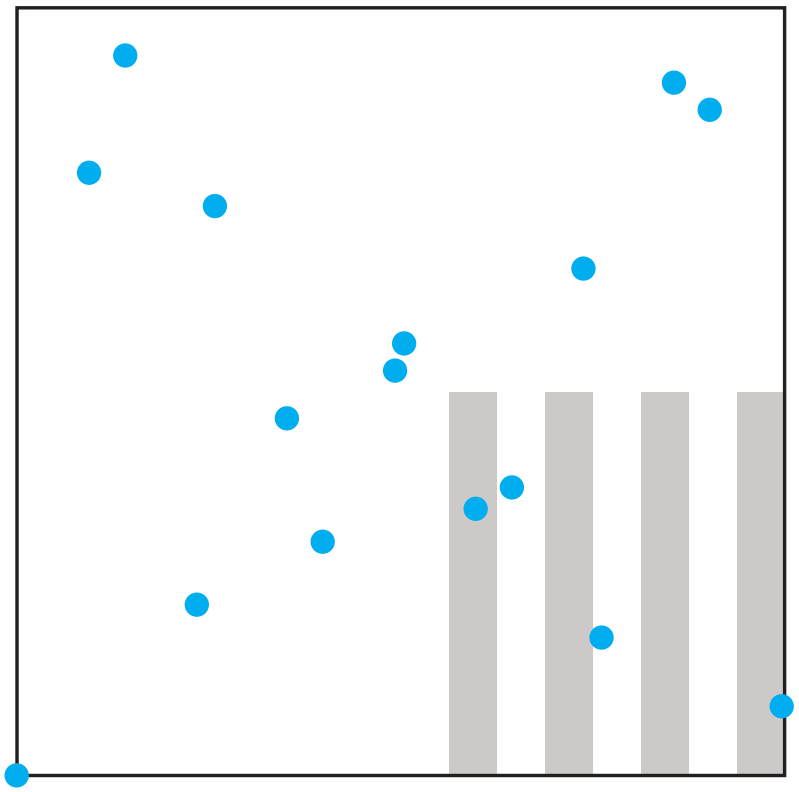
# Interlaced Sobol' point sets with $\alpha = 2$



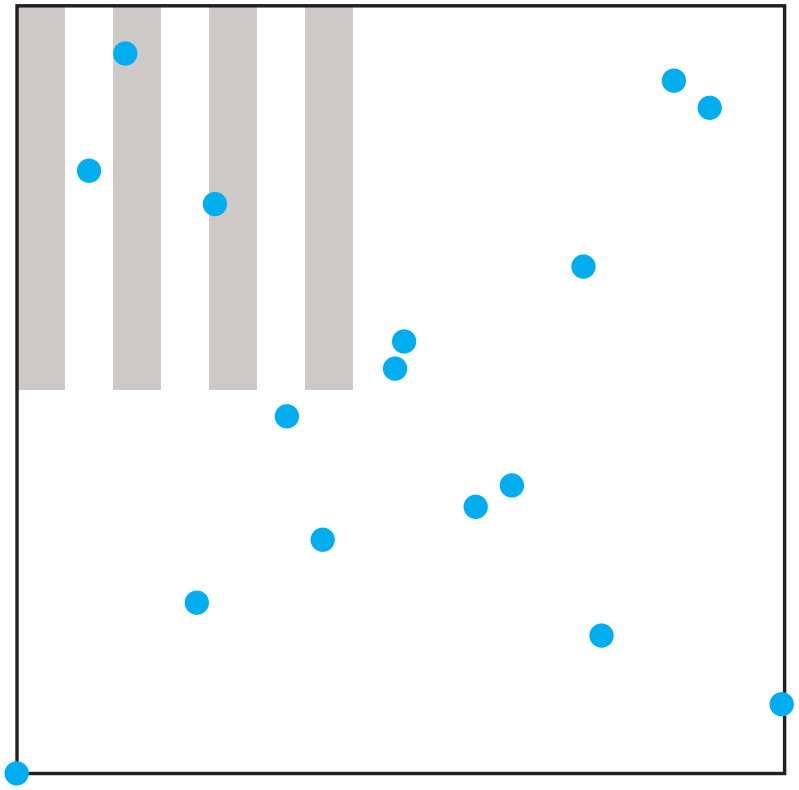
# Interlaced Sobol' point sets with $\alpha = 2$



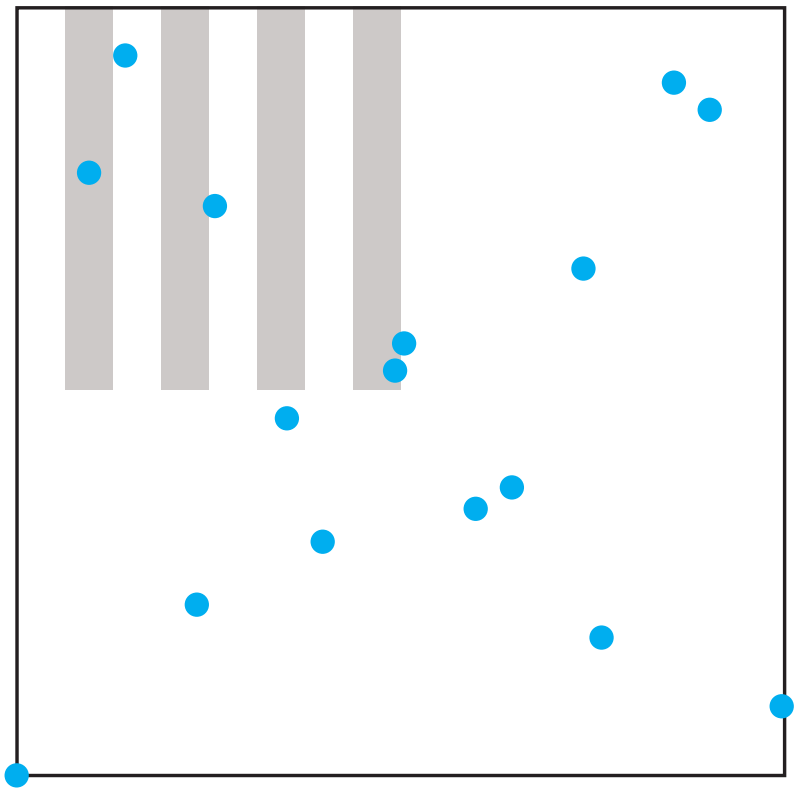
# Interlaced Sobol' point sets with $\alpha = 2$



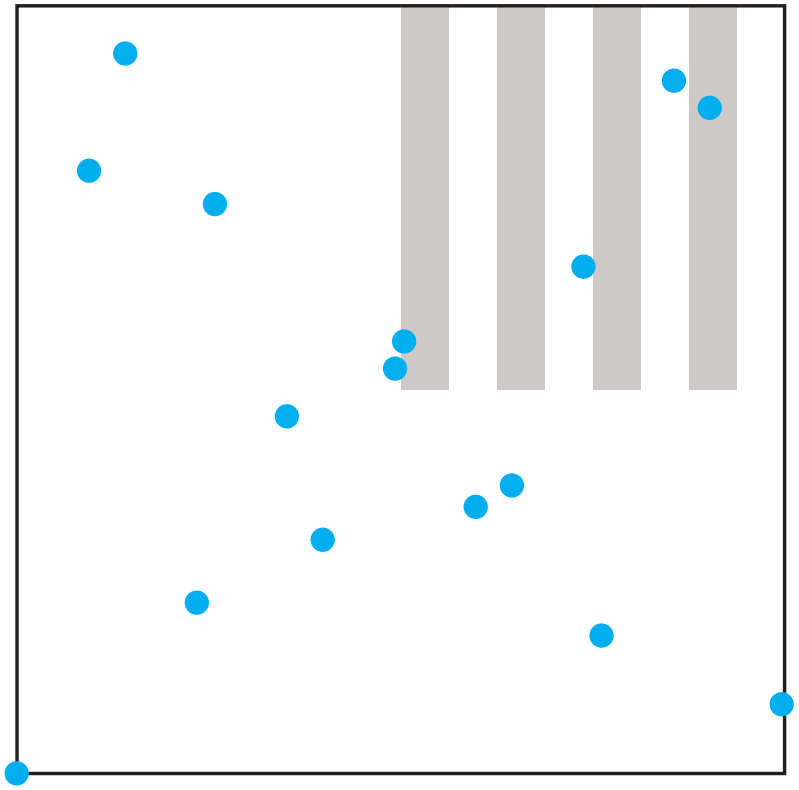
# Interlaced Sobol' point sets with $\alpha = 2$



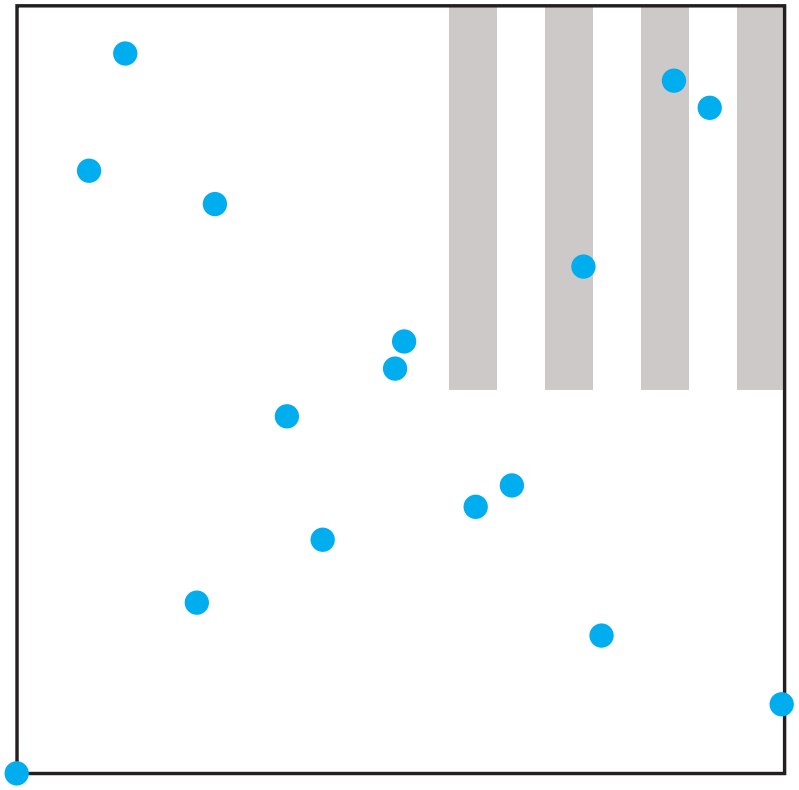
# Interlaced Sobol' point sets with $\alpha = 2$



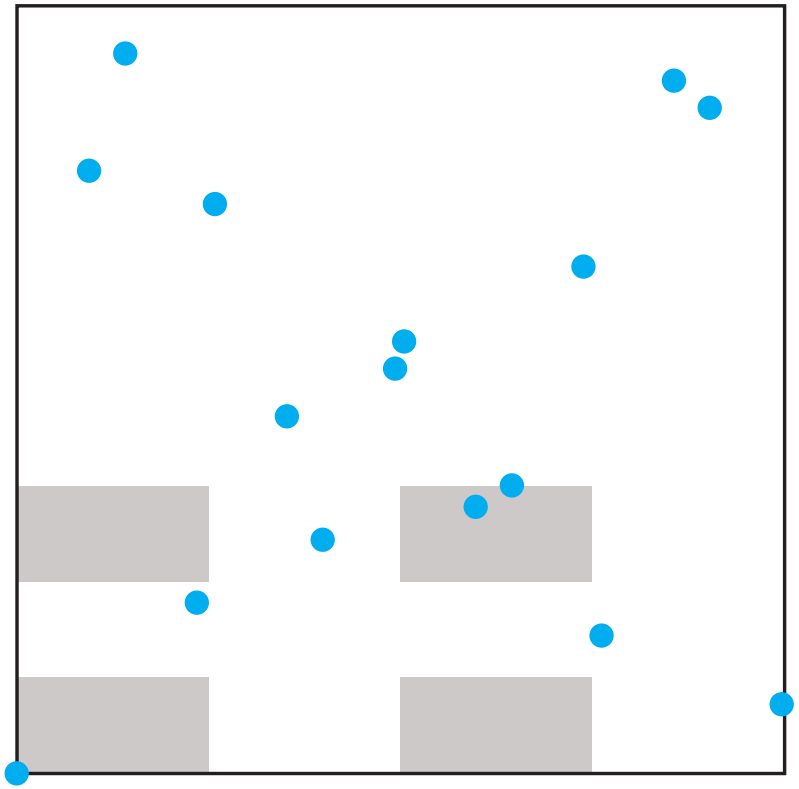
# Interlaced Sobol' point sets with $\alpha = 2$



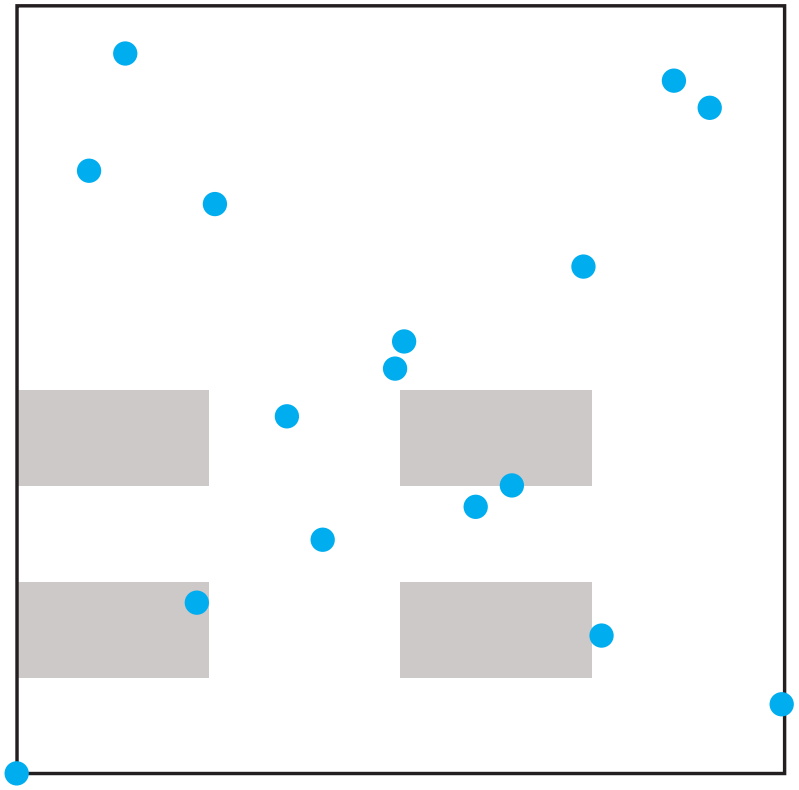
# Interlaced Sobol' point sets with $\alpha = 2$



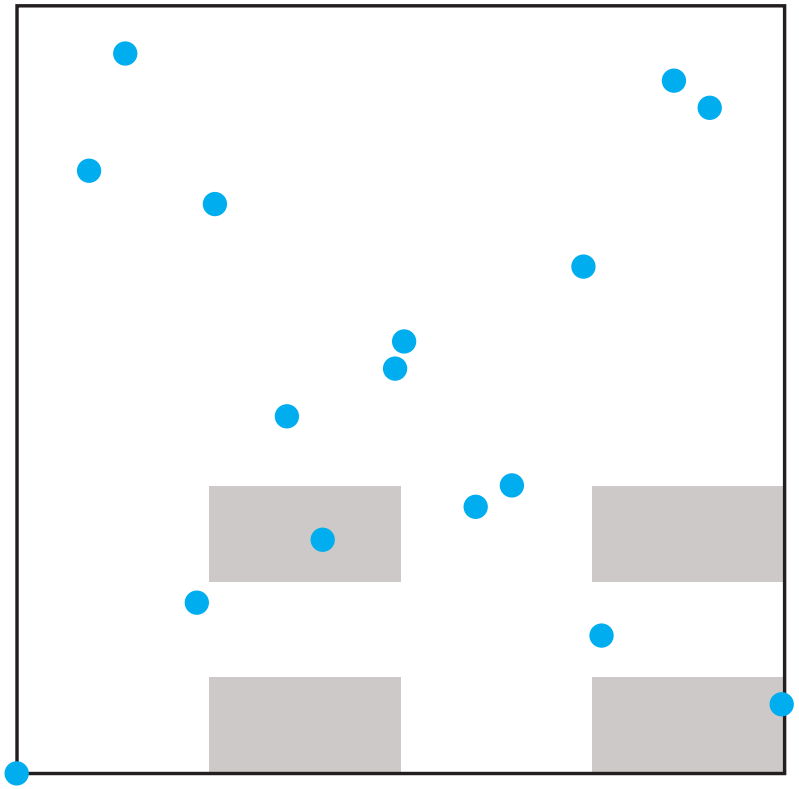
# Interlaced Sobol' point sets with $\alpha = 2$



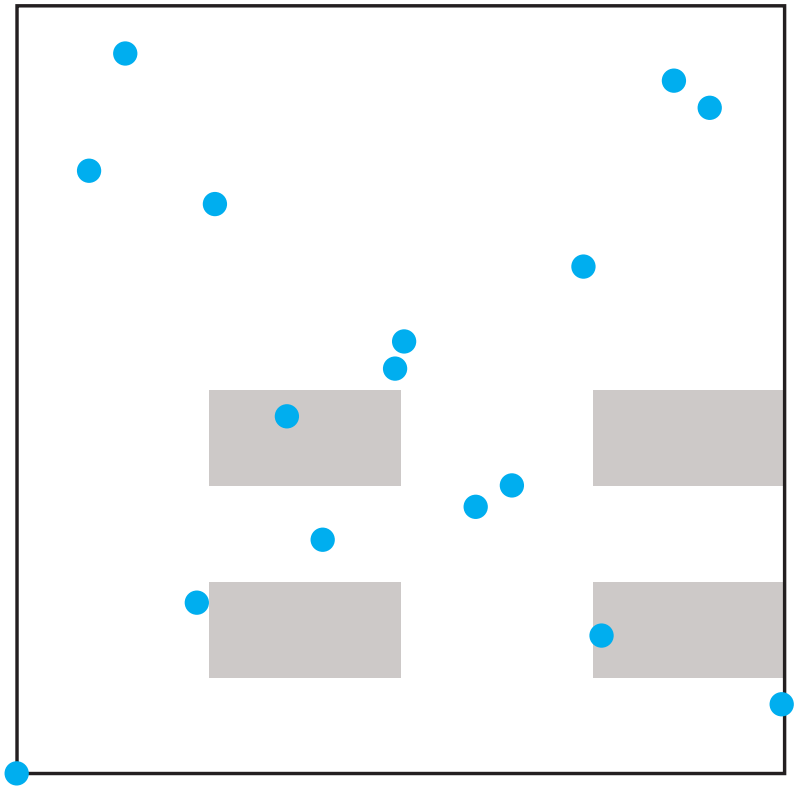
# Interlaced Sobol' point sets with $\alpha = 2$



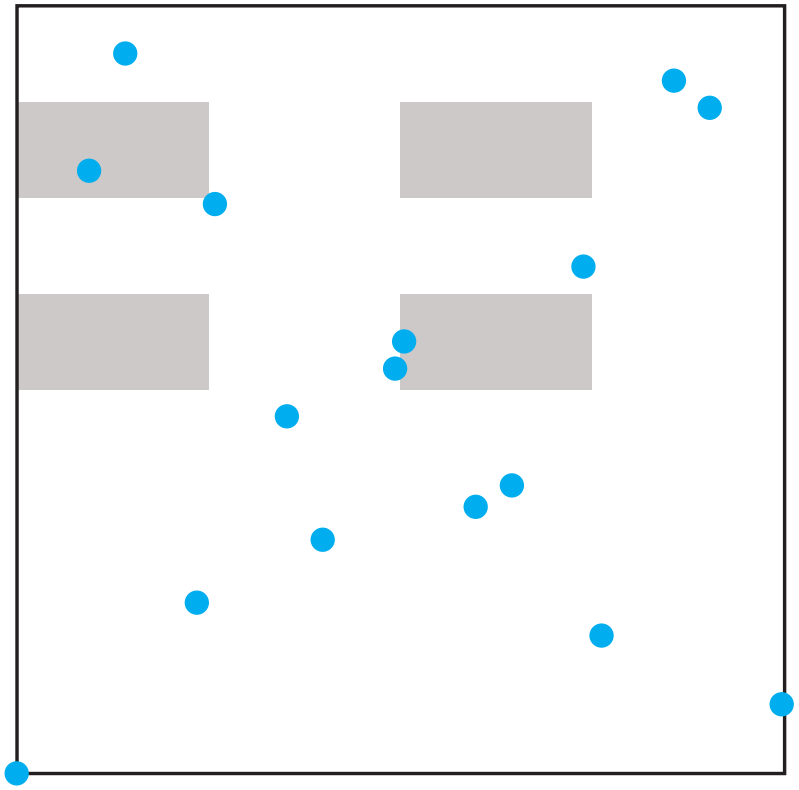
# Interlaced Sobol' point sets with $\alpha = 2$



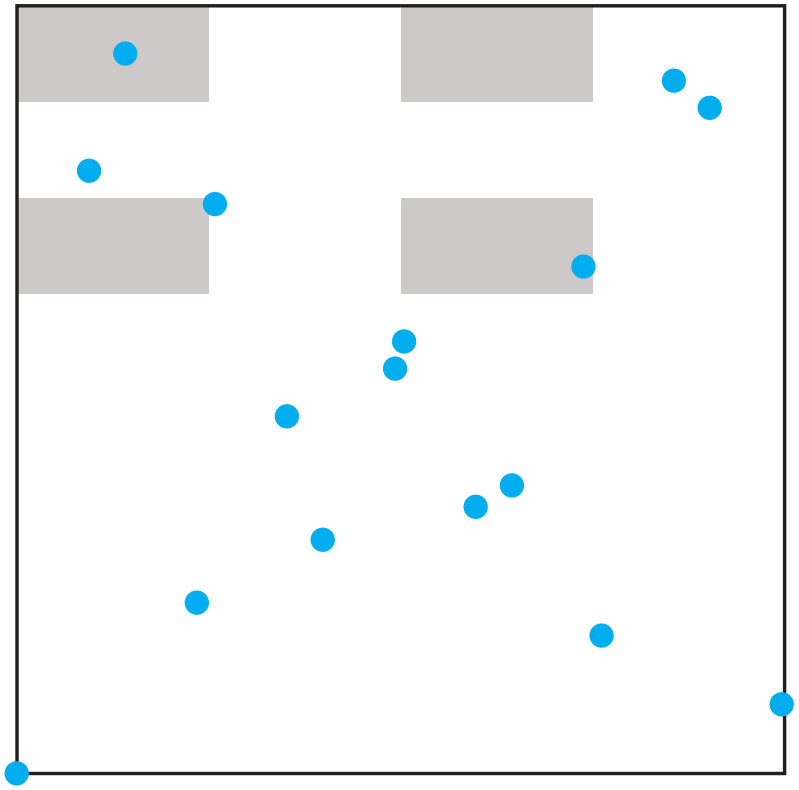
# Interlaced Sobol' point sets with $\alpha = 2$



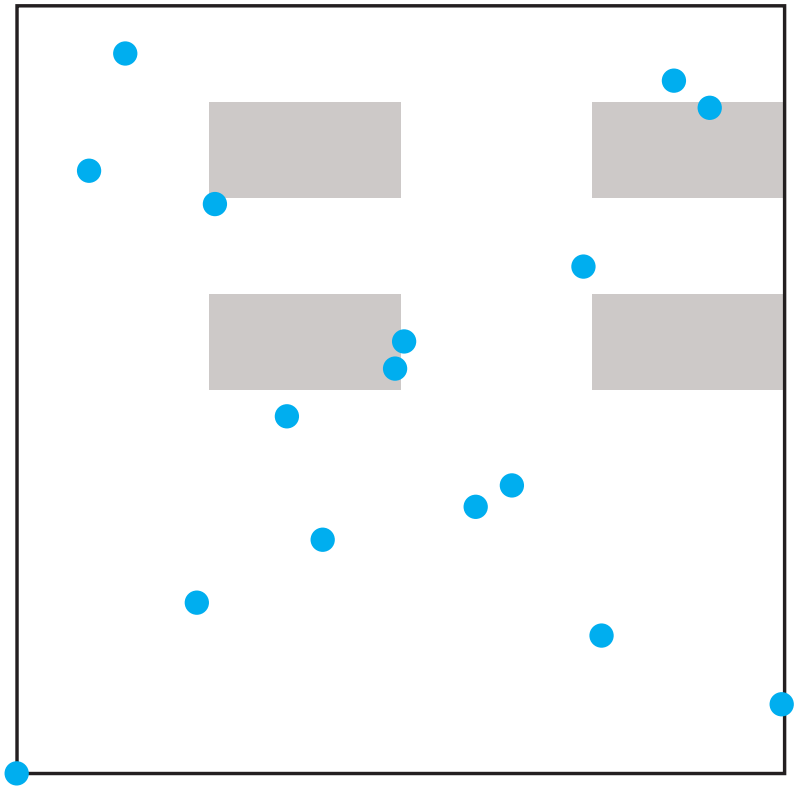
# Interlaced Sobol' point sets with $\alpha = 2$



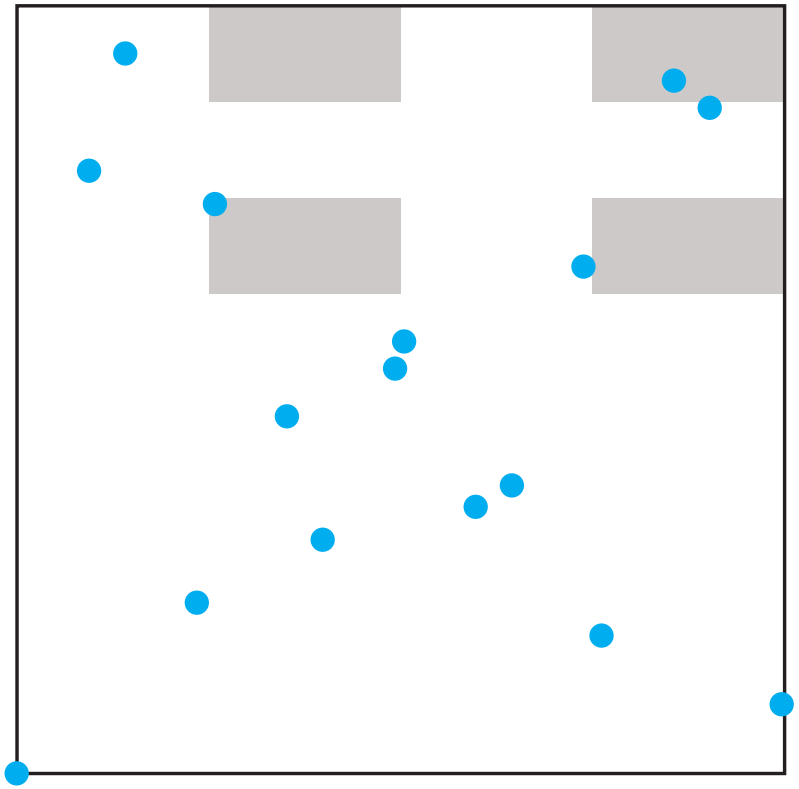
# Interlaced Sobol' point sets with $\alpha = 2$



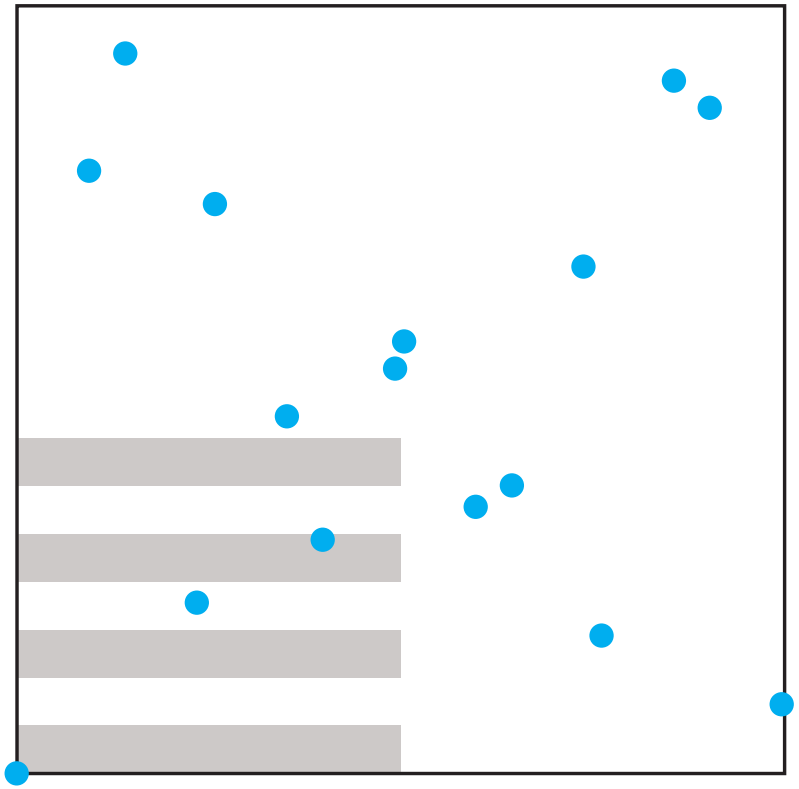
# Interlaced Sobol' point sets with $\alpha = 2$



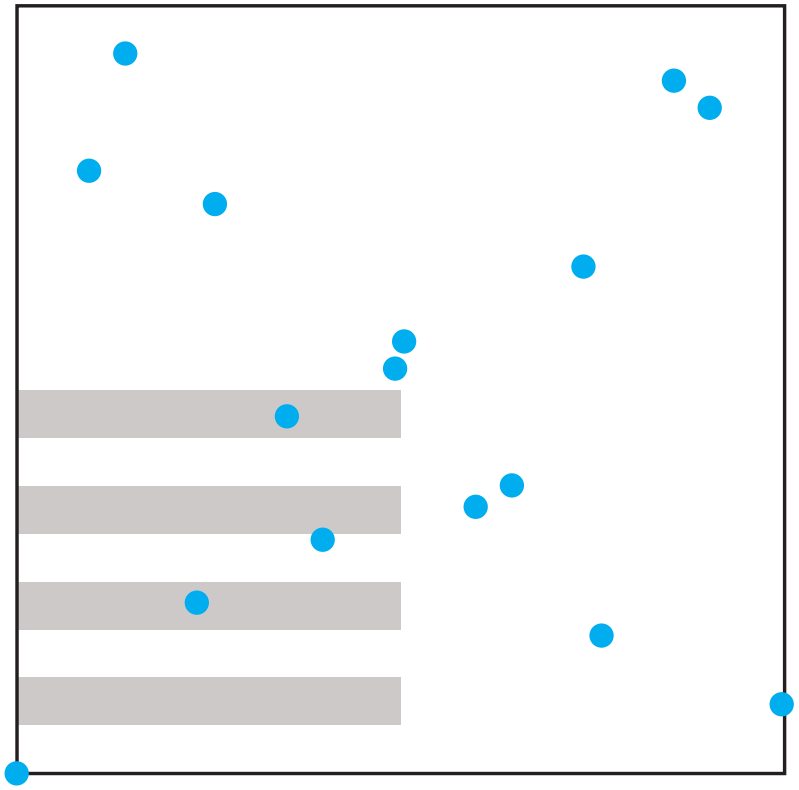
# Interlaced Sobol' point sets with $\alpha = 2$



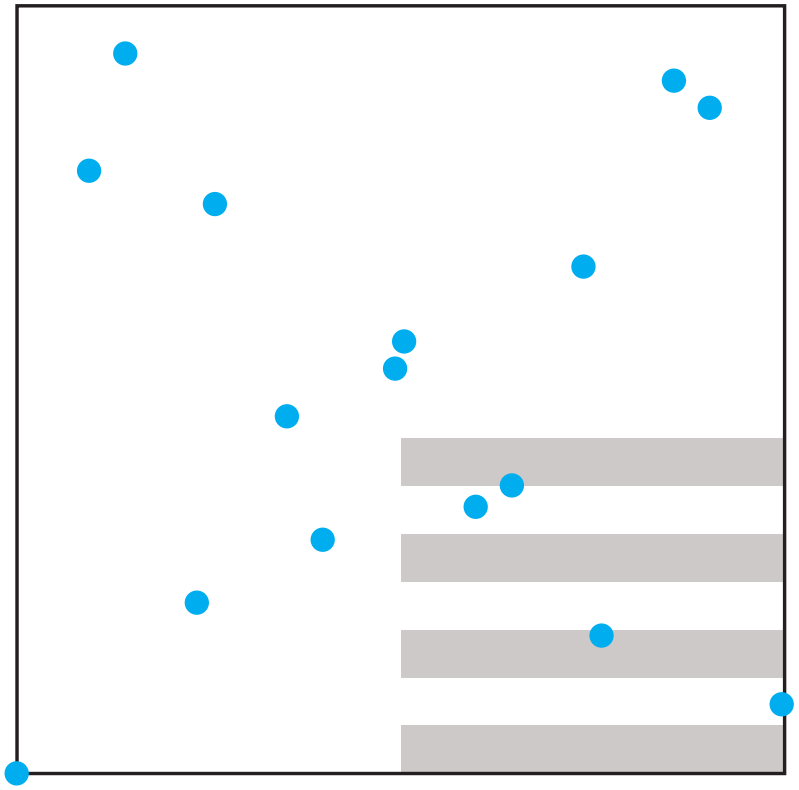
# Interlaced Sobol' point sets with $\alpha = 2$



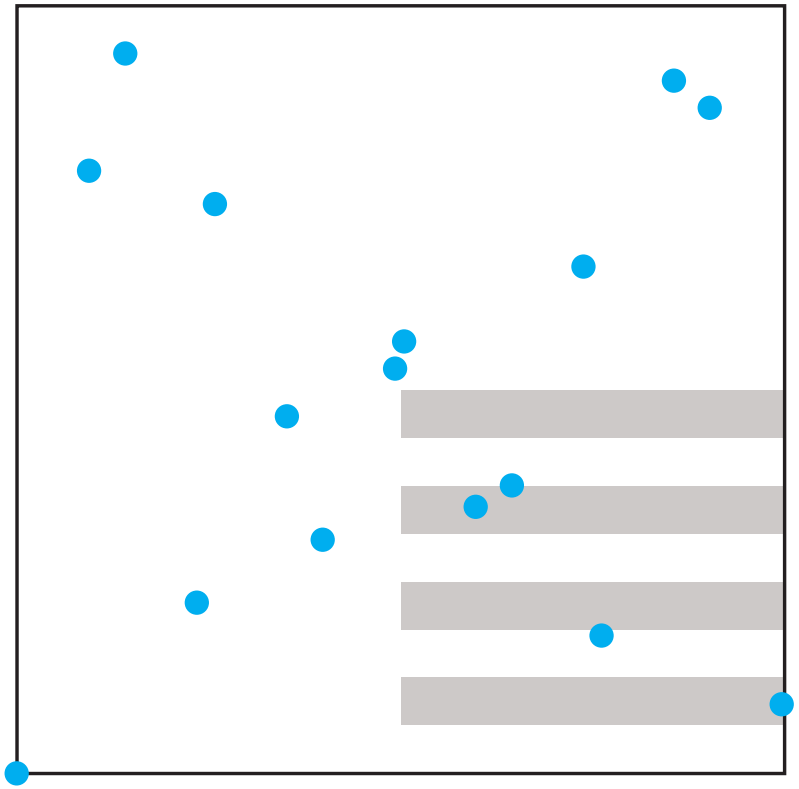
# Interlaced Sobol' point sets with $\alpha = 2$



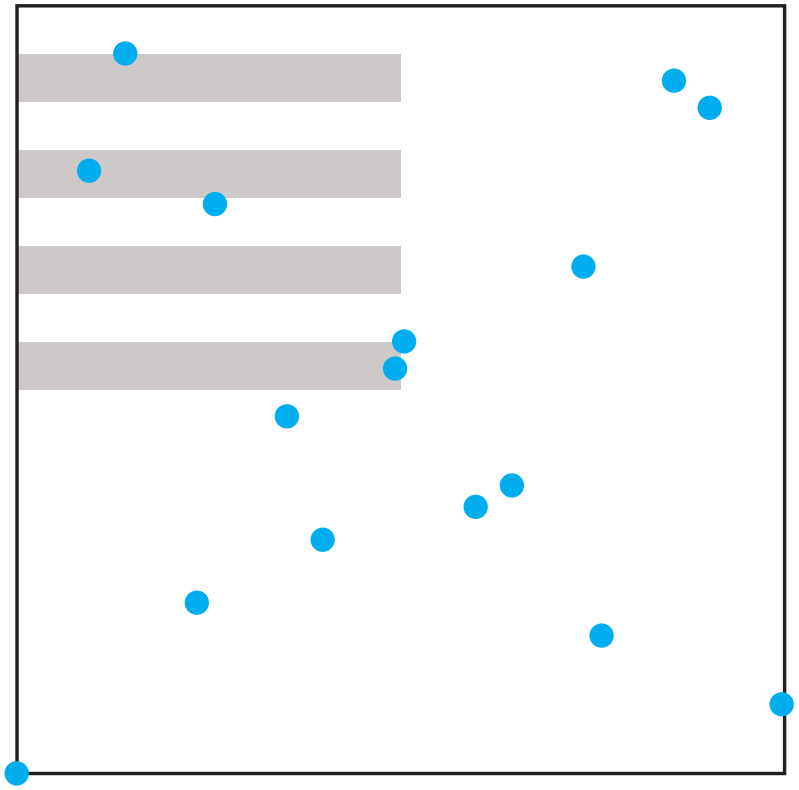
# Interlaced Sobol' point sets with $\alpha = 2$



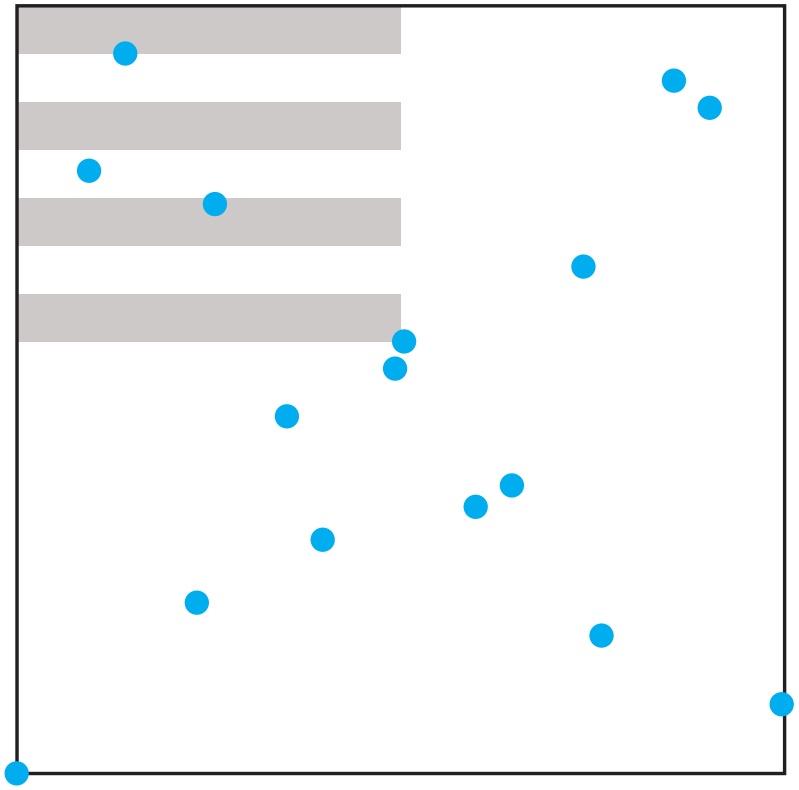
# Interlaced Sobol' point sets with $\alpha = 2$



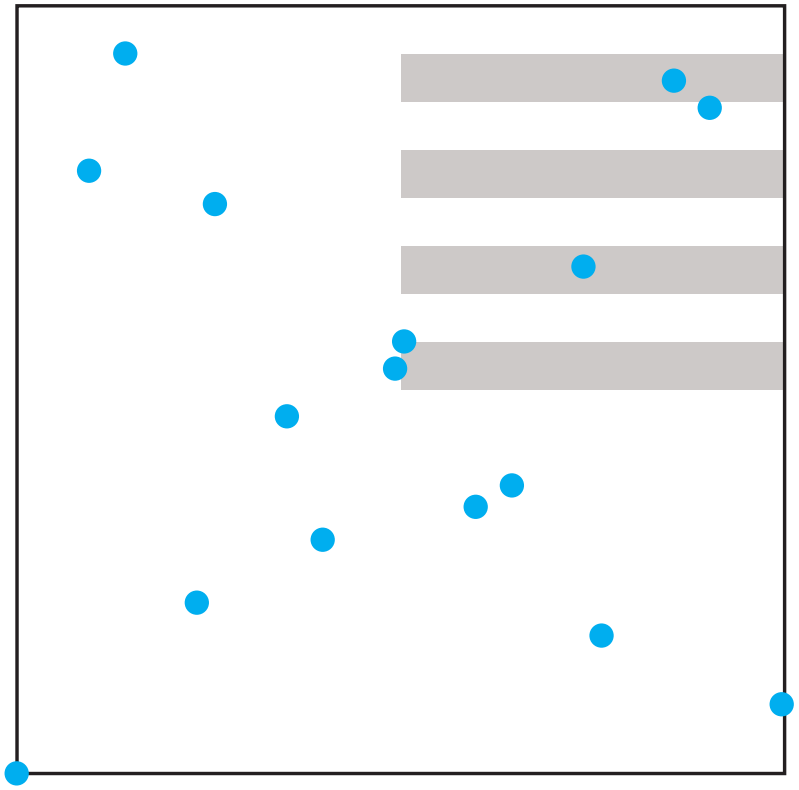
# Interlaced Sobol' point sets with $\alpha = 2$



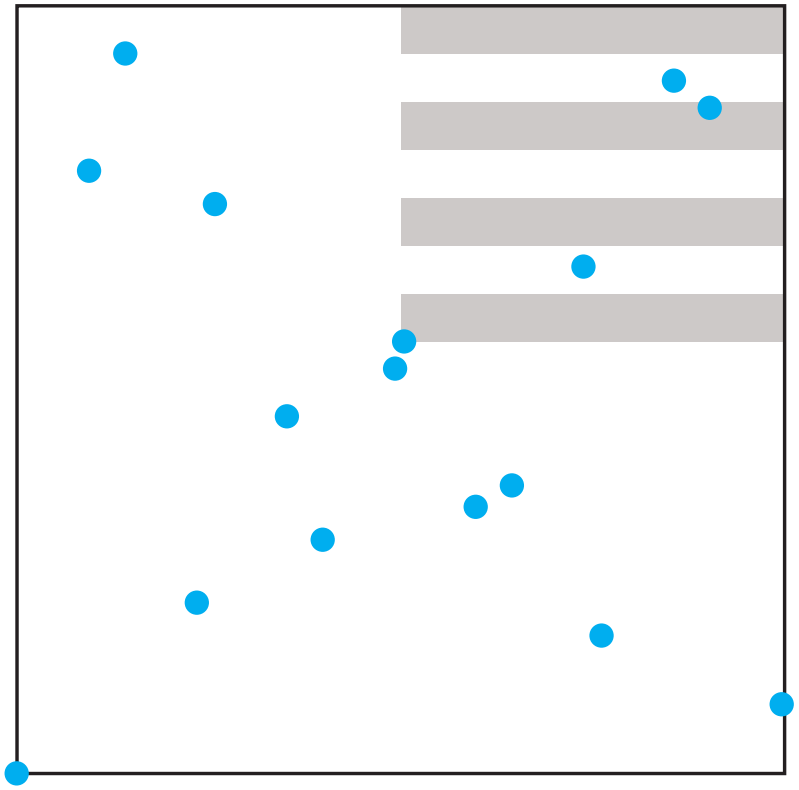
# Interlaced Sobol' point sets with $\alpha = 2$



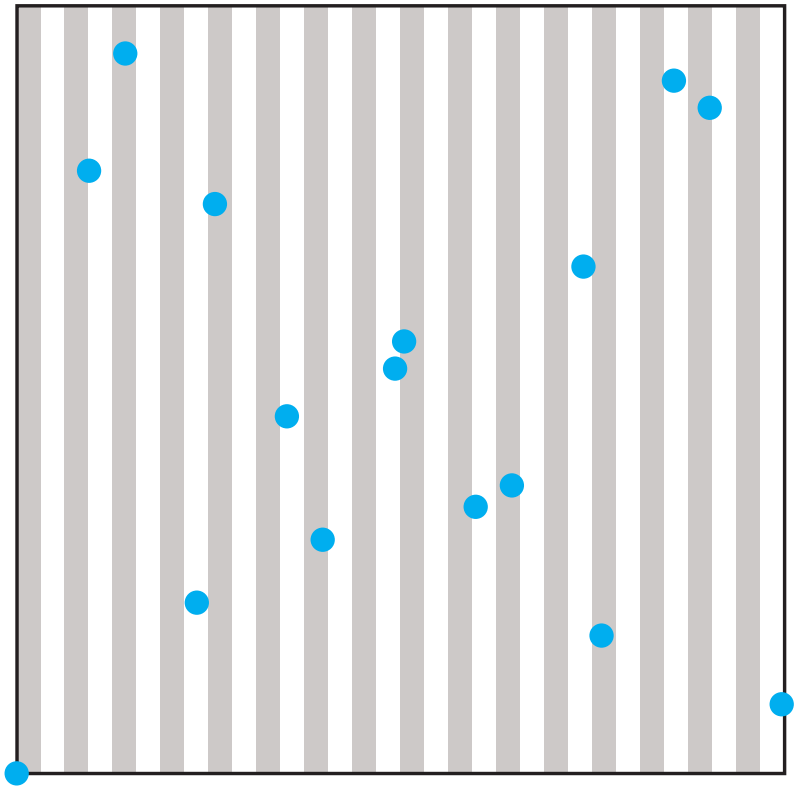
# Interlaced Sobol' point sets with $\alpha = 2$



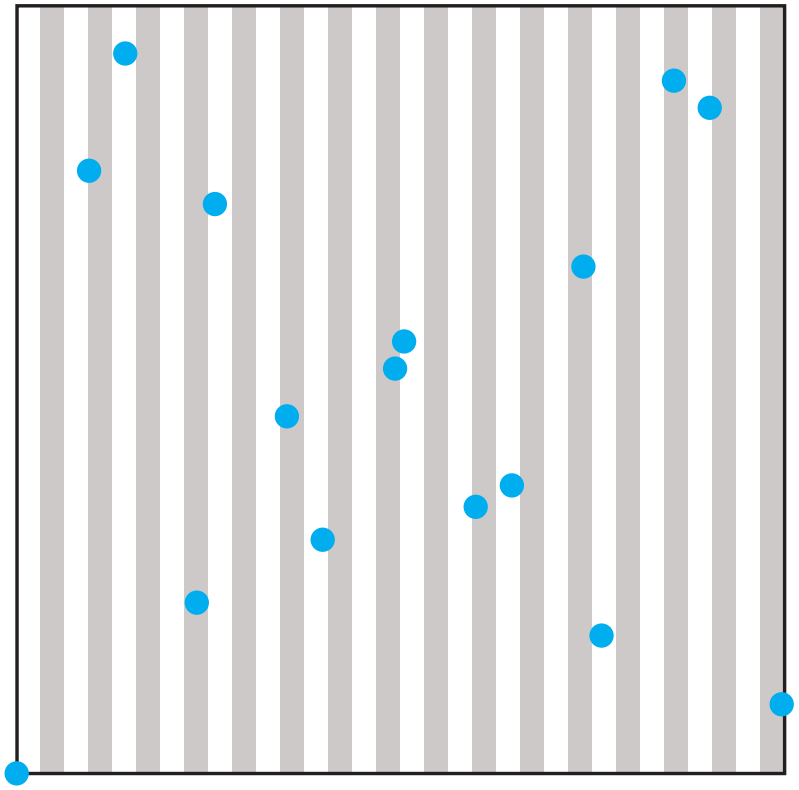
# Interlaced Sobol' point sets with $\alpha = 2$



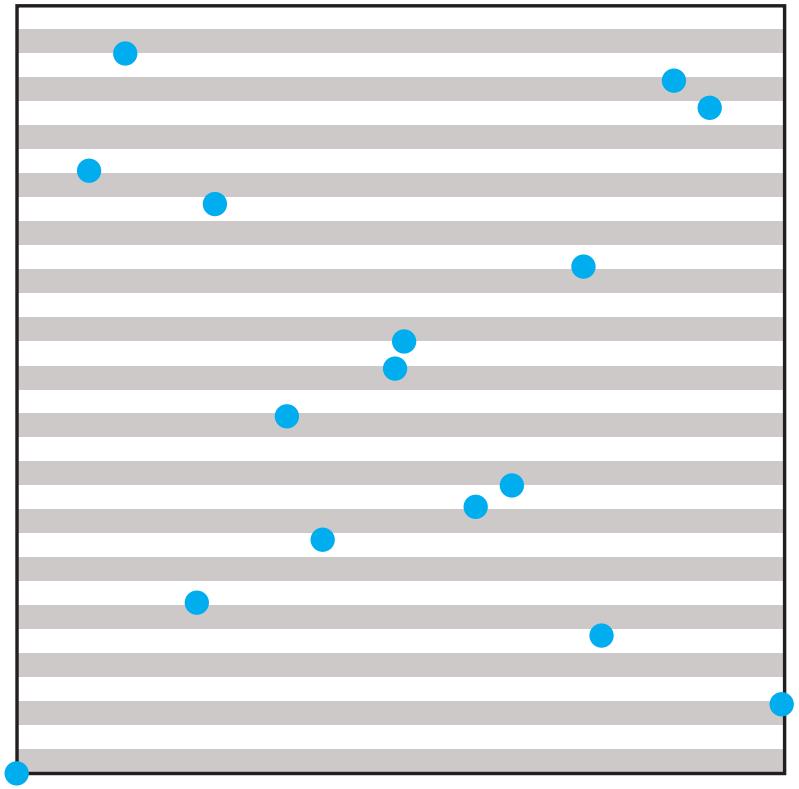
# Interlaced Sobol' point sets with $\alpha = 2$



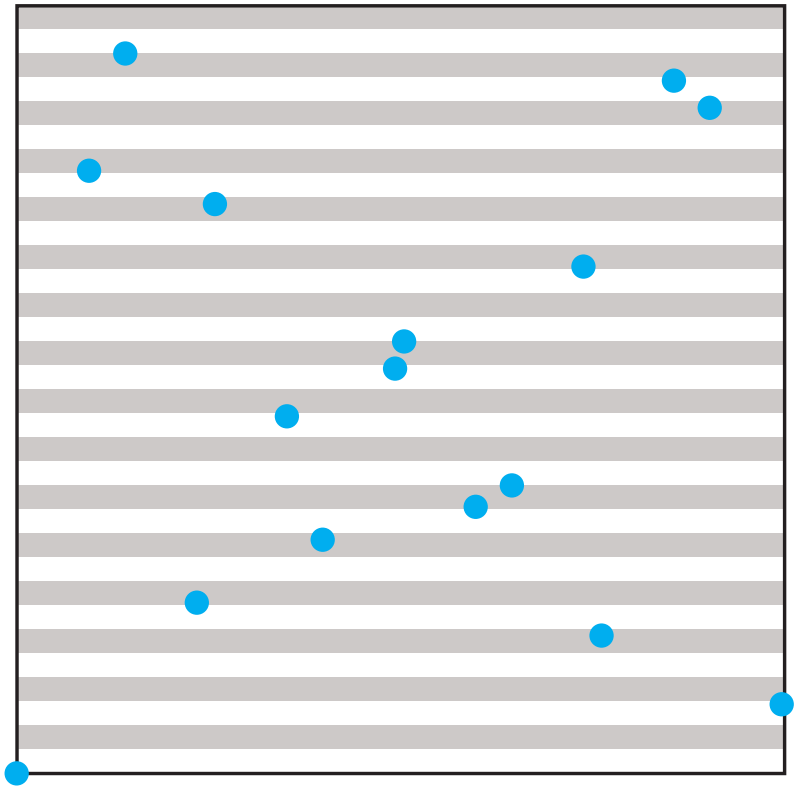
# Interlaced Sobol' point sets with $\alpha = 2$



# Interlaced Sobol' point sets with $\alpha = 2$



# Interlaced Sobol' point sets with $\alpha = 2$



# Interlaced polynomial lattice point sets

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$, and let $\boldsymbol{q} = (q_1, \ldots, q_{\alpha s}) \in (\mathbb{F}_2[x])^{\alpha s}$ with $\deg(q_j) < m$.
- An *interlaced polynomial lattice point set* is just $\mathcal{D}_\alpha(P(p, \boldsymbol{q}))$.

# Interlaced polynomial lattice point sets

- Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$, and let $\boldsymbol{q} = (q_1, \ldots, q_{\alpha s}) \in (\mathbb{F}_2[x])^{\alpha s}$ with $\deg(q_j) < m$.
- An *interlaced polynomial lattice point set* is just $\mathcal{D}_\alpha(P(p, \boldsymbol{q}))$.

- The vector $\boldsymbol{q}$ can be constructed component by component. In the simplest case, the necessary construction cost is of $O(\alpha s N \log N)$ with $O(N)$ memory (G. & Dick, 2015; G., 2015). In applications to PDEs with random coefficients, the criterion sometimes becomes a bit complicated, requiring $O(\alpha s N \log N + \alpha^2 s^2 N)$ construction cost with $O(\alpha s N)$ memory (Dick, Kuo, Le Gia, Nuyens & Schwab, 2014).

# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# Walsh functions

- For $k = (\ldots \kappa_1 \kappa_0)_2 \in \mathbb{N}_0$, the $k$-th Walsh function is defined by

$$\mathrm{wal}_k(x) = (-1)^{\kappa_0 \xi_1 + \kappa_1 \xi_2 + \cdots},$$

where $x = (0.\xi_1 \xi_2 \ldots)_2 \in [0, 1)$, unique in the sense that infinitely many $\xi_i$ are equal to 0.

# Walsh functions

- For $k = (\ldots \kappa_1 \kappa_0)_2 \in \mathbb{N}_0$, the $k$-th Walsh function is defined by

$$\mathrm{wal}_k(x) = (-1)^{\kappa_0 \xi_1 + \kappa_1 \xi_2 + \cdots},$$

  where $x = (0.\xi_1 \xi_2 \ldots)_2 \in [0, 1)$, unique in the sense that infinitely many $\xi_i$ are equal to 0.

- For $s \geq 1$ and $\boldsymbol{k} = (k_1, \ldots, k_s) \in \mathbb{N}_0^s$, the $\boldsymbol{k}$-th Walsh function is defined by

$$\mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) = \prod_{j=1}^{s} \mathrm{wal}_{k_j}(x_j).$$

# Walsh functions

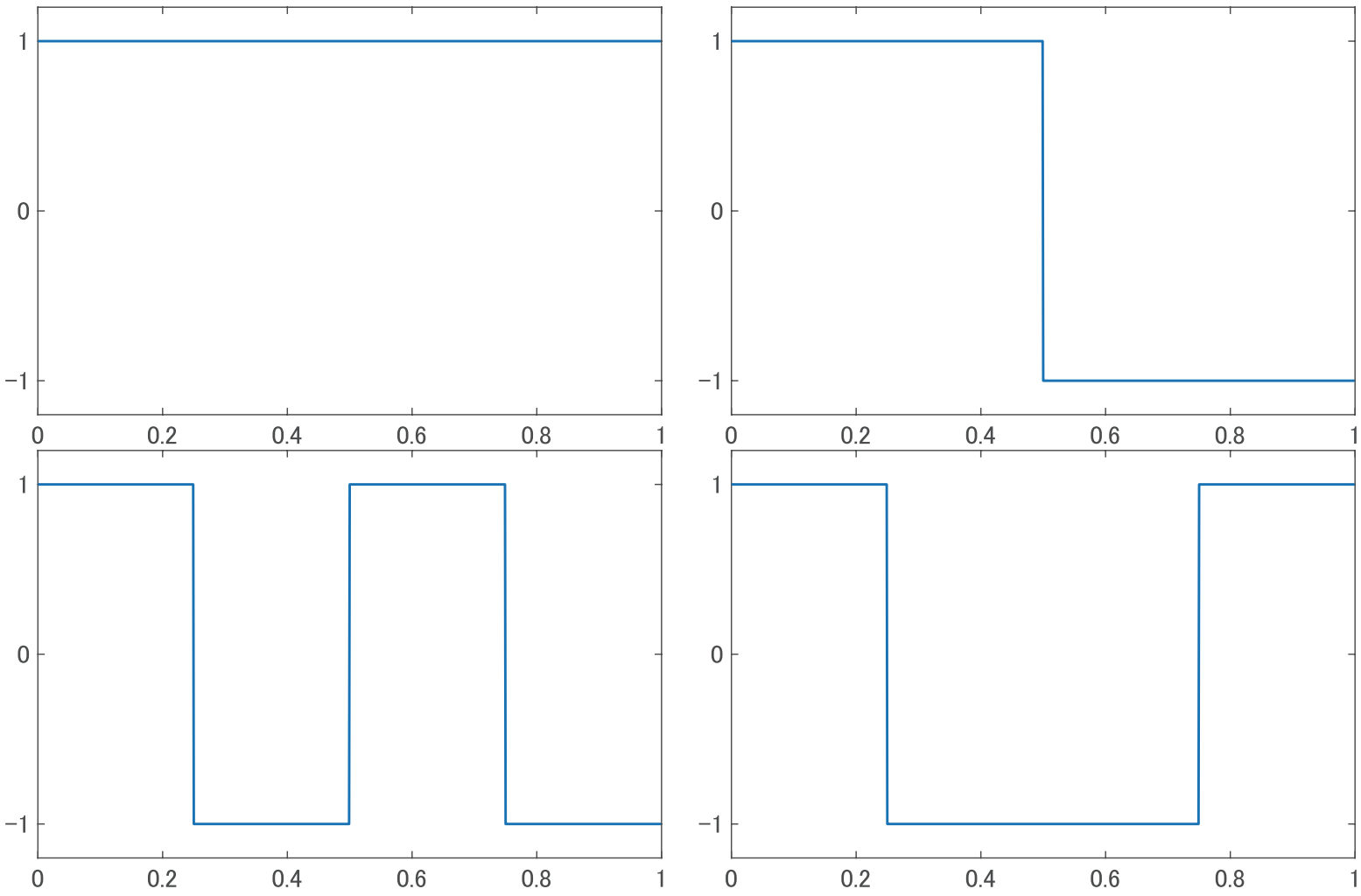



Figure: The $k$-th Walsh functions for $k = 0, 1, 2, 3$

# Walsh functions

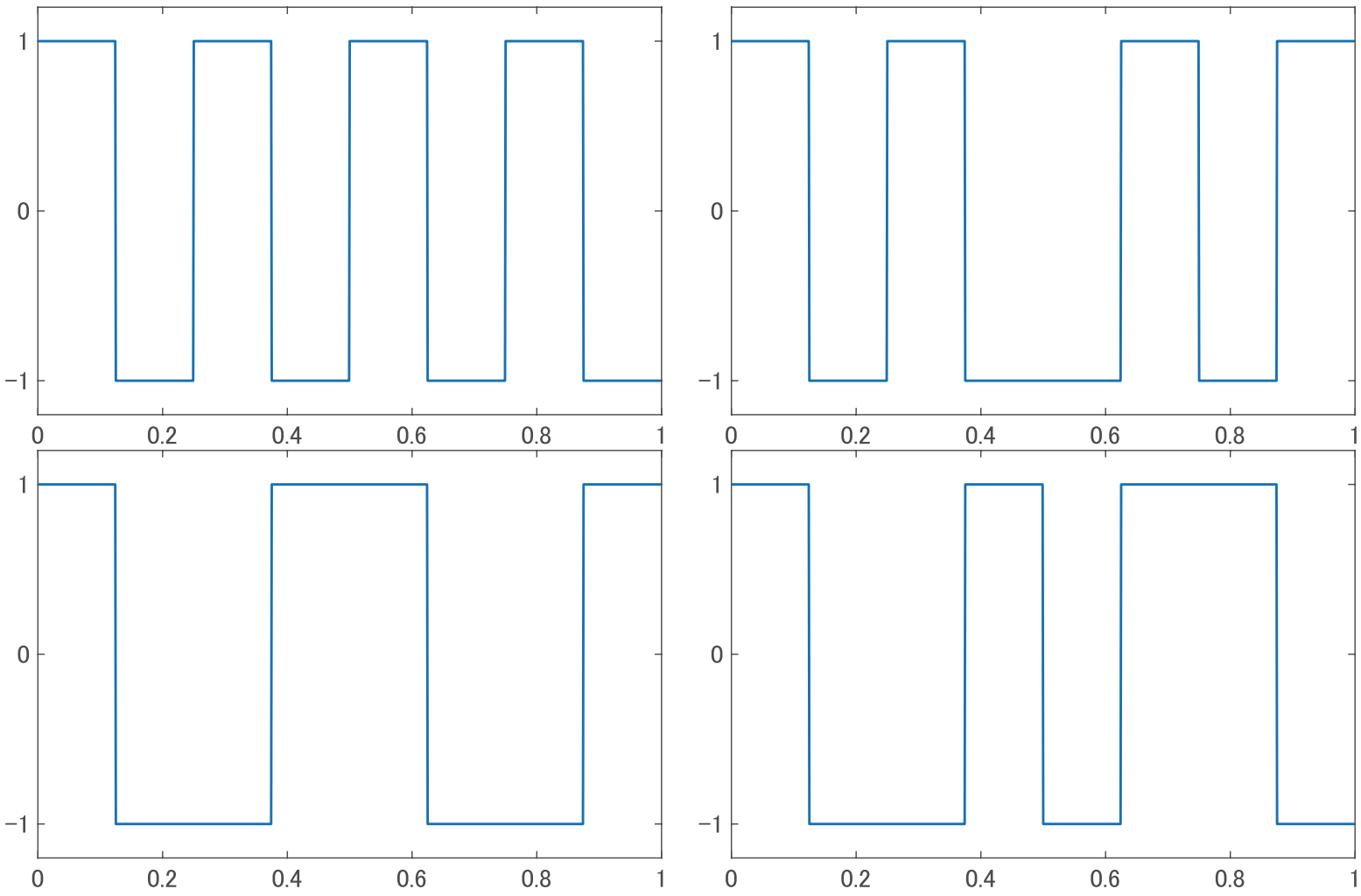



Figure: The $k$-th Walsh functions for $k = 4, 5, 6, 7$

# Walsh functions

- Every Walsh function is a piecewise constant function.

# Walsh functions

- Every Walsh function is a piecewise constant function.

- Importantly the system of Walsh functions $\{\mathrm{wal}_{\boldsymbol{k}} \mid \boldsymbol{k} \in \mathbb{N}_0^s\}$ is a complete orthonormal basis in $L_2([0,1)^s)$. Thus we have a Walsh series of $f$:

$$\sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x})$$

where

$$\hat{f}(\boldsymbol{k}) = \int_{[0,1)^s} f(\boldsymbol{x}) \overline{\mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x})} \, \mathrm{d}\boldsymbol{x}.$$

# Dual nets

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, the dual net is defined by

$$P^\perp = \left\{ \boldsymbol{k} \in \mathbb{N}_0^s \,\middle|\, C_1^\top \begin{pmatrix} \kappa_{1,0} \\ \vdots \\ \kappa_{1,n-1} \end{pmatrix} \oplus \cdots \oplus C_s^\top \begin{pmatrix} \kappa_{s,0} \\ \vdots \\ \kappa_{s,n-1} \end{pmatrix} = \boldsymbol{0} \in \mathbb{F}_2^m \right\},$$

where we write

$$k_j = (\ldots \kappa_{j,1} \kappa_{j,0})_2.$$

# Dual nets

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, the dual net is defined by

$$P^\perp = \left\{ \boldsymbol{k} \in \mathbb{N}_0^s \,\middle|\, C_1^\top \begin{pmatrix} \kappa_{1,0} \\ \vdots \\ \kappa_{1,n-1} \end{pmatrix} \oplus \cdots \oplus C_s^\top \begin{pmatrix} \kappa_{s,0} \\ \vdots \\ \kappa_{s,n-1} \end{pmatrix} = \boldsymbol{0} \in \mathbb{F}_2^m \right\},$$

where we write

$$k_j = (\ldots \kappa_{j,1} \kappa_{j,0})_2.$$

**Trivial fact**

$\boldsymbol{k} \in P^\perp$ if $2^n \mid \boldsymbol{k}$, i.e., $2^n \mid k_j$ for all $j = 1, \ldots, s$.

# Character property

- *Most* of the Walsh functions can be integrated exactly:

$$\frac{1}{2^m}\sum_{\boldsymbol{x}\in P}\mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{k}\in P^{\perp}, \\ 0 & \text{otherwise.} \end{cases}$$

# Character property

- *Most* of the Walsh functions can be integrated exactly:

$$\frac{1}{2^m}\sum_{\boldsymbol{x}\in P}\mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{k}\in P^{\perp}, \\ 0 & \text{otherwise.} \end{cases}$$

- I will also use the following fact:

$$\begin{aligned}
\frac{1}{2^n}\sum_{i=0}^{2^n-1}\mathrm{wal}_k\left(\frac{i}{2^n}\right) &= \frac{1}{2^n}\sum_{\xi_0,\ldots,\xi_{n-1}\in\{0,1\}}\prod_{i=1}^{n}(-1)^{\kappa_{i-1}\xi_i} \\
&= \frac{1}{2^n}\prod_{i=1}^{n}\sum_{\xi_i\in\{0,1\}}(-1)^{\kappa_{i-1}\xi_i} \\
&= \begin{cases} 1 & \text{if } \kappa_0=\ldots=\kappa_{n-1}=0, \text{ i.e., if } 2^n \mid k, \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}$$

# Decomposition of integration error

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, we have

$$Q_P(f) - I(f) = \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0})$$

# Decomposition of integration error

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, we have

$$
\begin{aligned}
Q_P(f) - I(f) &= \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0})
\end{aligned}
$$

# Decomposition of integration error

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, we have

$$
\begin{aligned}
Q_P(f) - I(f) &= \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\}} \hat{f}(\boldsymbol{k})
\end{aligned}
$$

# Decomposition of integration error

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, we have

$$
\begin{aligned}
Q_P(f) - I(f) &= \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\}} \hat{f}(\boldsymbol{k}) \\
&= \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k})
\end{aligned}
$$

# Decomposition of integration error

- For a digital net $P$ with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times m}$, we have

$$
\begin{aligned}
Q_P(f) - I(f) &= \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \frac{1}{2^m} \sum_{\boldsymbol{x} \in P} \mathrm{wal}_{\boldsymbol{k}}(\boldsymbol{x}) - \hat{f}(\boldsymbol{0}) \\
&= \sum_{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\}} \hat{f}(\boldsymbol{k}) \\
&= \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) \\
&= \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \sum_{\substack{\boldsymbol{k} \in \mathbb{N}_0^s \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}).
\end{aligned}
$$

# What is the second term?

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \sum_{\substack{\boldsymbol{k} \in \mathbb{N}_0^s \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k})$$

- The second term is...

# What is the second term?

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \sum_{\substack{\boldsymbol{k} \in \mathbb{N}_0^s \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k})$$

- The second term is...

$$\sum_{\substack{\boldsymbol{k} \in \mathbb{N}_0^s \setminus \{\boldsymbol{0}\} \\ 2^n \mid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) = \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \frac{1}{2^{ns}} \sum_{i_1, \ldots, i_s = 0}^{2^n - 1} \mathrm{wal}_{\boldsymbol{k}} \left( \frac{i_1}{2^n}, \ldots, \frac{i_s}{2^n} \right) - \hat{f}(\boldsymbol{0})$$

$$= \frac{1}{2^{ns}} \sum_{i_1, \ldots, i_s = 0}^{2^n - 1} \sum_{\boldsymbol{k} \in \mathbb{N}_0^s} \hat{f}(\boldsymbol{k}) \mathrm{wal}_{\boldsymbol{k}} \left( \frac{i_1}{2^n}, \ldots, \frac{i_s}{2^n} \right) - I(f)$$

$$= \frac{1}{2^{ns}} \sum_{i_1, \ldots, i_s = 0}^{2^n - 1} f \left( \frac{i_1}{2^n}, \ldots, \frac{i_s}{2^n} \right) - I(f)$$

**the integration error for a regular grid!**

# Digital net as a subset of regular grid

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \left[ \frac{1}{2^{ns}} \sum_{i_1, \ldots, i_s = 0}^{2^n - 1} f\left( \frac{i_1}{2^n}, \ldots, \frac{i_s}{2^n} \right) - I(f) \right]$$

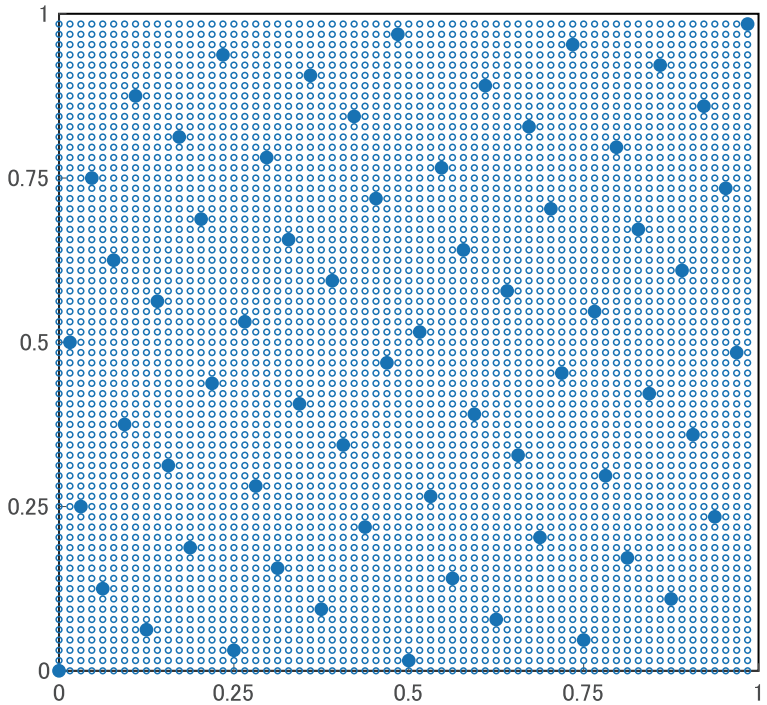



Figure: Hammersley point set for $m = n = 6$ with regular grid

# Euler-Maclaurin

- If $f$ has partial mixed derivatives up to order $\alpha$ in each variable,

$$\frac{1}{2^{ns}} \sum_{i_1,\ldots,i_s=0}^{2^n-1} f\left(\frac{i_1}{2^n},\ldots,\frac{i_s}{2^n}\right) - I(f)$$
$$= \frac{c_1(f)}{2^n} + \cdots + \frac{c_{\alpha-1}(f)}{2^{(\alpha-1)n}} + O(2^{-\alpha n}),$$

  where

$$c_\tau(f) = \sum_{\substack{\tau_1,\ldots,\tau_s \geq 0 \\ \tau_1+\cdots+\tau_s=\tau}} I(f^{(\tau_1,\ldots,\tau_s)}) \prod_{\substack{j=1 \\ \tau_j>0}}^{s} b_{\tau_j}$$

  with $b_i$ denoting the $i$-th Bernoulli number divided by $i!$.
- For a detailed proof of this claim, see Dick, G. & Yoshiki (2019).

# Euler-Maclaurin

- Using this result, we have

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \frac{c_1(f)}{2^n} + \cdots + \frac{c_{\alpha-1}(f)}{2^{(\alpha-1)n}} + O(2^{-\alpha n}).$$

# Euler-Maclaurin

- Using this result, we have

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \frac{c_1(f)}{2^n} + \cdots + \frac{c_{\alpha-1}(f)}{2^{(\alpha-1)n}} + O(2^{-\alpha n}).$$

- Consider high precision, say $n = \alpha m$. Noting that $N = 2^m$, the first term becomes dominant since

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^\perp \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \frac{c_1(f)}{N^\alpha} + \cdots + \frac{c_{\alpha-1}(f)}{N^{\alpha(\alpha-1)}} + O(N^{-\alpha^2}).$$

**Remark:** For functions with dominating mixed smoothness $\alpha$, the integration error cannot be better than $O(N^{-\alpha})$ in general.

# Richardson extrapolation

- Consider the case $n = m$.
- Let us simplify the situation for a moment and consider the following expansions for successive values of $m$:

# Richardson extrapolation

- Consider the case $n = m$.
- Let us simplify the situation for a moment and consider the following expansions for successive values of $m$:

$$
\begin{aligned}
I^{(1)}_{m-\alpha+1} &:= c_0 + \frac{c_1}{2^{m-\alpha+1}} + \frac{c_2}{2^{2(m-\alpha+1)}} + \cdots + \frac{c_{\alpha-1}}{2^{(\alpha-1)(m-\alpha+1)}} \\
&\vdots \\
I^{(1)}_{m-1} &:= c_0 + \frac{c_1}{2^{m-1}} + \frac{c_2}{2^{2(m-1)}} + \cdots + \frac{c_{\alpha-1}}{2^{(\alpha-1)(m-1)}} \\
I^{(1)}_{m} &:= c_0 + \frac{c_1}{2^{m}} + \frac{c_2}{2^{2m}} + \cdots + \frac{c_{\alpha-1}}{2^{(\alpha-1)m}}
\end{aligned}
$$

**Problem:** Estimate $c_0$ from values of $I^{(0)}_{m-\alpha+1}, \ldots, I^{(0)}_{m}$ as precisely as possible (without knowing $c_1, \ldots, c_{\alpha-1}$).

# Richardson extrapolation

- Do the following recursion: For $\tau = 1, \ldots, \alpha - 1$, compute

$$I_n^{(\tau+1)} := \frac{2^\tau I_n^{(\tau)} - I_{n-1}^{(\tau)}}{2^\tau - 1} \quad \text{for } n = m - \alpha + \tau + 1, \ldots, m$$

Schematically this recursion goes like

$$\begin{array}{ccccccccc}
I_{m-\alpha+1}^{(1)} & & I_{m-\alpha+2}^{(1)} & & \cdots & & I_{m-1}^{(1)} & & I_m^{(1)} \\
 & \searrow & \downarrow & \searrow & \cdots & & \downarrow & \searrow & \downarrow \\
 & & I_{m-\alpha+2}^{(2)} & & \cdots & & I_{m-1}^{(2)} & & I_m^{(2)} \\
 & & & \searrow & \cdots & & \downarrow & \searrow & \downarrow \\
 & & & & \ddots & & & \vdots & \\
 & & & & & & I_{m-1}^{(\alpha-1)} & & I_m^{(\alpha-1)} \\
 & & & & & & & \searrow & \downarrow \\
 & & & & & & & & I_m^{(\alpha)}
\end{array}$$

- We will end up with $I_m^{(\alpha)} = c_0$. In reality, we have other terms...

# Use a *set* of digital nets

- For $m - \alpha < n \leq m$, let $P_n$ be a digital net with $C_1, \ldots, C_s \in \mathbb{F}_2^{n \times n}$. Here we do NOT require $P_{m-\alpha+1} \subset P_{m-\alpha+2} \subset \cdots \subset P_m$.

- Richardson extrapolation can push out the intermediate terms of the expansion

$$Q_{P_n}(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P_n^{\perp} \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \frac{c_1(f)}{2^n} + \cdots + \frac{c_{\alpha-1}(f)}{2^{(\alpha-1)n}} + O(2^{-\alpha n})$$

without knowing $c_1(f), \ldots, c_{\alpha-1}(f)$.

# Use a *set* of digital nets

- This way we arrive at an extrapolated QMC rule which satisfies

$$\sum_{n=m-\alpha+1}^{m} w_n Q_{P_n}(f) - I(f) = \sum_{n=m-\alpha+1}^{m} w_n \sum_{\substack{\boldsymbol{k}\in P_n^{\perp}\setminus\{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + O(2^{-\alpha m}),$$

where $w_n$'s depend only on $\alpha$,

$$\sum_{n=m-\alpha+1}^{m} w_n = 1 \quad \text{and} \quad \sum_{n=m-\alpha+1}^{m} |w_n| \le \prod_{i=1}^{\alpha-1} \frac{2^i+1}{2^i-1}.$$

**NOTE:** The total number of points is $2^{m-\alpha+1} + \cdots + 2^m < 2^{m+1}$.

# What we want

$$\sum_{n=m-\alpha+1}^{m} w_n Q_{P_n}(f) - I(f) = \sum_{n=m-\alpha+1}^{m} w_n \sum_{\substack{\boldsymbol{k}\in P_n^{\perp}\setminus\{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + O(2^{-\alpha m}),$$

- If we have digital nets with $C_1, \ldots, C_s \in \mathbb{F}_2^{n\times n}$ which satisfy

$$\sum_{\substack{\boldsymbol{k}\in P_n^{\perp}\setminus\{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) = O(2^{-(\alpha-\varepsilon)n}) \quad \text{with arbitrarily small } \varepsilon > 0,$$

  extrapolated QMC rule achieves the almost optimal rate of convergence for smooth functions.

# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# Truncation operator

- For $x = (0.\xi_1\xi_2\ldots)_2 \in [0,1)$, define the map

$$\mathrm{tr}_m(x) := (0.\xi_1\xi_2\ldots\xi_m 00\ldots)_2,$$

  For a point set $P \subset [0,1)^s$, we write

$$\mathrm{tr}_m(P) = \{\mathrm{tr}_m(\boldsymbol{x}) \mid \boldsymbol{x} \in P\},$$

  where $\mathrm{tr}_m$ is applied componentwise.

- If $P$ is a digital net with $C_1, \ldots, C_s \in \mathbb{F}_2^{n\times m}$ for $n \geq m$, $\mathrm{tr}_m(P)$ is a digital net with **upper** $m \times m$ **submatrices** of $C_1, \ldots, C_s$.

# What G. (2019+) proves

**Theorem**

*Let $P$ be an order $\alpha$ digital $(t, m, s)$-net. For functions with dominating mixed smoothness $\alpha \geq 2$, we have*

$$\sum_{\substack{\boldsymbol{k} \in (tr_m(P))^{\perp} \setminus \{\boldsymbol{0}\} \\ 2^m \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) = O\left(\frac{(\log N)^{\alpha s}}{N^{\alpha}}\right),$$

*with $N = 2^m$.*

**Compare:**

$$\sum_{\boldsymbol{k} \in P^{\perp} \setminus \{\boldsymbol{0}\}} \hat{f}(\boldsymbol{k}) = O\left(\frac{(\log N)^{\alpha s}}{N^{\alpha}}\right) \quad \text{but} \quad \sum_{\boldsymbol{k} \in (\mathrm{tr}_m(P))^{\perp} \setminus \{\boldsymbol{0}\}} \hat{f}(\boldsymbol{k}) = O\left(\frac{1}{N}\right)$$

# Truncation of higher order digital nets

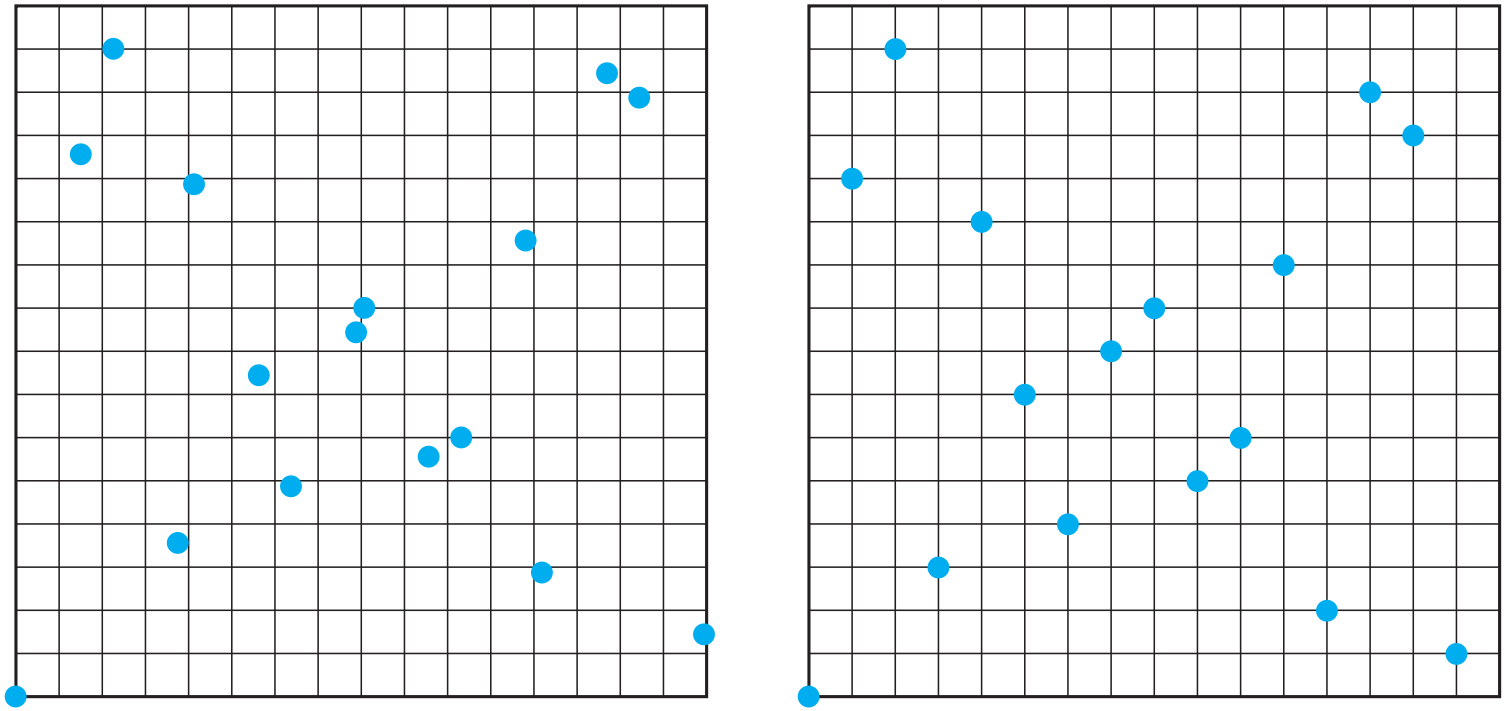


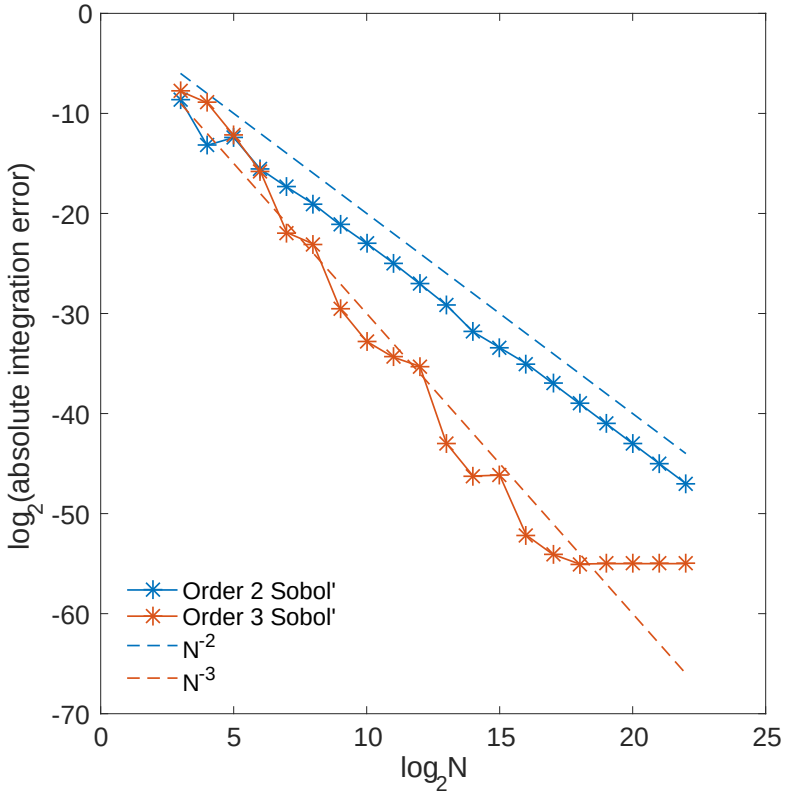
Figure: Interlaced Sobol' point sets: original (left) and after truncation (right)

# Numerical results: original higher order QMC

- Univariate test function:

$$f(x) = x^3 (\log x + 0.25).$$

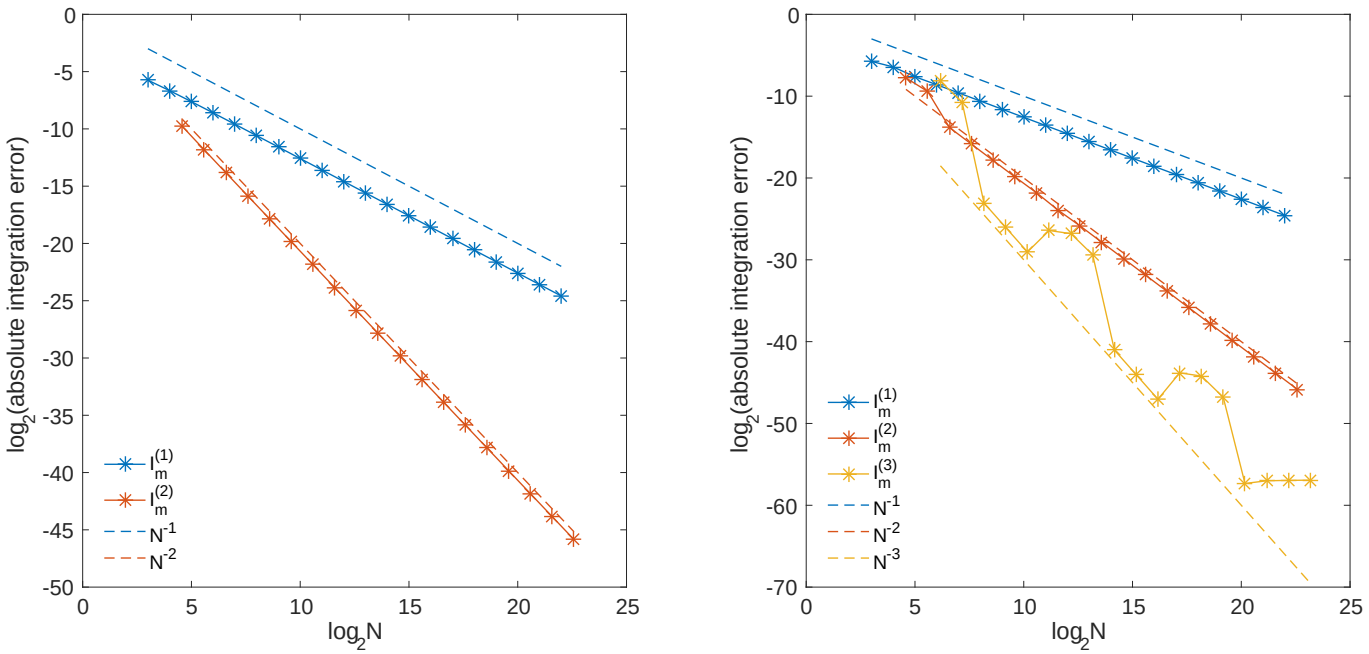
- Interlaced Sobol' sequences: If $\alpha m > 52$, $\mathrm{tr}_{52}$ is applied.



# Numerical results: extrapolation

$$Q_P(f) - I(f) = \sum_{\substack{\boldsymbol{k} \in P^{\perp} \setminus \{\boldsymbol{0}\} \\ 2^n \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) + \frac{c_1(f)}{2^n} + \cdots + \frac{c_{\alpha-1}(f)}{2^{(\alpha-1)n}} + O(2^{-\alpha n}).$$

- Extrapolation applied to *truncated* interlaced Sobol' sequences: order 2 (left) and order 3 (right)

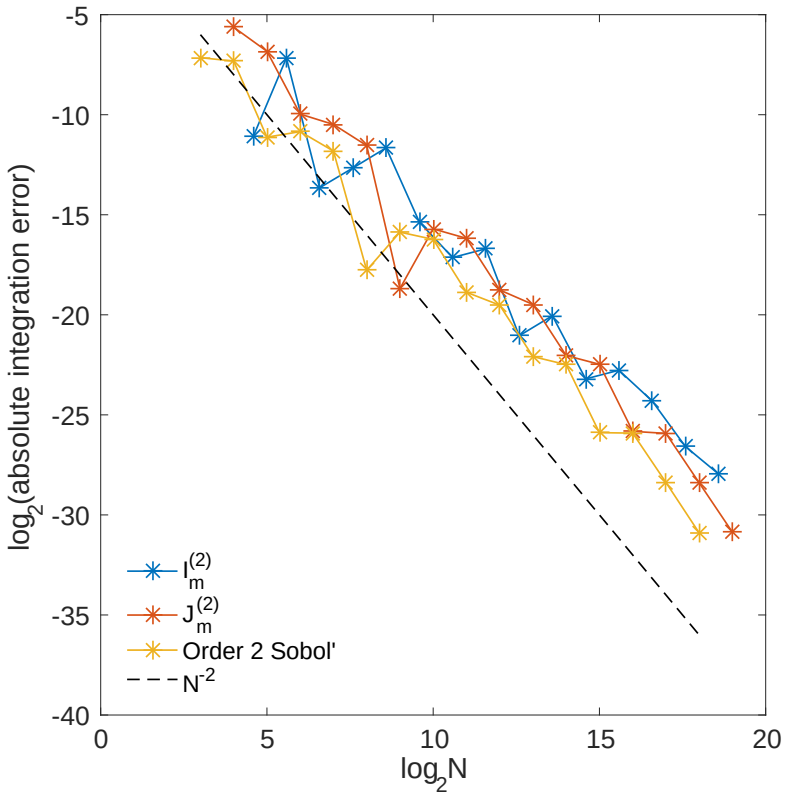


# Numerical results: comparison for large $s$

- Test function:

$$f(\boldsymbol{x}) = \prod_{j=1}^{s} \left[ 1 + \gamma_j \left( x_j^c - \frac{1}{c+1} \right) \right],$$

with $s = 100$, $c = 1.3$ and $\gamma_j = j^{-2}$.



# Outline

- QMC and digital nets/sequences
- Classical QMC
- Higher order QMC
- Richardson extrapolation and QMC
- Application 1: Truncation of higher order nets and sequences
- Application 2: Extrapolated polynomial lattice rules

# What Dick, G. & Yoshiki (2019) prove

**Theorem**

*Let $p \in \mathbb{F}_2[x]$ be irreducible with $\deg(p) = m$. Component-by-component algorithm based on a computable criterion can find good $\boldsymbol{q} \in (\mathbb{F}_2[x])^s$ which satisfies*

$$\sum_{\substack{\boldsymbol{k} \in P(p,\boldsymbol{q})^{\perp} \setminus \{\boldsymbol{0}\} \\ 2^m \nmid \boldsymbol{k}}} \hat{f}(\boldsymbol{k}) = O(2^{-(\alpha-\varepsilon)m}) \quad \text{with arbitrarily small } \varepsilon > 0,$$

*for functions with dominating mixed smoothness $\alpha$.*

- Here "a computable criterion" has been provided by Baldeaux, Dick, Leobacher, Nuyens & Pillichshammer (2012), which stems from the decay of Walsh coefficients for smooth functions.

# Construction cost

- Since we do NOT require $P_{m-\alpha+1} \subset P_{m-\alpha+2} \subset \cdots \subset P_m$, polynomial lattice point set of each size can be constructed independently.

- Construction cost of each is of $O((s+\alpha)n2^n)$. In total, the cost will be of order

$$\sum_{n=m-\alpha+1}^{m} (s+\alpha)n2^n \leq (s+\alpha)N\log_2 N,$$

where $N = 2^{m-\alpha+1} + \cdots + 2^m$. This is better than interlaced polynomial lattice point sets for which we need $O(\alpha s N \log N)$.

# Fast QMC matrix-vector product

- In some applications, we want to estimate the integral of the form

$$\int_{[0,1)^s} f(A\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}.$$

  In terms of computational cost, computing $A\boldsymbol{x}$ can be dominant.

- Even if $A$ does not have any special structure, this computation can be made fast by using (polynomial) lattice point sets (Dick, Kuo, Le Gia, Schwab, 2015).

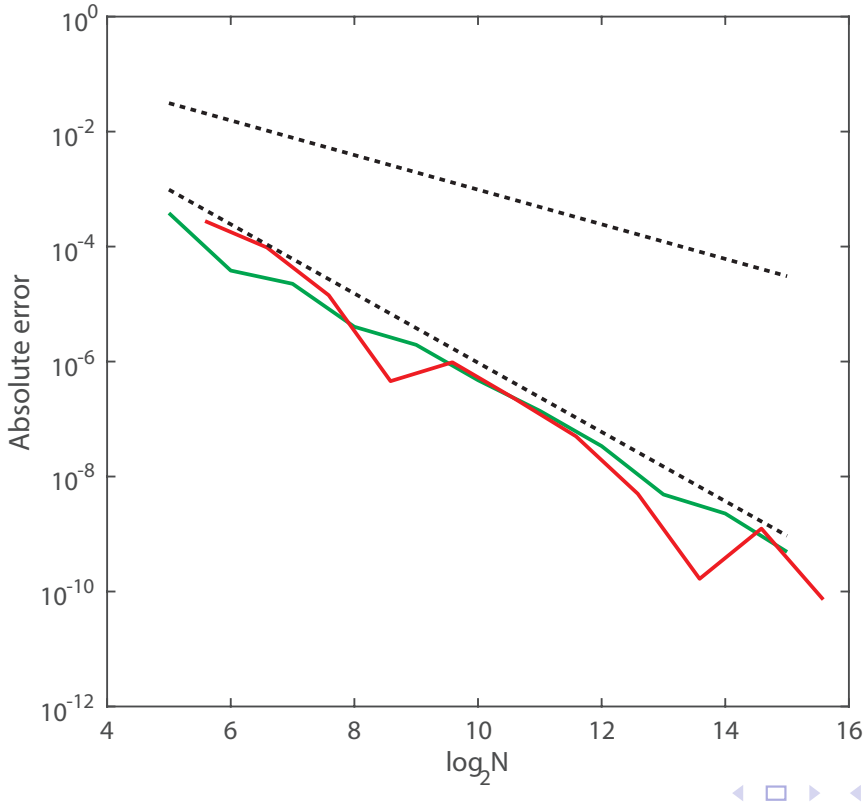
- This strategy does not work for interlaced polynomial lattice rules, whereas it does work for extrapolated polynomial lattice rules.

# Numerical results ($\alpha = 2$)

- Test function:

$$f(\boldsymbol{x}) = \prod_{j=1}^{s} \left[ 1 + \frac{\gamma_j}{1 + \gamma_j x_j^2} \right],$$

  with $s = 100$ and $\gamma_j = j^{-2}$.
- Extrapolated rule with $\alpha = 2$ (red), interlaced rule with $\alpha = 2$ (green)

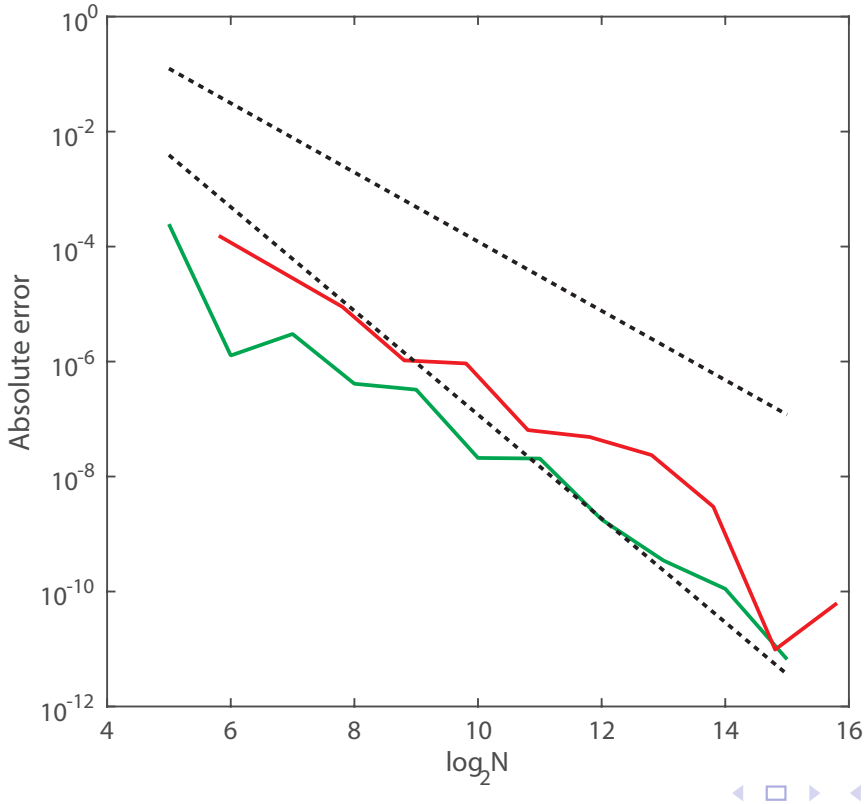


# Numerical results ($\alpha = 3$)

- Test function:

$$f(\boldsymbol{x}) = \prod_{j=1}^{s} \left[ 1 + \frac{\gamma_j}{1 + \gamma_j x_j^2} \right],$$

  with $s = 100$ and $\gamma_j = j^{-2}$.
- Extrapolated rule with $\alpha = 3$ (red), interlaced rule with $\alpha = 3$ (green)



# Conclusions

- As far as I know, this is the first attempt to apply an extrapolation technique to QMC. Roughly speaking, Richardson extrapolation is shown useful for reducing necessary precision significantly.
- Two results are obtained so far
  1. truncation of higher order digital nets/sequences: high-order convergence can be recovered by applying Richardson extrapolation.
  2. introduction of extrapolated polynomial lattice rules: without relying on interlacing, we can construct high-order convergent QMC-based rules.

# Conclusions

- As far as I know, this is the first attempt to apply an extrapolation technique to QMC. Roughly speaking, Richardson extrapolation is shown useful for reducing necessary precision significantly.
- Two results are obtained so far
  1. truncation of higher order digital nets/sequences: high-order convergence can be recovered by applying Richardson extrapolation.
  2. introduction of extrapolated polynomial lattice rules: without relying on interlacing, we can construct high-order convergent QMC-based rules.

- Do we have another good application of extrapolation? What about lattice rules?
- Can we randomize extrapolated QMC rule? Partially yes, if the same randomly chosen digital shift is applied. What about scrambling?

If you are interested, please refer to

- J. Dick, TG, T. Yoshiki: Richardson extrapolation of polynomial lattice rules, SIAM Journal on Numerical Analysis, 57(1):44-69, 2019.
- TG: Richardson extrapolation allows truncation of higher order digital nets and sequences, IMA Journal of Numerical Analysis, online (arXiv:1805.09490).

Thank you for your attention!