EUROGRAPHICS 2018 / D. Gutierrez and A. Sheffer
(Guest Editors)

Volume 37 (2018), Number 2

Sequences with Low-Discrepancy Blue-Noise 2-D Projections

Hélène Perrier[1] David Coeurjolly[1] Feng Xie[2] Matt Pharr[3] Pat Hanrahan[2] Victor Ostromoukhov[1]

[1]Université de Lyon, CNRS, LIRIS, France [2]Stanford, USA [3] Google, USA

In this supplementary material, Section 1 details our sampler and demonstrate its properties. We first define useful concepts: $(q_1,\ldots,q_s)$-equidistributed sets (which are the basis for (t,k,s)-nets) and the relationship between equidistribution and discrepancy (Section 1.1). We also present the Sobol sequence (since our sampler relies on its properties), and look at the binary representation of points. Once those concepts are detailed, we define the notations used in this document and present our sampler in finer details (Section 1.2). Finally, from this fine understanding of our method, we demonstrate that our 2-D permutation preserves the equidistribution of the samples in Lemmas 2 , 3 and 4 (Section 1.3). Section 2 details theoretical rationale behind the LD perserving set of admissible permutations Π. Then, Section 3.2 compares classical Owen's scrambling to our minor modification of it (to create a hierarchical Owen's scrambling). Finally, Section 4 presents our adaptivity, and Section 5 presents the pseudo code for direct generation of the i^{th} sample.

Contents

s	the number of dimensions
K	the subdivision factor
λ	the level of subdivision
$\mathbf{x}$	a s-D point
$\mathbf{x}_d$	the d^{th} coordinate of $\mathbf{x}$
x_d^i	the i^{th} bit in binary representation of the $\mathbf{x}_d$
$T_\mathbf{r}^\lambda$	A tile at a level λ and with corner $\mathbf{r}$
$\mathcal{S}^\lambda := \{\{\mathbf{so}^{(i)}\}_\mathbf{r}^\lambda\}$	A set of first $K^{s\lambda}$ Sobol samples, indexed from 0 to $K^{s\lambda}-1$
$\{\mathbf{so}\}_\mathbf{r}^\lambda$	The set of Sobol samples in the tile $T_\mathbf{r}^\lambda$
$\mathcal{P}^\lambda$	The permuted set obtained from $\mathcal{S}^\lambda$
$\mathcal{V}^\lambda$	The set of all the shift vectors
$\mathcal{Q}^\lambda$	the union of all the sets obtained by shifting

Table 1: *Notations. Note that contrary to the notations used in the paper, the tiles are here indiced using the coordinate of their lower corner instead of the index of the tile in the set of tiles at level λ.*

1. Low discrepancy properties of our scrambling

1.1. Preliminaries

1.1.1. Equidistribution

We start by defining what it means for a set of samples to be a (t,k,s)-net in base b. and a (t,s)-sequence in base b.

Definition 1 (Lemieux, p156) A set of b^k samples is said to be $(q_1,...,q_s)$-equidistributed if for $q = \sum_{i=1}^{s} q_i$, $0 \le q \le k$, the series of elementary intervals $\mathcal{J}^{q_1,...,q_s}$ with

$$\mathcal{J}^{q_1,...,q_s} := \prod_{i=1}^{s} \left[\frac{r_i}{b^{q_i}}, \frac{r_i+1}{b^{q_i}}\right),$$

for $0 \le r_i < b^{q_i}$ contain b^{k-q-t} samples.

An example of such intervals is given in Figure 1.

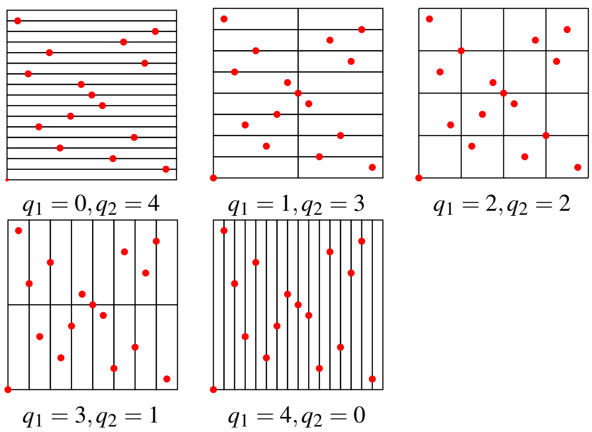


Figure 1: *This set is a (t,k,s)-net in base b, with $t = 0, k = 4, s = 2$, and $b = 2$. It contains $2^4 = 16$ samples. If we list all the possible elementary intervals $\mathcal{J}^{q_1,q_2}$ for $q_1 + q_2 = k$, each of them contains $2^k - q = 2^0$ samples.*

We note that if a set is $(q_1,\dots,q_s)$-equidistributed, each elementary interval $\mathcal{J}^{q_1,\dots,q_s}$ contains b^{k-q-t} samples and each $\mathcal{J}^{q_1,\dots,q_s}$ interval is of area $\left(\frac{1}{b^q}\right)^s$.

We now take a set $(q_1,\dots,q_s)$-equidistributed, with $q' = \sum_{i=1}^{s} q'_i$ and $q' > q$. The new $\mathcal{J}^{q'_1,\dots,q'_s}$ are of size $(\frac{1}{b^{q'}})^s$ and

$$\begin{aligned}\left(\frac{1}{b^{q'}}\right)^s &= \left(\frac{1}{b^{q+(q'-q)}}\right)^s\\ &= \left(\frac{1}{b^q}\frac{1}{b^{q'-q}}\right)^s\\ &= \left(\frac{1}{b^q}\right)^s\left(\frac{1}{b^{q'-q}}\right)^s\end{aligned}$$

This means that each interval $\mathcal{J}^{q'_1,\dots,q'_s}$ covers $b^{q'-q^s}$ intervals $\mathcal{J}^{q_1,\dots,q_s}$. Therefore, each $\mathcal{J}^{q'_1,\dots,q'_s}$ contains $b^{k-q-t}\cdot b^{q'-q} = b^{k-q'-t}$, meaning that if a set is $(q_1,\dots,q_s)$-equidistributed for q, it is also $(q'_1,...,q'_s)$-equidistributed for $\forall q' > q$. This leads to the following property

Property 1 If a point set is $(q_1,\dots,q_s)$-equidistributed in base b for $q = \sum_{i=1}^{s} q_i$ and $q < q'$, then it is also $(q'_1,\dots,q'_s)$-equidistributed in base b.

This is illustrated Figure 2

Definition 2 (Lemieux, p156) A point set P with b^k samples is a (t,k,s)-net iff it is $(q_1,\dots,q_s)$-equidistributed in base b whenever $q \leq k-t$.

Definition 3 (Lemieux, p156) A point set P with b^k samples is a (t,s)-sequence iff within each b-ary segment of the form $\mathbf{x}_{lb^k},\dots,\mathbf{x}_{(l+1)b^k-1}$ with $k \geq t$ and $l \geq 0$ is a (t,k,s)-net in base b.

If a sequence is a (t,s)-sequence, it is a low discrepancy sequence ([Nie88], Eq 3, p 53).

Here, our scrambling applies on sets and therefore cannot be qualified as a low discrepancy sequence using this definition. However, this illustrates the correlation between the discrepancy of a point set and the fact that it is a (t,k,s)-net.

1.1.2. Binary representation

Understanding what is a (t,k,s)-net is necessary to understand the Lemmas 3 and 2. However, another requirement is to understand the binary representation of points in $[0,1)^s$.

Any s-D sample $\mathbf{x} \in [0,1)^s$ can be expressed as $(\mathbf{x}_1,\mathbf{x}_2,\dots,\mathbf{x}_s)$ with

$$\mathbf{x}_d = \sum_{i=1}^{\infty} x_d^i 2^{-1-i}$$

$$\mathbf{x}_d = .x_d^1 x_d^2 x_d^3 \dots$$

where x_d^i is the i^{th} bit of $\mathbf{x}_d$.

Note that the samples are defined in $[0,1)^s$, therefore, x_d^1 is the most significant bit, meaning x_d^1 corresponds to 0.5, x_d^2 to 0.25, ...

1.1.3. Sobol Sequence

The Sobol sequence [Sob67] is a (t,s)-sequence in base 2, created by performing bitwise operation on samples. More precisely, it generates samples from a set of s generative matrices $G_1, G_2, \dots, G^s$, one for each dimension.

To generate the n^{th} sample, we create the vector $a = (a_0, a_1, \dots, a_N)$ with a_i the i^{th} bit of n and $N = \lceil log_2(n) \rceil$. We can then compute the matrix-vector products between a and each G_d in $\mathbb{F}_2$. If we denote $\mathbf{so}^{(n)}$ the n^{th} sample of the Sobol sequence,

$$\mathbf{so} = (a \otimes G_1, \dots, a \otimes G_s).$$

Since the generative matrices G_d are created from primitive polynomials in $\mathbb{F}_2$, the Sobol sequence is a (t,s)-sequence with

$$t = \sum_{d=1}^{s} (n_d - 1)$$

where n_d is the degree of the primitive polynomial for the dimension d [Lem09].

By definition of the Sobol sequence, our set $\mathcal{S}^\lambda$ (defined in the paper as the set containing the first $K^{s(\lambda+1)}$ samples of the sobol sequence with $K := 2^n$, see below), is a $(t, ns(\lambda+1), s)$-net in base $b = 2$. Since Sobol is a (t,k,s)-net in base $b = 2$, it is equidistributed for all intervals of size $[\frac{1}{2^{q_1}},\dots,\frac{1}{2^{q_s}}]$ with $q_1 + \dots + q_s \leq k$.

We note that the family of all intervals of size $\left(\frac{1}{2^{nsq'_1}} \times \dots \times \frac{1}{2^{nsq'_s}}\right)$ is a subset of the family of intervals that are equidistributed for Sobol, with $q_i = nsq'_i$. Therefore

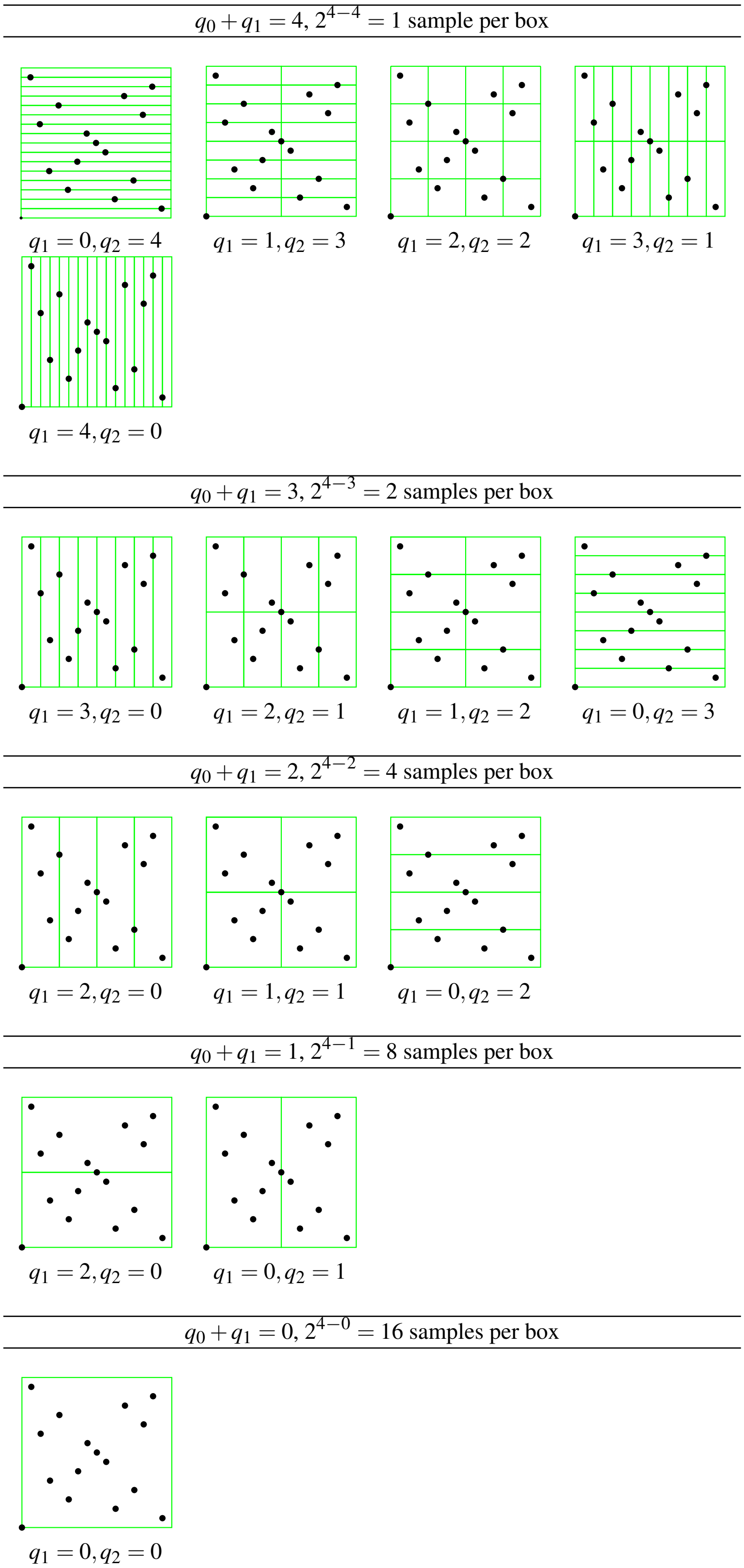


Figure 2: *This set is a (t,k,s)-net in base b, with $t = 0, k = 4, s = 2$, and $b = 2$. It contains $2^4 = 16$ samples. It can be seen that all intervals such that $q_1 + q_2 < 4$ are a union of intervals where $q_1 + q_2 = 4$.*

Lemma 1 $\forall \lambda \in \mathbb{N}$, if $\mathcal{S}^\lambda$ is a $(t, ns(\lambda+1), s)$-net in base $b = 2$, $\mathcal{S}^\lambda$ is a $(\frac{t}{ns}, \lambda+1, s)$-net in base K^s, with $K := 2^n$.

1.1.4. Domain subdivision

Our scrambling considers a regular subdivision of the domain $[0,1)^s$ with a subdivision factor $K := 2^n$, with $n \in \mathbb{N}*$. We denote λ the level of subdivision of the domain (Fig 3).

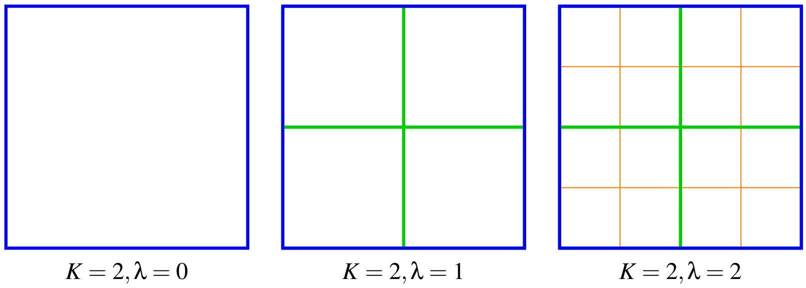


Figure 3: *We subdivide the domain* $[0,1)^s$ *while increasing* λ

This subdivision leads to square tiles. Each time we subdivide the domain, we subdivide the tiles at level λ by the factor K, creating new tiles at level $\lambda+1$. A tile $T_{\mathbf{r}}^\lambda$ is an s-D interval of space defined at a level λ spanning over $\left[\frac{\mathbf{r}_d}{K^\lambda}, \frac{\mathbf{r}_d+1}{K^\lambda}\right)^s$. The union of all the tiles at a level λ forms a partition of the domain $[0,1)^s$ (Fig 4).

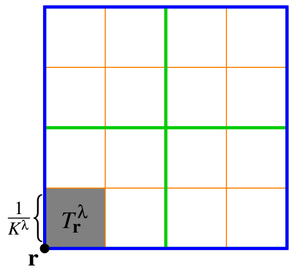


Figure 4: $\lambda = 2, K = 2, s = 2$. *Each interval* $T_{\mathbf{r}}^\lambda$ *will contain* K^s *samples.*

To sample the domain subdivided at a level λ, we will generate the first $K^{s(\lambda+1)} = 2^{ns(\lambda+1)}$ samples from the Sobol sequence [Sob67] (Fig 5).

By definition of the Sobol sequence, $\mathcal{S}^\lambda$ is a (t,k,s)-net in base $b = 2$. Since $\mathcal{S}^\lambda$ contains $2^{ns(\lambda+1)}$ samples, we have $b^k = 2^{ns(\lambda+1)}$ and therefore $\mathcal{S}^\lambda$ is a $(t, ns(\lambda+1), s)$-net in base $b = 2$.

From Lemma 1, we have that if the t value for the Sobol sequence verifies $t \leq ns$, we are guaranteed to have K^s samples inside each tile $T_{\mathbf{r}}^\lambda$. Note that if $t > ns$, we can still have K^s samples in each tile, there is just no guarantee for it. In our paper, we use pairs of indices for the Sobol sequences that do not ensure $t > ns$ but that where found valid empirically.

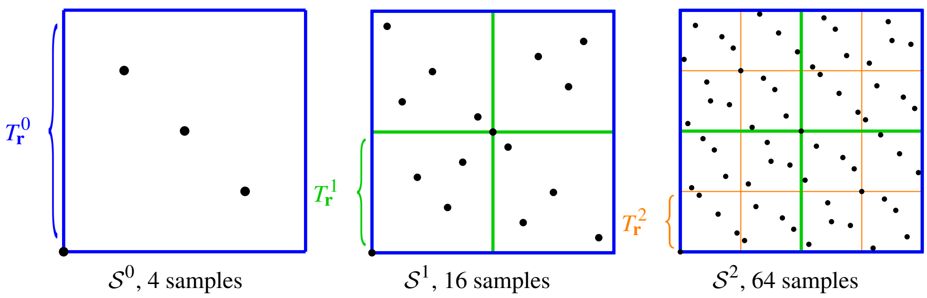


Figure 5: $s = 2, K = 2$. *Generating the first* $K^{s(\lambda+1)}$ *samples of the Sobol sequence with* $t = 0$ *fills each tile* $T_{\mathbf{r}}^\lambda$ *with* K^s *samples.*

We now define $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda} := \mathcal{S}^{\lambda} \cap T_{\mathbf{r}}^{\lambda}$ the set of samples from $\mathcal{S}^{\lambda}$ that belong to the tile $T_{\mathbf{r}}^{\lambda}$. We also define $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda_0,\lambda_1} := \mathcal{S}^{\lambda_0} \cap T_{\mathbf{r}}^{\lambda_1}$ the set of sample from $\mathcal{S}^{\lambda_0}$ that belong to the tile $T_{\mathbf{r}}^{\lambda_1}$ (see Figure 6).

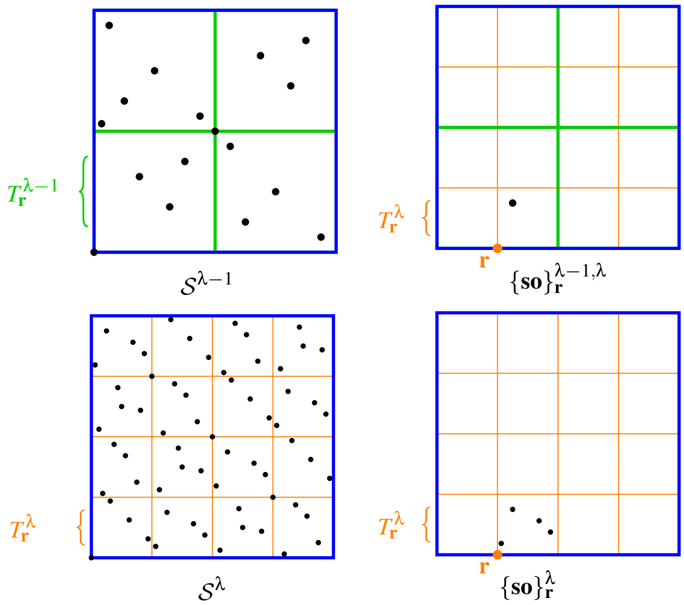


Figure 6: $s = 2, K = 2, \lambda = 2$. *$\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}$ is the intersection between $\mathcal{S}^{\lambda}$ and the tile $T_{\mathbf{r}}^{\lambda}$. Similarly, $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda}$ is the intersection between $\mathcal{S}^{\lambda-1}$ and the tile $T_{\mathbf{r}}^{\lambda}$*

It is important to note that since the samples of $\mathcal{S}^{\lambda}$ come from the Sobol sequence, they are indexed [Sob67]. We will denote $\mathbf{so}^{(i)}$ the i^{th} sample of the sequence $\mathcal{S}^{\lambda}$. A reader familiar with the Sobol sequence may note that the samples of $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}$ are also indexed but their indexes i are not consecutive (Fig 7).

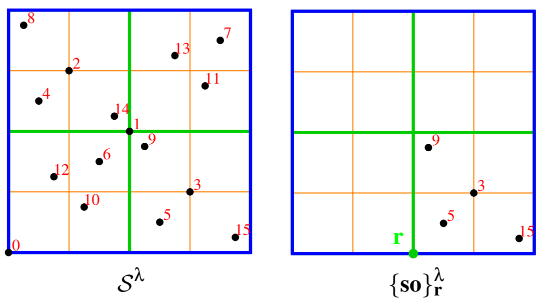


Figure 7: $s = 2, K = 2, \lambda = 1$. *Indexes of the samples in $\mathcal{S}^{\lambda}$ and of the samples in $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}$*

1.2. Our scrambling

We then permute the samples from $\mathcal{S}^{\lambda}$, where $\mathcal{S}^{\lambda}$ is a set containing the first $K^{s(\lambda+1)}$ samples of the Sobol sequence, to generate a new set $\mathcal{P}^{\lambda}$ of samples in $[0,1)^s$ with the following properties:

- If $s = 2$, $\forall\lambda$, if $\mathcal{S}^{\lambda}$ is a $(t, \lambda+1, s)$-net in base K^s, $\mathcal{P}^{\lambda}$ is a $(t, \lambda+1, s)$-net in base K^s (Lemma 4).
- $\forall\lambda$, $\mathcal{P}^{\lambda}$ presents a user defined Fourier spectrum.
- $\forall\lambda > 0$, $\mathcal{P}^{\lambda-1} \subset \mathcal{P}^{\lambda}$ (otherwise P wouldn't be a sequence), and the samples in $\mathcal{P}^{\lambda}$ are indexed (Lemma 5).

Our algorithm creates a set $\mathcal{P}^{\lambda}$ by permuting the set $\mathcal{S}^{\lambda}$ while taking as input the two sets, $\mathcal{P}^{\lambda-1}$ and $\mathcal{S}^{\lambda-1}$. Note that we define $\mathcal{P}^{-1}$ as the point set containing a single sample $\mathbf{so} = 0$, and $\mathcal{S}^{-1} = \mathcal{P}^{-1}$. The sets $\mathcal{S}^{\lambda}$ and $\mathcal{S}^{\lambda-1}$ are coming from the same specific Sobol sequence such that $\mathcal{S}^{\lambda}$ is stratified $\forall\lambda$ and so that $\mathcal{S}^{\lambda-1} \subset \mathcal{S}^{\lambda}$

Then, our permutation is done in two steps;

- The first permutation, denoted Ψ, applies local digital shifts on $\mathcal{S}^\lambda$ to create a new point set $\mathcal{Q}^\lambda$ such that $\mathcal{P}^{\lambda-1} \subset \mathcal{Q}^\lambda$.
- The second permutation, denoted Π, applies a local permutation $\pi^\lambda_{\mathbf{r}}$ on the K^s samples of each tile $T^\lambda_{\mathbf{r}}$ of $\mathcal{Q}^\lambda$, spatially rearranging the samples, so that the final set $\mathcal{P}^\lambda$ locally presents the targeted Fourier spectrum. We achieve this by optimizing this second permutation in order to control the spatial organisation of samples, and therefore, their Fourier spectrum.

1.2.1. Creating $\mathcal{Q}^\lambda$ from $\mathcal{S}^\lambda$ and $\mathcal{P}^{\lambda-1}$

The aim of our first permutation is to create a set $\mathcal{Q}^\lambda$ from $\mathcal{S}^\lambda$, so that $\mathcal{P}^{\lambda-1} \subset \mathcal{Q}^\lambda$. To achieve this, we first compute a set of vectors, such that each vector will represent the displacement between each sample of $\mathcal{S}^{\lambda-1}$ and $\mathcal{P}^{\lambda-1}$. We do this by xoring each sample of $\mathcal{S}^{\lambda-1}$ with its corresponding sample in $\mathcal{P}^{\lambda-1}$. Since the samples are indexed, two samples are corresponding if they have the same index. Therefore, our new vector set, denoted $\mathcal{V}^{\lambda-1}$ is also ordered, and its i^{th} vector, denoted $\mathbf{v}^{(i)}$ is formally defined as

$$\mathbf{v}^{(i)} := \mathbf{so}^{(i)} \oplus \mathbf{p}^{(i)}, \tag{1}$$

where $\mathbf{so}^{(i)}$ is the i^{th} sample of $\mathcal{S}^{\lambda-1}$ and $\mathbf{p}^{(i)}$ is the i^{th} sample of $\mathcal{P}^{\lambda-1}$.

$\mathcal{S}^{\lambda-1}$ is stratified with a single sample inside each tile $T^\lambda_{\mathbf{r}}$. Thus, each vector $\mathbf{v}^{(i)}$ is associated with one and only one tile $T^\lambda_{\mathbf{r}}$, with $\mathbf{so}^{(i)} \in \{\mathbf{so}\}^\lambda_{\mathbf{r}}$. We denote $\mathbf{v}^\lambda_{\mathbf{r}}$ the vector $\mathbf{v}^{(i)}$ associated with the tile $T^\lambda_{\mathbf{r}}$. We can now define formally $\mathcal{V}^{\lambda-1}$ as

$$\mathcal{V}^{\lambda-1} := \bigcup_{\mathbf{r}} \left[\mathbf{v}^\lambda_{\mathbf{r}}\right] \tag{2}$$

which is illustrated Figure 8.

This leads to the formal definition of $\mathcal{Q}^\lambda$ as

$$\mathcal{Q}^\lambda := \bigcup_{\mathbf{r}} \{\{\mathbf{so}\}^\lambda_{\mathbf{r}} \oplus \mathbf{v}^\lambda_{\mathbf{r}}\} \tag{3}$$

where we define the $\oplus$ operation between a sample $\mathcal{X}$ and a single vector $\mathbf{v}$ as $\mathcal{X} \oplus \mathbf{v} = \mathbf{x}^{(i)} \oplus \mathbf{v}$ where $\mathbf{x} \in \mathcal{X}$. This is illustrated Figure 9.

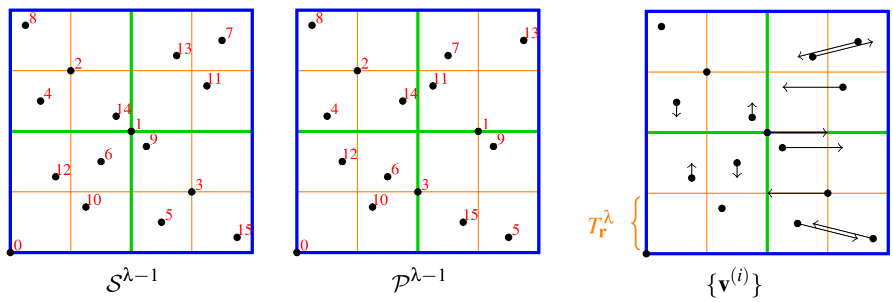


Figure 8: *$s = 2, K = 2, \lambda = 2$. Each vector $\mathbf{v}^{(i)}$ is made by xoring the samples $\mathbf{so}^{(i)} \in \mathcal{S}^{\lambda-1}$ and $\mathbf{p}^{(i)} \in \mathcal{P}^{\lambda-1}$*

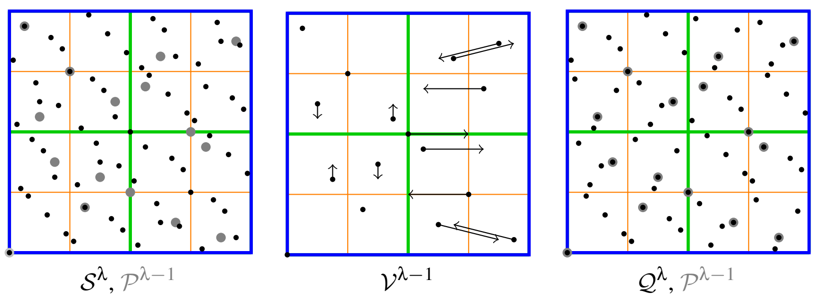


Figure 9: *$s = 2, K = 2, \lambda = 2$. The set $\mathcal{Q}^\lambda$ is made by xoring each samples of each set $\{\mathbf{so}\}^\lambda_{\mathbf{r}}$ with the vector $\mathbf{v}^\lambda_{\mathbf{r}}$ associated to the tile $T^\lambda_{\mathbf{r}}$*

We also note that each pattern $\{\mathbf{so}\}^\lambda_{\mathbf{r}}$ is xored with a single vector to create $\{\mathbf{q}\}^\lambda_{\mathbf{r}}$, thus, as the `xor` operator is net preserving [Lem09], we have the following property for $\mathcal{Q}^\lambda$

Property 2 If $\forall T^\lambda_{\mathbf{r}}$, $N(\{\mathbf{so}\}^\lambda_{\mathbf{r}})$, is a (t, ns, s)-net in base 2, then $\forall T^\lambda_{\mathbf{r}}$, $N(\{\mathbf{q}\}^\lambda_{\mathbf{r}})$, is a (t, ns, s)-net in base 2.

1.2.2. Creating $\mathcal{P}^\lambda$ from $\mathcal{Q}^\lambda$

Then, we permute the samples inside each tile $T_{\mathbf{r}}^\lambda$ of $\mathcal{Q}^\lambda$ to create $\mathcal{P}^\lambda$. Note that there are K^s samples of $\mathcal{Q}^\lambda$ inside $T_{\mathbf{r}}^\lambda$, denoted $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$. Each set $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$ contains a unique sample $\mathbf{p}$ such that $\mathbf{p} \in \mathcal{P}^{\lambda-1}$.

Note that each set $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$ is defined in the domain of its tile. We denote $N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$ the operation that scale this set to the domain $[0,1)^s$, with $N^{-1}(N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)) = \{\mathbf{q}\}_{\mathbf{r}}^\lambda$

We associate with each tile $T_{\mathbf{r}}^\lambda$ a permutation $\pi_{\mathbf{r}}^\lambda$, with $\pi_{\mathbf{r}}^\lambda \in \Pi$, where the set Π contains all the permutations $\pi_{\mathbf{r}}^\lambda$ that have the following properties

Definition 4 If $N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$, is a (t,k,s)-net in base 2, $N(\pi_{\mathbf{r}}^\lambda(\{\mathbf{q}\}_{\mathbf{r}}^\lambda))$ is also a (t,k,s)-net in base 2.

Definition 5 If $\mathbf{p} \in \{\mathbf{q}\}_{\mathbf{r}}^\lambda$ and $\mathbf{p} \in \mathcal{P}^{\lambda-1}$, $\mathbf{p} \in \pi_{\mathbf{r}}^\lambda(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$.

Definition 6 $\pi_{\mathbf{r}}^\lambda$ preserves the 1-D projections of $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$.

We can now formally define $\mathcal{P}^\lambda$ with

$$\mathcal{P}^\lambda := \bigcup_{\mathbf{r}} N^{-1}(\Pi_{\mathbf{r}}^\lambda(N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda))) \tag{4}$$

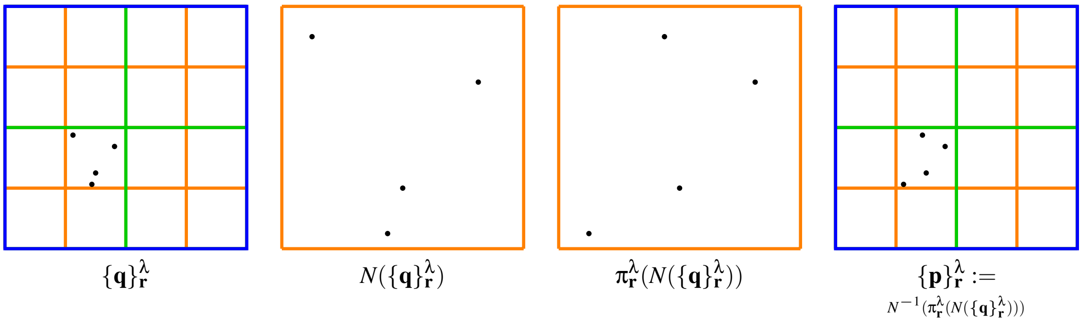


Figure 10: $s=2, K=2, \lambda=2$. *We permute each tile $T_{\mathbf{r}}^\lambda$ from $\mathcal{Q}^\lambda$ to create a new arrangement of samples within each tile with the desired spectral properties.*

An example of an admissible set Π is given in the paper, and will be redefined in Section 2.

1.3. Proofs

We will now prove that our scrambling preserves the net properties of $\mathcal{S}^\lambda$. We will first define some notations, and then we will prove that both Ψ and Π are $(t,\lambda+1,2)$-net preserving in base K^s. Finally, we will prove that $\mathcal{P}^\lambda$ does contain the samples of $\mathcal{P}^{\lambda-1}$, identically indexed.

1.3.1. Notations

In this section we will present all the notations that we will use in the following proofs. Please note that even though our permutation is valid in s-D, it only preserves the net properties in 2-D. Therefore all those notations consider a 2-D domain.

Each 2-D sample (x,y) in a point set $\mathcal{X}$, where all the samples in $\mathcal{X}$ are defined on $ns(\lambda+1)$ bits, and where $\mathcal{X}$ is a $(t,\lambda+1,2)$-net, can be expressed in binary as

$$x = \sum_{i=0}^{ns(\lambda+1)} x^i 2^{-1-i},$$

$$y = \sum_{i=0}^{ns(\lambda+1)} y^i 2^{-1-i},$$

where x^i is the i^{th} bit of x and y^i is the i^{th} bit of y.

We define $x|_a$ (resp. $y|_a$) as a binary truncating of the x (resp. y) coordinate of a sample (x,y) where

$$x|_a = \sum_{i=0}^{a} x^i 2^{-1-i},$$

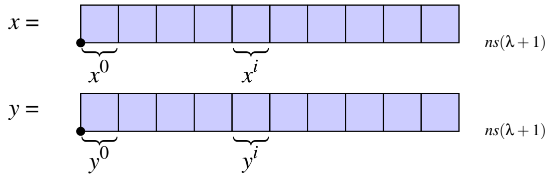


Figure 11: *Binary representation of a sample* $(x,y) \in [0,1)^s$

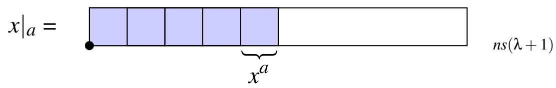


Figure 12: *Binary representation of* $x|_a$

We extend this definition to $(x,y)|_a$, for $(x,y) \in \mathcal{X}$ in a specific way. We thus define $(x,y)|_a$ as

$$(x,y)|_a = (x|_a, y|_{ns(\lambda+1-t)-a})$$

This means that we truncate a total of $ns(\lambda+1-t)$ bits from (x,y), truncating a bits from x and $ns(\lambda+1-t)-a$ bits from y.

Finally, we define $\mathcal{X}|_a$ as $\{(x,y)|_a, \forall (x,y) \in \mathcal{X}\}$, meaning that $\mathcal{X}|_a$ contains truncated samples from $\mathcal{X}$.

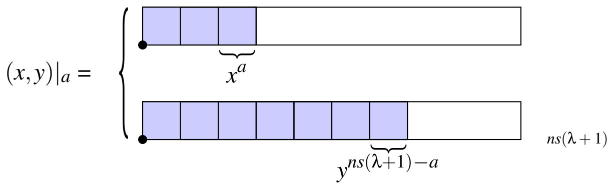


Figure 13: *Binary representation of* $(x,y)|_a$

Each elementary interval $\mathcal{J}^{q_0}$ of $\mathcal{X}$, is of size $\frac{1}{K^{sq_0}}, \frac{1}{K^{s(\lambda+1-t-q_0)}}$, with $0 \leq q_0 \leq \lambda+1-t$,

$$\begin{aligned} q_0 + q_1 &= \lambda+1-t \\ \Leftrightarrow q_1 &= \lambda+1-t-q_0. \end{aligned}$$

It can thus be encoded in binary with nsq_0 bits for x and $ns(\lambda+1-t-q_0) = ns(\lambda+1-t) - nsq_0$ for y. Therefore, for all sample $(x,y) \in \mathcal{X}$, their truncated version $(x,y)|_{nsq_0}$ is equivalent to the minimal corner of the elementary interval $\mathcal{J}^{nsq_0}$ that contains it.

Following this, a point set $\mathcal{X}$ is a $(t,k,2)$-net in base K^s if each sample $(x,y)|_{nsi}$ is present K^{st} times in the set $\mathcal{X}|_{nsi}$, with $0 \leq i < k-t$.

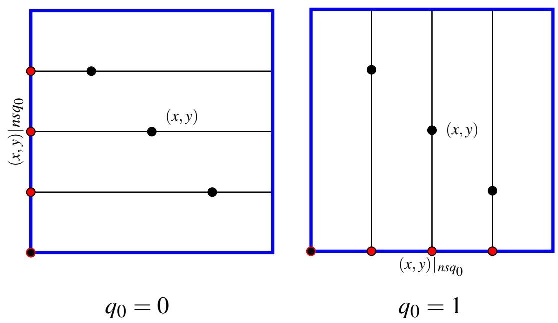


Figure 14: $K = 2, n = 1$. *$\mathcal{Q}^0$ is a $(0,1,s)$-net in base 2^2 iff $\forall q_0 \in \mathbb{N}$ with $0 \leq q_0 \leq 1$, each sample $(x,y)|_{2*2*q_0}, (x,y) \in \mathcal{Q}^0$ appears exactly one time in $\mathcal{Q}^\lambda|_{2\cdot 2\cdot q_0}$.*

The direct corollary is that a permutation Π is $(t,k,2)$-net preserving if $\forall a \in \mathbb{N}, a \leq k-t$,

$$\Pi(\mathcal{X})|_a = \mathcal{X}|_a. \tag{5}$$

Finally, we have one last major property, illustrated Figure 15.

Property 3 $\forall (x,y) \in [0,1)^2$, if $(x|_a, y|_{a'}) = (x'|_a, y'|_{a'})$ then $\forall b \leq a$ and $\forall b' \leq a'$, $(x|_b, y|_{b'}) = (x'|_b, y'|_{b'})$

This property means that if we truncate the x coordinate (resp. y) of the samples of a set to a (resp. b) digits, and if this truncated set is equal to the truncated set from another point set, then the subsets of those sets that are more truncated, with $a' < a$ (resp. $b' < b$), are also equal.

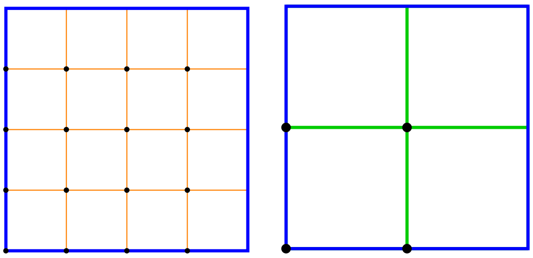


Figure 15: *Geometric illustration of Property 3*

1.3.2. The permutation Π is $(t,\lambda+1,2)$ preserving in base K^2

Lemma 2 For $s=2$, if $\mathcal{Q}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^2, $\mathcal{P}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^2.

Proof The point set $\mathcal{P}^\lambda$ is defined as $\Pi(\mathcal{Q}^\lambda)$, where Π applies an scrambling $\pi_{\mathbf{r}}^\lambda$ over each set $N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$ from $\mathcal{Q}^\lambda$ (Equation 4). The operator N removes the first $n\lambda$ bits of the samples in $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$. Therefore, those first $n\lambda$ bits will be unaffected by the permutation. We apply N^{-1} by re-adding the first $n\lambda$ bits of $\{\mathbf{q}\}_{\mathbf{r}}^\lambda$.
In our particular case, $\pi_{\mathbf{r}}^\lambda$ is not any random permutation but is part of a set of admissible permutations Π (see Section 2 and Section 1.2.2 for details). Each permutation $\pi_{\mathbf{r}}^\lambda \in \Pi$ has two important properties.
First, $\pi_{\mathbf{r}}^\lambda$ preserves the net properties of $N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$, and second, each set $N(\{\mathbf{q}\}_{\mathbf{r}}^\lambda)$ is a $(t', ns, 2)$- net in base 2 as it comes from the Sobol sequence. From here, we can translate property 4 in binary as

$$(x|_{n\lambda+q'_0}, y|_{n\lambda+ns-t'-q'_0}) = \Pi\left((x|_{n\lambda+q'_0}, y|_{n\lambda+ns-t'-q'_0})\right) \tag{6}$$

$\forall q'_0, q'_0 \in \mathbb{N}, 0 \leq q'_0 \leq ns-t', q'_0+q'_1 = ns-t' \Leftrightarrow q'_1 = ns-t'-q'_0$.
Furthermore, since Π preserves the 1-D projections of samples we have for $q'_0 = 0$,

$$(x|_{ns(\lambda+1)}, y|_{n\lambda+ns-t'-q'_0}) = \Pi\left((x|_{ns(\lambda+1)}, y|_{n\lambda+ns-t'-q'_0})\right) \tag{7}$$

and

$$(x|_{n\lambda}, y|_{ns(\lambda+1)}) = \Pi\left((x|_{n\lambda}, y|_{ns(\lambda+1)})\right) \tag{8}$$

In 2-D, the elementary intervals in $\mathcal{Q}^\lambda$ for $q_0+q_1 = \lambda+1-t$ are of size $\frac{1}{2^{nsq_0}} \times \frac{1}{2^{nsq_1}}$ with $q_1 = \lambda+1-t-q_0$. This means that each elementary interval can be encoded using nsq_0 bits for x and $ns(\lambda+1-t-q_0)$ bits for y. As $\mathcal{Q}^\lambda$ is a $(t,\lambda+1,s)$-net in base K^s, each sample $(x,y)|_{nsq_0}, (x,y) \in \mathcal{Q}^\lambda$ appears exactly K^{st} times in $\mathcal{Q}^\lambda|_{nsq_0}$. Therefore, $\forall q_0 \in \mathbb{N}$ with $0 \leq q_0 \leq \lambda+1-t$, each sample from the set $\mathcal{Q}^\lambda|_{nsq_0}$ stands for an elementary interval.
From here, we have 3 possibilities

- Either $nsq_0 \leq n\lambda$, in which case from 3 and equation 8, we have $\mathcal{Q}^\lambda|_{nsq_0} = \mathcal{P}^\lambda|_{nsq_0}$
- Either $ns(\lambda+1-t-q_0) \leq n\lambda$, in which case from 3 and equation 7, we also have $\mathcal{Q}^\lambda|_{nsq_0} = \mathcal{P}^\lambda|_{nsq_0}$
- Or $nsq_0 \geq n\lambda$ and $ns(\lambda+1-t-q_0) \geq n\lambda$. In which case, we seek if there is a q'_0 such that $\mathcal{Q}^\lambda|_{nsq_0} = \mathcal{Q}^\lambda|_{n\lambda+q'_0}$, meaning such that $nsq_0 = n\lambda+q'_0$ and $ns(\lambda+1-t-q_0) = n(\lambda+s)-t'-q'_0$

$$nsq_0 = n\lambda + q'_0$$
$$\Leftrightarrow q'_0 = n(sq_0-\lambda)$$

then

$$\begin{aligned} n(\lambda+s)-t'-q_0' &= n(\lambda+s)-t'-n(sq_0-\lambda) \\ &= n(\lambda+s-sq_0+\lambda)-t' \\ &= n(2\lambda+s-sq_0)-t' \\ &= ns(\lambda+1-\frac{t'}{ns}-q_0) \end{aligned}$$

which means that for $t'=nst$, from equation 6 and property 3, $\mathcal{Q}^\lambda|_{nsq_0}=\mathcal{P}^\lambda|_{nsq_0}$. Note that as t'' is the minimal value such that $N(\{\mathbf{q}\}^\lambda_\mathbf{r})$ is a $(t'',ns,2)$-net, if $t''\le nst$, we have that $N(\{\mathbf{q}\}^\lambda_\mathbf{r})$ is a $(t',ns,2)$ with $t'=nst$.

Therefore from Eq (5), Π preserves the net properties of $\mathcal{Q}^\lambda$ and thus if $\mathcal{Q}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^s, $\Pi(\mathcal{Q}^\lambda)=\mathcal{P}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^2.
□

1.3.3. The permutation Ψ is $(t,\lambda+1,2)$ preserving in base K^2

Lemma 3 For $s=2$, if $\mathcal{S}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^2, $\mathcal{Q}^\lambda$ is a $(t,\lambda+1,2)$-net in base K^2.

Proof
$\mathcal{Q}^\lambda$ is a $(t,\lambda+1,s)$ if $\forall(x,y)\in\mathcal{S}^\lambda$ and $\forall(x',y')\in\mathcal{Q}^\lambda$ we have

$$(x|_{nsa_0},y|_{ns(\lambda+1-t-q_0)})=(x'|_{nsa_0},y'|_{ns(\lambda+1-t-q_0)}), \tag{9}$$

$\forall q_0\in\mathbb{N}$ and $0\le q_0\le\lambda+1$.
From Eq (3), we have

$$\mathcal{Q}^\lambda := \bigcup_\mathbf{r}(\{\mathbf{so}\}^\lambda_\mathbf{r}\oplus\mathbf{v}^\lambda_\mathbf{r}) \tag{10}$$

For simplicity, we will extend the xor operator $\oplus$ to indexed sets, by defining that for two indexed sets P,P', defined at a subdivision level λ, with $\mathbf{p}\in P$, and $\mathbf{p}'\in P'$ with $Card(P)=Card(P')$,

$$P\oplus P'=\bigcup\mathbf{p}^{(i)}\oplus\mathbf{p}'^{(i)}$$

where $\mathbf{p}^{(i)}$ and $\mathbf{p}'^{(i)}$ belong to the same tile $T^\lambda_\mathbf{r}$.
Furthermore, we will define the operator $\phi^\lambda_{\lambda'}$. This operator takes as input a stratified point set at a level $\lambda'\le\lambda$, and duplicates the vector inside each tile $T^{\lambda'}_\mathbf{r}$, in order to repeat this vector inside all the subtiles $T^\lambda_\mathbf{r}$ of $T^{\lambda'}_\mathbf{r}$.
More formally, we define $\phi^\lambda_{\lambda'}$ so that for two ordered sets, $\mathcal{S}^\lambda,\mathcal{S}^{\lambda'}$, where $Card(\mathcal{S}^\lambda)=K^{xs}Card(\mathcal{S}^{\lambda'})$, $x\in\mathbb{N}$, and with $\mathbf{so}\in\mathcal{S}^\lambda$, $\mathbf{so}'\in\mathcal{S}^{\lambda-1}$,

$$\mathcal{S}^\lambda\oplus\phi^\lambda(\mathcal{S}^{\lambda-1})=\bigcup_\mathbf{r}(\{\mathbf{so}\}^\lambda_\mathbf{r}\oplus\{\mathbf{so}'\}^\lambda_\mathbf{r}) \tag{11}$$

where as $\mathcal{S}^{\lambda-1}$ is stratified at level $\lambda-1$, $\{\mathbf{so}'\}^\lambda_\mathbf{r}$ is a singleton.
This operator will let us denote easily the xoring between two point sets defined at different subdivision level (thus with different cardinalities).
From here, we can rewrite Equation 10 as

$$\mathcal{Q}^\lambda := \mathcal{S}^\lambda\oplus\phi^\lambda_{\lambda-1}\mathcal{V}^{\lambda-1}, \tag{12}$$

since the set $\mathcal{S}^\lambda$ contains $K^{s(\lambda+1)}$ samples. From Equation 2, the set $\mathcal{V}^{\lambda-1}$ contains $K^{s\lambda}$ elements, one for each tile $T^\lambda_\mathbf{r}$.
With those new definitions, we denote

$$\pi^{\lambda-1}=\mathcal{V}^{\lambda-1}\oplus\phi(\mathcal{V}^{\lambda-2})$$

, we can thus derive from $\mathcal{V}^{\lambda-1}$:

$$\begin{aligned} &\pi^{\lambda-1}=\mathcal{V}^{\lambda-1}\oplus\phi^{\lambda-1}_{\lambda-2}(\mathcal{V}^{\lambda-2}) \\ \Leftrightarrow&\mathcal{V}^{\lambda-1}=\pi^{\lambda-1}\oplus\phi^{\lambda-1}_{\lambda-2}(\mathcal{V}^{\lambda-2}) \\ \Leftrightarrow&\mathcal{V}^{\lambda-1}=\pi^{\lambda-1}\oplus\phi^{\lambda-1}_{\lambda-2}(\pi^{\lambda-2})\oplus\phi^{\lambda-1}_{\lambda-3}(\mathcal{V}^{\lambda-3}) \\ \Leftrightarrow&\mathcal{V}^{\lambda-1}=\oplus^{\lambda-1}_{i=0}\phi^{\lambda-1}_i(\pi^i) \end{aligned}$$

with $\mathcal{V}^{-1}=0$.

Therefore, $\mathcal{Q}^{\lambda}$ is made from the set $\mathcal{S}^{\lambda}$ through successive $\oplus$ with $\phi(\pi^{i})$. From here, we derive $\forall\lambda', 0 < \lambda' \leq \lambda$,

$$\begin{aligned}
&\mathcal{S}^{\lambda'-1} \oplus \mathcal{P}^{\lambda'-1} = \mathcal{V}^{\lambda'-1} \\
\Leftrightarrow &\mathcal{S}^{\lambda'-1} \oplus \phi^{\lambda'-1}_{\lambda'-2}(\mathcal{V}^{\lambda'-2}) \oplus \mathcal{P}^{\lambda'-1} = \mathcal{V}^{\lambda'-1} \oplus \phi^{\lambda'-1}_{\lambda'-2}(\mathcal{V}^{\lambda'-2}) \\
\Leftrightarrow &\mathcal{Q}^{\lambda'-1} \oplus \mathcal{P}^{\lambda'-1} = \pi^{\lambda'-1} \\
\Leftrightarrow &\mathcal{P}^{\lambda'-1} = \pi^{\lambda'-1} \oplus \mathcal{Q}^{\lambda'-1} \\
\Leftrightarrow &\Pi(\mathcal{Q}^{\lambda'-1}) = \pi^{\lambda'-1} \oplus \mathcal{Q}^{\lambda'-1}
\end{aligned}$$

Therefore, xoring $\mathcal{Q}^{\lambda'}$ with $\pi^{\lambda'}$ is similar to applying the permutation Π that created $\mathcal{P}^{\lambda'}$ from $\mathcal{Q}^{\lambda'}$.
From Lemma 2, if $(x,y) \in \mathcal{S}^{\lambda}$ and $(x',y') \in \mathcal{S}^{\lambda} \oplus \phi(\pi^{\lambda'})$,

$$(x|_{nsq'_0}, y|_{ns(\lambda'+1-t'-q'_0)}) = (x'|_{nsq'_0}, y'|_{ns(\lambda'+1-t'-q'_0)}) \tag{13}$$

Furthermore, since $\pi^{\lambda'} = \mathcal{Q}^{\lambda'} \oplus \phi(\mathcal{P}^{\lambda'})$, when written in binary we have for $\mathbf{p} \in \pi^{\lambda'-1}$

$$\mathbf{p}_d = \sum_{i=0}^{ns(\lambda'+1)} p_d^i 2^{-1-i}$$

where $\mathbf{p}_d$ is the d^{th} coordinate of $\mathbf{p}$ and p_d^i is the i^{th} bit of $\mathbf{p}_d$. We note that $\forall i > ns(\lambda'), p_d^i = 0$.
Geometrically, xoring with $\pi^{\lambda'}$ means that we move all samples that belong to an elementary interval $\mathcal{J}^{q'_0}$ to another $\mathcal{J}^{q'_0}$. However, since all digits of $\mathbf{p} \in \pi^{\lambda'}$ are 0 when $i > ns(\lambda')$, all samples in intervals that are smaller than $\frac{1}{2^{ns(\lambda')}}$ on every dimension, meaning all samples from intervals such that $nsq_0 > nsq'_0$ and $ns(\lambda+1-q_0) > ns(\lambda'+1-q'_0)$, are preserved.

$$(x|_{nsq_0}, y|_{ns(\lambda+1-t-q_0)}) = (x'|_{nsq_0}, y'|_{ns(\lambda+1-t-q_0)}) \tag{14}$$

Furthermore, since we xor with 0 for every digit $> ns(\lambda')$, similarly to the previous proof, we can always reduce the size of the interval on a single dimension, leading to the following equations

$$(x|_{nsq'_0}, y|_{ns(\lambda+1)}) = (x'|_{nsq'_0}, y'|_{ns(\lambda+1)}) \tag{15}$$

$$(x|_{ns(\lambda+1)}, y|_{ns(\lambda'+1-t'-q'_0)}) = (x'|_{ns(\lambda+1)}, y'|_{ns(\lambda'+1-t'-q'_0)}) \tag{16}$$

$\forall q_0 \in \mathbb{N}, 0 \leq q_0 \leq ns(lvl+1) - n(\lambda'+1)$.
From here we have several possibilities for q_0

- $q_0 = q'_0$, then equation 13 equals equation 9.
- $nsq'_0 > nsq_0$ and $ns(\lambda'+1-t'-q'_0) > ns(\lambda+1-t-q_0)$ then from property 3, equation 9 is implied by 13.
- $nsq'_0 < nsq_0$ and $ns(\lambda'+1-t'-q'_0) < ns(\lambda+1-t-q_0)$ then 14 applies.
- $nsq'_0 > nsq_0$ and $ns(\lambda'+1-t'-q'_0) < ns(\lambda+1-t-q_0)$ (or the opposite), equation 15 (or 16) can be used jointly with Property 3 to imply 9

□

1.3.4. $\mathcal{P}^{\lambda}$ is a $(t,\lambda+1,s)$-net in base K^s

Lemma 4 For $s = 2$, if $\mathcal{S}^{\lambda}$ is a $(t,\lambda+1,2)$-net in base K^2, $\mathcal{P}^{\lambda}$ is a $(t,\lambda+1,2)$-net in base K^2.

Proof $\mathcal{P}^{\lambda}$ is made from $\mathcal{S}^{\lambda}$ by successively applying two permutations. First, the Ψ permutation is applied, creating a point set $\mathcal{Q}^{\lambda}$, which is a $(t,\lambda+1,2)$-net in base K^2 from Lemma 3. Then, we create $\mathcal{P}^{\lambda}$ by applying Π to $\mathcal{Q}^{\lambda}$, and as from Lemma 2, Π preserves the net properties of $\mathcal{Q}^{\lambda}$, $\mathcal{P}^{\lambda}$ is a $(t,\lambda+1,2)$-net in base K^2. □

1.3.5. The samples from $\mathcal{P}^{\lambda-1}$ are also into $\mathcal{P}^{\lambda}$

Lemma 5 $\mathcal{P}^{\lambda-1} \subset \mathcal{P}^{\lambda}$

Proof

Since $\mathcal{S}^{\lambda-1}$ is a $(t,\lambda+1,s)$-net in base K^s from property 1 with $t < ns$ and both $\mathcal{Q}^{\lambda-1}$ and $\mathcal{P}^{\lambda-1}$ are $(t,\lambda+1,s)$-nets in base K^s from Lemmas 3 and 2, for $t=0$, each of those sets contains one and only one sample inside each tile $T_{\mathbf{r}}^{\lambda}$.

$$\mathcal{S}^{\lambda-1} = \bigcup_{\mathbf{r}} \{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \tag{17}$$

$$\mathcal{Q}^{\lambda-1} = \bigcup_{\mathbf{r}} \{\mathbf{q}\}_{\mathbf{r}}^{\lambda-1,\lambda} \tag{18}$$

$$\mathcal{P}^{\lambda-1} = \bigcup_{\mathbf{r}} \{\mathbf{p}\}_{\mathbf{r}}^{\lambda-1,\lambda} \tag{19}$$

For the same reason, we have that $\mathcal{S}^{\lambda}$ contains K^s samples inside each tile $T_{\mathbf{r}}^{\lambda}$. As $\mathcal{S}^{\lambda-1} \subset \mathcal{S}^{\lambda}$ from [Sob67], we have that $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \in \mathcal{S}^{\lambda}$. From Eq (3), we have

$$\mathcal{Q}^{\lambda} = \bigcup_{\mathbf{r}} \{\mathbf{so}\}_{\mathbf{r}}^{\lambda} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda} \tag{20}$$

where $\mathbf{v}_{\mathbf{r}}^{\lambda}$ is a singleton defined as

$$\mathbf{v}_{\mathbf{r}}^{\lambda} = \{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \oplus \{\mathbf{p}\}_{\mathbf{r}'}^{\lambda-1,\lambda} \tag{21}$$

with $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda}$ the i^{th} sample in $\mathcal{S}^{\lambda-1}$ and $\{\mathbf{p}\}_{\mathbf{r}'}^{\lambda-1,\lambda}$ the i^{th} sample in $\mathcal{P}^{\lambda-1}$.
Therefore

$$\begin{aligned}
\{\mathbf{so}\}_{\mathbf{r}}^{\lambda} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda} &= (\{\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}/\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda}\} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda}) \cup (\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda}) \\
&= (\{\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}/\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda}\} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda}) \cup (\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \oplus \{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda} \oplus \{\mathbf{p}\}_{\mathbf{r}'}^{\lambda-1,\lambda}) \\
&= (\{\{\mathbf{so}\}_{\mathbf{r}}^{\lambda}/\{\mathbf{so}\}_{\mathbf{r}}^{\lambda-1,\lambda}\} \oplus \mathbf{v}_{\mathbf{r}}^{\lambda}) \cup (\{\mathbf{p}\}_{\mathbf{r}'}^{\lambda-1,\lambda})
\end{aligned}$$

Therefore, $\mathcal{P}^{\lambda-1} \subset \mathcal{Q}^{\lambda}$.
Then, from equation 4 we have

$$\mathcal{P}^{\lambda} := \bigcup_{\mathbf{r}} N^{-1}(\Pi_{\mathbf{r}}^{\lambda}(N(\{\mathbf{q}\}_{\mathbf{r}}^{\lambda}))) \tag{22}$$

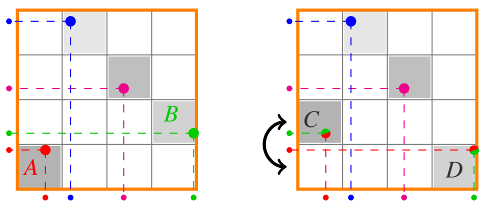
and by property 5, we know that Π preserves the position of the samples from $\mathcal{P}^{\lambda-1}$. Therefore, is $\mathcal{P}^{\lambda-1} \subset \mathcal{Q}^{\lambda}$, $\mathcal{P}^{\lambda-1} \subset \mathcal{P}^{\lambda}$.
□

2. Our local permutations are admissible permutations

In this section, we demonstrate that the set of permutation $\{\pi_r^{\lambda}\}$ built by our method (See main article Section 4) is a subset of the admissible set of permutation Π.

Lemma 6 The local permutations $\{\pi_r^{\lambda}\}$ as defined in Section 4.1 of the main article define an admissible Π.

Proof We already showed in the main article how our permutation can preserve the position of a given marked sample (constraint (i) leading to Lemma 5). We give here more details on how we exchange the trailing bits, to demonstrate that we preserve the net properties of the initial set and its 1-D projections.



Initial set Our permutation

When we remove the trailing bits of a sample, *i.e.* when we quantize this sample to the corner of its elementary interval T_r^{λ} (gray cells in above figure), we do not change the LD net properties at level λ. We then apply Owen's scrambling on this quantized set, which is also net-preserving ([Owe95], Proposition 1). And finally, when we reapply the trailing bits, even if we exchange those bits between the samples, those bits are not significant enough to take a sample out of its elementary interval. As an example, if Owen's scrambling at level λ swaps

rows containing $A := (x, y)$ and $B := (x, y)$, we construct the points $C := (x, y)$ and $D := (x, y)$. This exchange of trailing bits is then net-preserving, and therefore our permutation is net-preserving.
Finally, if the initial set is a (t, k, s)-net, when we quantize it on k digits, all combinations of bits are present. Since Owen's scrambling is net-preserving, after scrambling, all combination of bits will still be represented. Then, simply by reapplying the trailing bits with the proper initial bits, we are guaranteed to preserve the 1-D projections of samples.
So our set $\{\pi_r^\lambda\}$ is a subset of the admissible set of permutation Π. $\square$

3. Higher dimensional sampler construction

In this section, we detail the generalization of our sampler to generate sequences in higher dimension with optimized 2-D projections.

3.1. Constraints induced on the factor K

In order to have the $(t, \lambda+1, 2)$-net in base K^2 property on the optimized pair of dimensions, we need each set $\{\mathbf{so}\}_{\mathbf{r}}^{\lambda} := \mathcal{S}^\lambda \cup T_i^\lambda$ to be a $(t, ns, 2)$-net in base 2 in these dimensions (see Prop. 3). In our implementation, we have used certain combinations of primitive polynomials' indices, found empirically. When K=4, we used for our 4 dimensions the indices $(1, 2)$ and the indices $(3, 7)$. When K=8, we used for our 6 dimensions the indices the indices $(1, 4)$, $(2, 3)$ and the indices $(5, 7)$. Note that as K increases, more and more dimensions can be optimized.

For the remaining dimensions, when the number of samples is fixed, one can use a classical Owen's scrambling of the point coordinates with a tree of fixed depth (please refer to [GRK12b] for efficient implementation). In our hierarchical setting when the number of points is unknown and increases on the fly, we simulate a tree of infinite depth by iteratively expanding a tree of fixed depth. In this process, we ensure that the expansions are consistent with previous trees (in the sense that $\mathcal{P}^{\lambda-1}$ remains in $\mathcal{P}^\lambda$) by enforcing some flags of the expanded tree to zero. In terms of spectral content, this process is barely distinguishable from the original Owen's scrambling and does not affect the discrepancy. The next section details this construction.

3.2. Hierarchical Owen's scrambling

Owen's scrambling [Owe95] relies on a tree of depth d to generate up to 2^d samples. A key feature of our sampler is its adaptive capability and the $\mathcal{P}^{\lambda-1} \subset \mathcal{P}^\lambda$ property. For the first property, we simulate this infinite depth by iteratively expanding a tree of depth d. To do so, we concatenate to each leaf node two new trees of depth d to create a final tree of depth $2d$, that can be used to generate up to 2^{2d} samples. However, when expanding this tree, we need to ensure that we are consistent with the previous tree. Meaning that the expanded tree should lead to the same first 2^d samples than the short tree. To ensure this, we need to enforce the flags applying to the first 2^d samples to 0 in the concatenated trees. This means that each new concatenated subtree must bear 0 flags onto its leftmost branch. This whole process is illustrated Figure 16.

The choice of the parameter d is crucial in this adaptation, if d is too low, there will be too many 0 flags in the tree to improve the spectrum. Figure 17 illustrates this. We note that in practice, for $d = 8$, we no longer see any visible artefacts compared to the original Owen.

4. Adaptive sampling

Since our sampler is adaptive, we can locally change the density of samples. Figure 18 illustrates this with a quadratic ramp function and 19 shows some adaptive results on the Julia set. Our sampler is clearly not as performant as stippling oriented samplers such as [dGBOD12] due to its square tiles and naive ranking. However, it is worth pointing that classical low discrepancy sequences are not natively adaptive. Adaptive variants exist, *e.g.* [GRK12a], but are difficult to implement efficiently. On the contrary our construction is natively adaptive as it is tiled based.

5. Direct access to the i^{th} sample

Directly accessing the i^{th} sample generated by our sampler can be achieved through the following algorithm. We compute the level λ of this sample as

$$\lambda = \lfloor \log_{K^s}(i) \rfloor.$$

We remind the reader that the level 0 is composed of a single tile containing K^s samples, hence the floor function. Note that the i^{th} sample in our sequence, denoted $\mathbf{p}^{(i)}$, can be analytically defined as

$$\mathbf{p}^{(i)} := \Pi_{\mathbf{r}}^{\lambda}\left(\mathcal{S}^{\lambda^{(i)}} \oplus \mathcal{Q}^{\lambda^{(i)}}\right)$$

where $\mathbf{r}$ is the tile at level λ that contains $\mathbf{p}^{(i)}$, and $\mathcal{Q}^{\lambda^{(i)}}$ is the xoring vector that, once applied on the sample $\mathcal{S}^{\lambda^{(i)}}$, moves it to the proper tile $\mathbf{r}$.

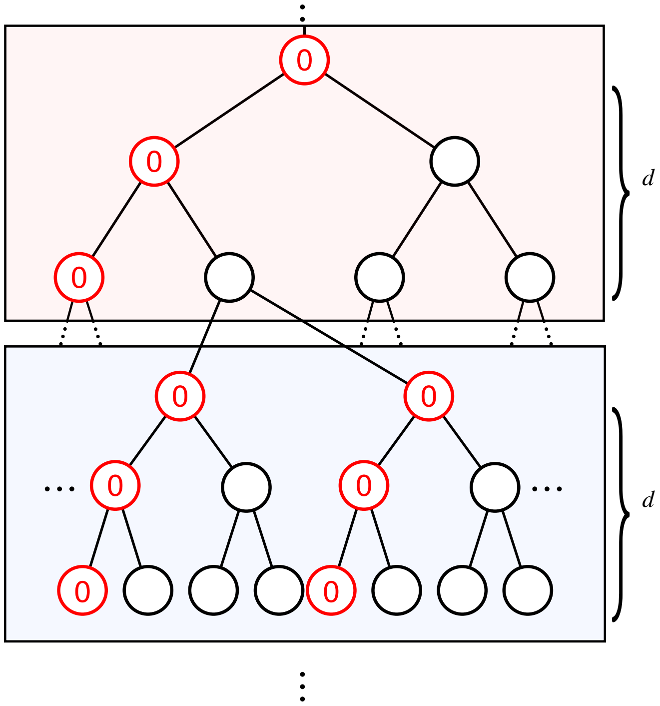


Figure 16: *Extension of an Owen's tree of depth d into a consistent new Owen's tree of depth* $2d$*. Each time we need to generate more than* 2^d *samples, we increase the depth of the tree by concatenating to each leaf node two new subtrees with the right flags at* 0*. All other flags are random.*

To solve this equation, we need to compute first $\mathcal{Q}^{\lambda^{(i)}}$, and then we need to retrieve the permutation $\Pi^{\lambda}_{\mathbf{r}}$. This vector is roughly defined as

$$\mathcal{Q}^{\lambda^{(i)}} = \mathcal{S}^{\lambda^{(i)}} \oplus \mathcal{V}^{\lambda-1^{(i)}}$$

where $\mathcal{V}^{\lambda-1^{(i)}}$ is the displacement vector computed from $\mathcal{S}^{\lambda-1}$ and $\mathcal{P}^{\lambda-1}$, leading to,

$$\mathcal{V}^{\lambda-1^{(i)}} = \mathcal{S}^{\lambda-1^{(i')}} \oplus \mathcal{P}^{\lambda-1^{(i')}}.$$

This index i' here, corresponds to the first Sobol index that falls into the tile $\mathbf{r}'$ to which the i^{th} Sobol sample $\mathbf{so}^{(i)}$ belongs. This index can be retrieved through methods as described in [GRK12a]. Then, we will repeat the process all over again to find $\mathcal{P}^{\lambda-1^{(i')}}$. The process stops when $\lambda = 0$, as $\mathcal{V}^{0^{(i)}} = 0$.

From here, all we have left is to find for all the $\mathcal{P}^{\lambda'^{(i')}}$ their corresponding permutations $\Pi^{\lambda}_{\mathbf{r}}$. This can be done very easily by generating all the samples of $\mathcal{P}^{\lambda'}$ that belong to the tile $\mathbf{r}$, which can be done as we now know the value of $\mathcal{V}^{\lambda-1^{(i)}}$. Then, a simple peek in the look up table gives us the permutation $\Pi^{\lambda}_{\mathbf{r}}$. Now, we can retrieve all the samples $\mathcal{P}^{\lambda+1^{(i)}}$ from the samples $\mathcal{P}^{\lambda^{(i')}}$, until we reach the sample $\mathbf{p}^{(i)}$ that we wanted in the first place.

The pseudo-code of this algorithm is given in Algorithm 1 and has a complexity in

$$O(\log_{K^s}(i)K^s)$$

which is the same complexity as finding all the samples in a given tile $\mathbf{r}$. Therefore, even though contrary to Sobol, we do not generate a point set by generating the samples one by one, we can still retrieve the i^{th} sample more efficiently than by generating n samples and taking the i^{th} one.

References

[ANHD17] Ahmed A., Nies T., Huang H., Deussen O.: An adaptive point sampler on a regular lattice. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2017 36*, 4 (2017), 138:1âĂŞ138:13. 17

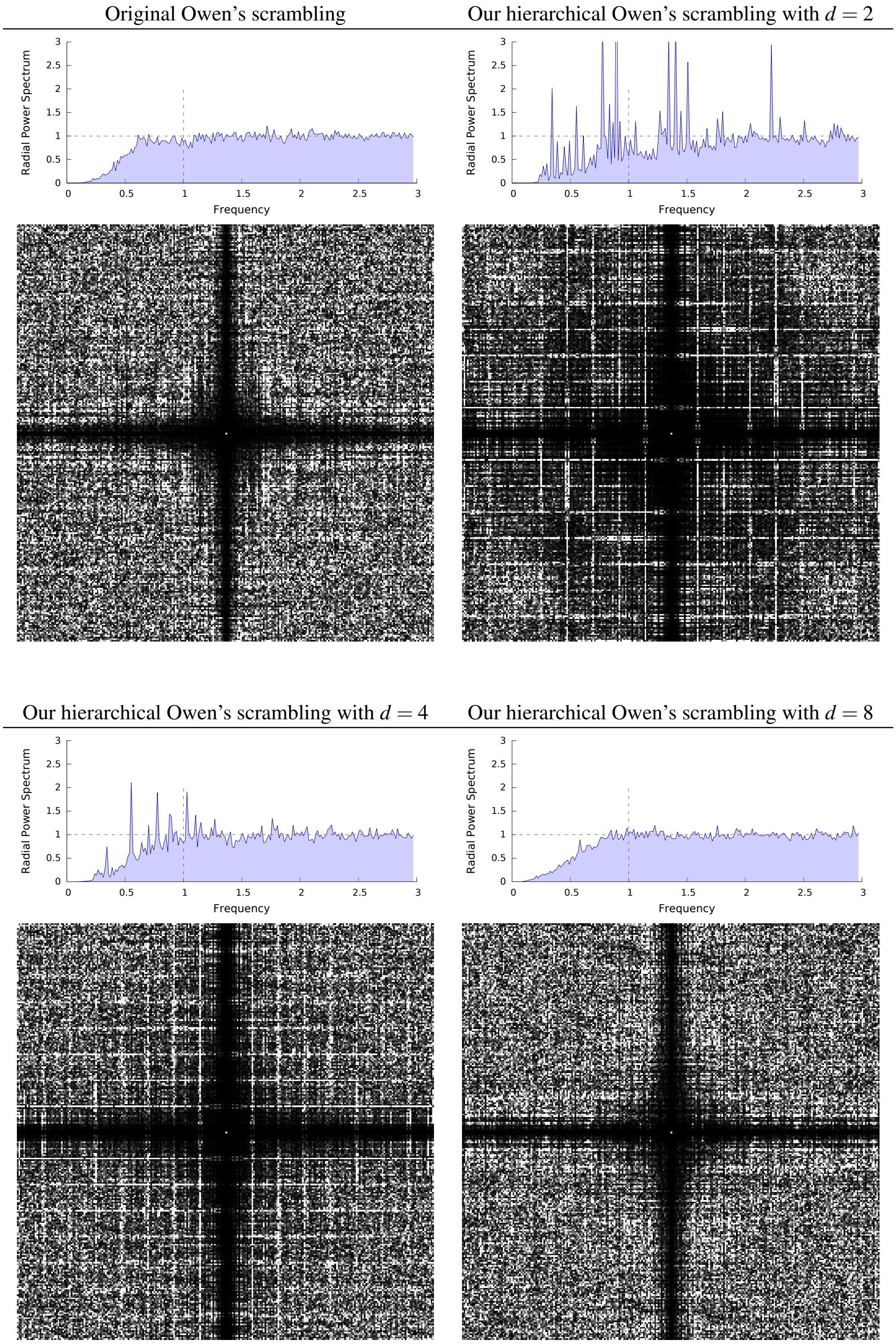


Figure 17: *Spectral comparison between Owen's scrambling, and our adaptation. We can see that for small d values, artefacts still appear but as d increases, there is no longer any visible difference between our set and the original Owen.*

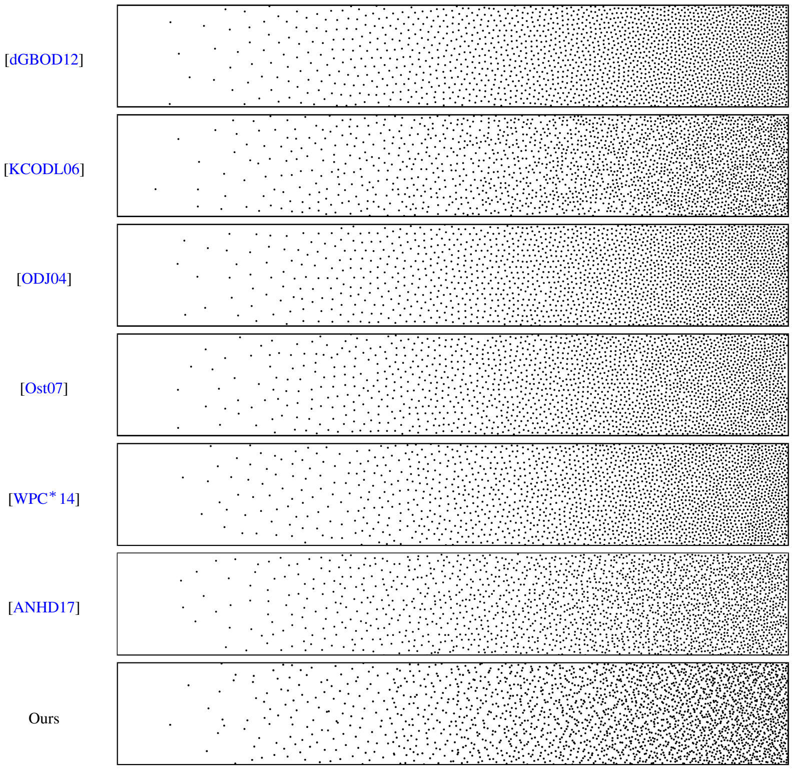


Figure 18: *Adaptive sampling of a quadratic density ramp with 2000 points for various sampling algorithms.*

Algorithm 1: $directAccess(i, LUT)$

---

input : An index i and the lookup table $LUT(\cdot)$.

output: The sample $\mathcal{P}^{\lambda}{}^{(i)}$.

$\mathbf{so}^{(i)} \leftarrow \mathcal{S}^{\lambda}{}^{(i)}$ $\mathbf{p}^{(i')} \leftarrow Referent(\mathcal{S}^{\lambda}{}^{(i)})$ $\mathcal{V}^{\lambda-1}{}^{(i)} \leftarrow \mathbf{p}^{(i')} \oplus directAccess(i', LUT)$ Retreive the tile $\mathbf{r}'$ from $\mathcal{V}^{\lambda-1}{}^{(i)}$ and $\mathbf{p}^{(i')}$ Retreive the set $\{\mathbf{q}\}_{\mathbf{r}'}^{\lambda}$ of the samples in the tile $\mathbf{r}'$ as **forall** *points* $\mathbf{so} \in \{\mathbf{so}\}_{\mathbf{r}}^{\lambda}$ **do**

  $\{\mathbf{q}\}_{\mathbf{r}'}^{\lambda} \leftarrow \{\mathbf{q}\}_{\mathbf{r}'}^{\lambda} \cup (\mathbf{so} \oplus \mathcal{V}^{\lambda-1}{}^{(i)}$

$\Pi^{\lambda}\mathbf{r}' \leftarrow LUT(\{\mathbf{q}\}_{\mathbf{r}'}^{\lambda})$ $\mathcal{P}^{\lambda}{}^{(i)} \leftarrow \Pi^{\lambda}\mathbf{r}'(\{\mathbf{q}\}_{\mathbf{r}'}^{\lambda}{}^{(i)}$ **return** $\mathcal{P}^{\lambda}{}^{(i)}$

[dGBOD12] DE GOES F., BREEDEN K., OSTROMOUKHOV V., DESBRUN M.: Blue noise through optimal transport. *ACM Trans. Graph. 31*, 6 (2012), 171:1–171:11. 14, 17, 18

[GRK12a] GRÜNSCHLOSS L., RAAB M., KELLER A.: Enumerating quasi-monte carlo point sequences in elementary intervals. In *Monte Carlo and Quasi-Monte Carlo Methods 2010*. Springer, 2012, pp. 399–408. 14, 15

[GRK12b] GRÜNSCHLOSS L., RAAB M., KELLER A.: *Monte Carlo and Quasi-Monte Carlo Methods 2010*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, ch. Enumerating Quasi-Monte Carlo Point Sequences in Elementary Intervals, pp. 399–408. 14

[KCODL06] KOPF J., COHEN-OR D., DEUSSEN O., LISCHINSKI D.: Recursive Wang tiles for real-time blue noise. *ACM Trans. Graph. 25*, 3 (2006), 509–518. 17

[Lem09] LEMIEUX C.: *Monte Carlo and Quasi Monte Carlo Sampling*. Springer-Verlag New York, 2009. 3, 7

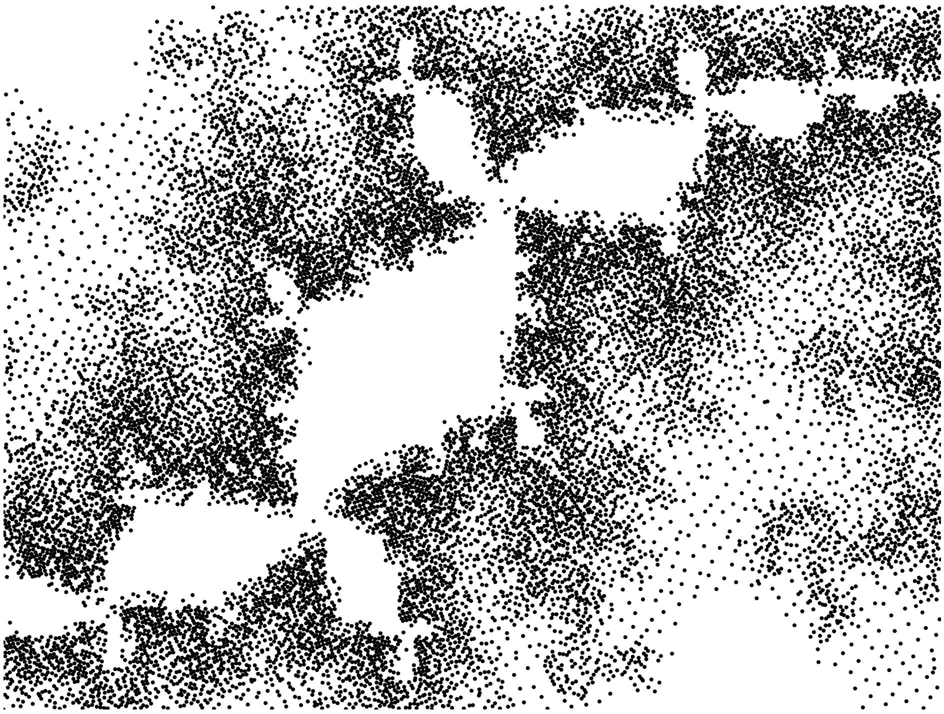
[Nie88] NIEDERREITER H.: Low-discrepancy and low-dispersion sequences. *Journal of Number Theory 30*, 1 (1988), 51 – 70. 3

[ODJ04] OSTROMOUKHOV V., DONOHUE C., JODOIN P.-M.: Fast hierarchical importance sampling with blue noise properties. *ACM Trans. Graph. 23*, 3 (2004), 488–495. 17

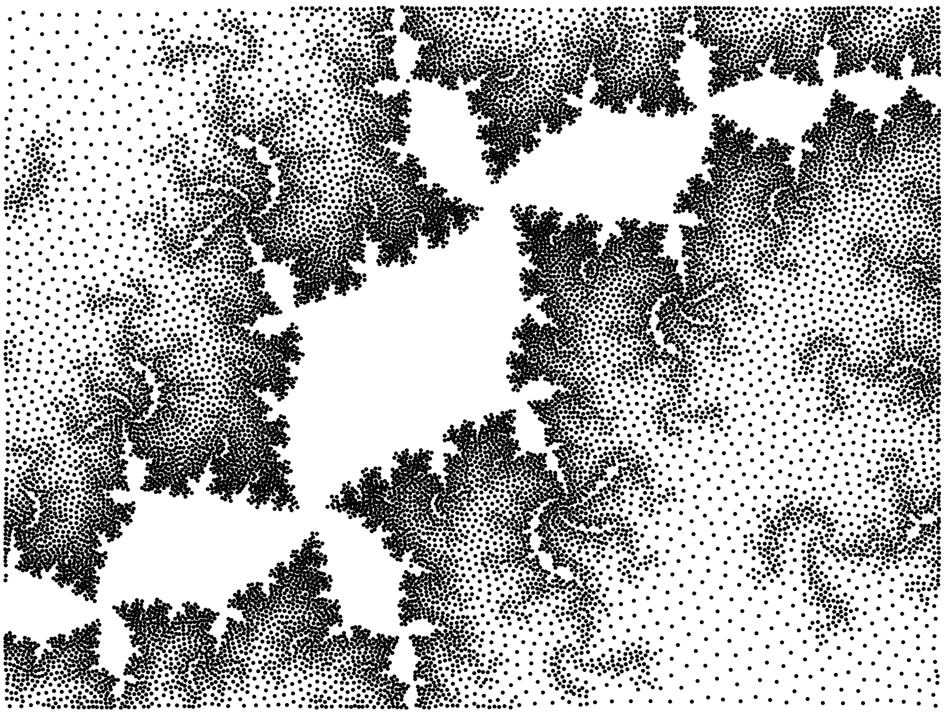
[Ost07] OSTROMOUKHOV V.: Sampling with polyominoes. *ACM Trans. Graph. 26*, 3 (2007), 78:1–78:6. 17

[Owe95] OWEN A. B.: *Randomly Permuted (t,m,s)-Nets and (t, s)-Sequences*. Springer New York, New York, NY, 1995, pp. 299–317. 13, 14

[Sob67] SOBOL' I. M.: On the distribution of points in a cube and the approximate evaluation of integrals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki 7*, 4 (1967), 784–802. 3, 5, 6, 13



Adaptive sampling obtained with our sampler



Adaptive sampling obtained with BNOT [dGBOD12]

Figure 19: *Julia set adaptively sampled using our method and BNOT with 16464 samples.*

[WPC*14] WACHTEL F., PILLEBOUE A., COEURJOLLY D., BREEDEN K., SINGH G., CATHELIN G., DE GOES F., DESBRUN M., OSTROMOUKHOV V.: Fast tile-based adaptive sampling with user-specified Fourier spectra. *ACM Trans. Graph. 33*, 4 (2014). 17