HAL
open science

Low-Discrepancy Blue Noise Sampling

Abdalla G.M. Ahmed, Hélène Perrier, David Coeurjolly, Victor Ostromoukhov, Jianwei Guo, Dong-Ming Yan, Hui Huang, Oliver Deussen

To cite this version:

Abdalla G.M. Ahmed, Hélène Perrier, David Coeurjolly, Victor Ostromoukhov, Jianwei Guo, et al.. Low-Discrepancy Blue Noise Sampling. ACM Transactions on Graphics, 2016, 35 (6), pp.247:1–247:13. 10.1145/2980179.2980218 . hal-01372542

HAL Id: hal-01372542

https://hal.science/hal-01372542v1

Submitted on 27 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Low-Discrepancy Blue Noise Sampling

Abdalla G. M. Ahmed*[1] Hélène Perrier [2] David Coeurjolly[2] Victor Ostromoukhov[2]
Jianwei Guo[3] Dong-Ming Yan[3] Hui Huang [4,5] Oliver Deussen[1,5]

[1]University of Konstanz [2]Université de Lyon
[3]NLPR, Institute of Automation, CAS [4]Shenzhen University [5]SIAT

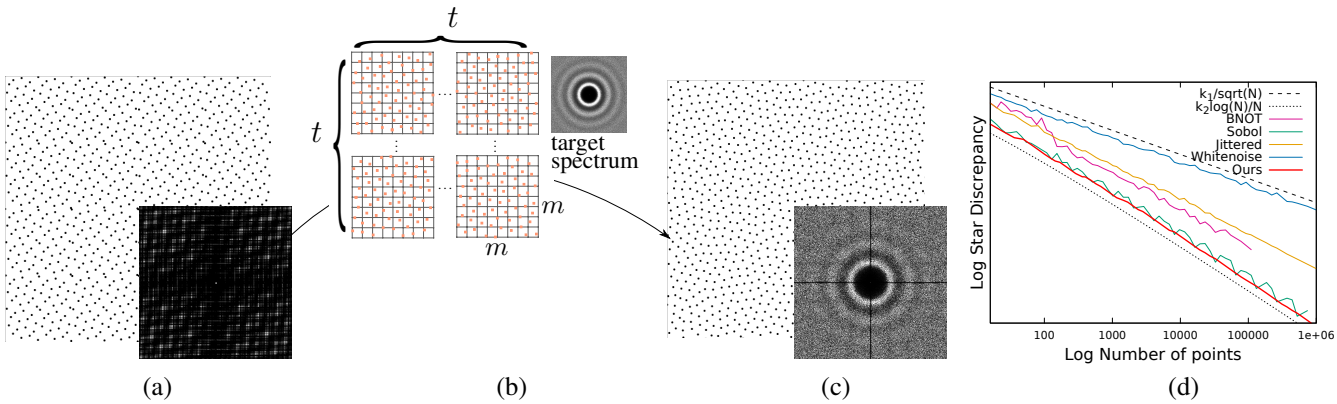


Figure 1: *Starting from (a) a template low-discrepancy (LD) point set we use (b) a segmented table of permutations to rearrange the LD set to match a reference set with the desired target spectrum. The permutations are localized and carefully constructed in such a way that they have minimal impact on the discrepancy of the underlying template set. (c) The resulting set inherits the spectral profile of the target set, while (d) still retains the discrepancy properties of the template set.*

Abstract

We present a novel technique that produces two-dimensional low-discrepancy (LD) blue noise point sets for sampling. Using one-dimensional binary van der Corput sequences, we construct two-dimensional LD point sets, and rearrange them to match a target spectral profile while preserving their low discrepancy. We store the rearrangement information in a compact lookup table that can be used to produce arbitrarily large point sets. We evaluate our technique and compare it to the state-of-the-art sampling approaches.

Keywords: Blue Noise, Low Discrepancy, Sampling, Monte Carlo, quasi-Monte Carlo

Concepts: •Computing methodologies → Rendering;

1 Introduction

Point sampling is a fundamental process in rendering applications in computer graphics [Pharr and Humphreys 2010]. For example, estimating the per pixel radiance requires the evaluation of complex integrals, and (quasi-)Monte Carlo methods are widely used to numerically estimate these integrals from samples.

*e-mail:abdalla_gafar@hotmail.com

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.
SA '16 Technical Papers,, December 05 - 08, 2016, , Macao
ISBN: 978-1-4503-4514-9/16/12
DOI: http://dx.doi.org/10.1145/2980179.2980218

The choice of a sampling pattern plays a crucial role to reach a prescribed rendering quality. Unfortunately, it is still not clear what is the ideal pattern for a given number of samples. At low sampling rates, some scholars argue in favor of blue noise patterns [Ulichney 1988], noting that these patterns are observed in the photoreceptors of animals [Yellot1983], and are therefore desirable from the visual perception perspective. Others argue in favor of low discrepancy (LD) sequences and point sets that are commonly used for sampling in various contexts, noting their computational efficiency [Shirley 1991; Kollig and Keller 2002]. These different perspectives lead to two different and relatively disjoint trends of research. In this paper we try to reconcile these two perspectives by developing a sampling pattern that simultaneously carries the blue noise and LD properties.

1.1 Motivation

LD sequences and point sets play a crucial role in quasi-Monte Carlo (QMC) integration — an active research field during the last few decades [Niederreiter 1992; Lemieux 2009; Keller 2012]. In computer graphics, (quasi-)Monte Carlo integration is extensively used to evaluate light transport in complex scenes that incorporate sophisticated geometry and/or objects with complex reflection/refraction/absorption properties, as well as having multiple light sources [Pharr and Humphreys 2010; Ramamoorthi et al. 2012]

The discrepancy of a point set is a quantitative measure of deviation from the uniform distribution. The extreme discrepancy of a point set in a unit domain is defined as the largest difference, over all axes-aligned rectangles, between the area of a rectangle and the proportion of the point set elements it contains. The star discrepancy is defined similarly, but one corner of the rectangles is fixed at the origin. These measures allow for bounding the numerical integration error; see the Koksma-Hlawka inequality [Niederreiter 1992]. A 2D set of N points is considered an LD set if its discrepancy is

$\mathcal{O}(\log(N)/N)$. A 2D LD sequence is a sequence that maintains $\mathcal{O}(\log^2(N)/N)$ discrepancy for all its sub-sequences. The concept of uniform, low-discrepancy sequences emerged in the 1930s, after the pioneering work of van der Corput [1935]. Many different LD constructions were proposed thereafter [Niederreiter 1992; Lemieux 2009].

Despite the wide adoption of LD sets and sequences in computer graphics, there are considerable differences compared to their application in, say, financial simulations. First, the final judgement on image quality of a QMC integration is performed on a *two-dimensional* projection plane, whereas most existing LD sequences were developed to improve the uniformity in very high dimensions. Secondly, the rendering results are evaluated *visually*, and the human eye is particularly sensitive to aliasing artifacts, which is much less important in financial simulations. Most LD sequences, such as Sobol, Faure or Halton sequences — without scrambling (see Section 2.6) — contain strong spectral peaks, and may generate strong aliasing artifacts when used in rendering applications.

The main goal of this work is to build a provably LD point set that bears a blue-noise spectrum. On the one hand, the variance in integration is directly dependent on the product of the power spectra of the integrand and the sampling pattern [Pilleboue et al. 2015]. Consequently, for integrands whose spectral content can be estimated, one can considerably reduce variance by choosing the parameters of the blue-noise sampling distribution in such a way that its vanished low-frequency spectral part contains the major part of the spectral content of the integrand. On the other hand, according to the Koksma-Hlawka inequality, the error in integration is bounded by the product of the integrand's variation, in the sense of Hardy and Krause, and the star discrepancy of the sampling distribution [Niederreiter 1992]. Thus, for bounded-variation integrands whose spectral content cannot be easily estimated, the LD property will still guarantee low variance, at least asymptotically. Finally, the absence of spectral peaks will avoid aliasing [Durand 2011; Pilleboue et al. 2015].

The realization of our goal consists in morphing a special stratified LD point set into a close approximation of a given stratified reference point set; thus adopting the spectral properties of the reference set, while retaining its LD properties. We achieve this by locally rearranging the coordinates first in the horizontal and then the vertical direction. We prove that such a rearrangement maintains LD properties, and we provide evidence that we achieve close approximations of the reference point set even with small LD sets. Our construction is conceptually very simple and computationally efficient. It minimizes aliasing and introduces only low levels of noise. Furthermore, it is easy to implement, and we provide code in a supplementary material.

Although we focus on blue-noise properties and very low spectral peaks, our method can be applied to any given stratified point set. An extension to adaptive LD sampling is not yet given. However, as pointed out by Pharr and Humphreys [2010], adaptive sampling in many practical applications is avoided because of the overhead it induces. If the sampler is fast enough, it is more efficient, for many scenes, to use global supersampling. Our method is compliant with this approach thanks to its computational efficiency.

2 Related Work

Our work builds upon many existing approaches for sampling in computer graphics, and combines many ideas. We briefly review the main categories of related work.

2.1 Stochastic Sampling

The computer graphics community became sensitive to the problem of sampling thanks to the pioneering works of Dippé and Wold [1985], Cook [1986], Mitchell [1991], and Shirley [1991]. They documented how aliasing artifacts arrise in regular sampling, and proposed stochastic sampling as an alternative, arguing that unstructured noise is better visually perceived than aliasing or Moiré effects. The dart-throwing and other Poisson disk algorithms became popular since then, and a lot of research was devoted to improving the computational efficiency and the sampling quality of the algorithms. To name a few, McCool and Fiume [1992] proposed a hierarchical version of dart throwing, and introduced Lloyd's relaxation [Lloyd 1982] as a post-process, Dunbar and Humphreys [2006] proposed a linear-time acceleration, Yuksel [2015] proposed an algorithm based on sample elimination, while Ebeida et al. [2011] focused on maximal Poisson disk algorithms.

Recently, a number of new optimization-based algorithms have been proposed for blue noise [Balzer et al. 2009; Chen et al. 2012; Schmaltz et al. 2010; Schlömer et al. 2011; Fattal 2011; de Goes et al. 2012; Jiang et al. 2015], as well as general noise [Zhou et al. 2012; Öztireli and Gross 2012; Heck et al. 2013]. Since our method relies on an offline optimization (Section 4), many of these algorithms can be used.

2.2 Stratified Sampling

Stratified sampling patterns are obtained by subdividing the sampled domain into a regular lattice of strata (cells), and placing one or more sample point on each stratum. As an efficient alternative to Poisson disc distributions, Dippé and Wold [1985] proposed a jittered grid by placing the samples randomly on each stratum. This approach became popular for its computational efficiency and decent anti-aliasing qualities. Shirely et al. [1994] proposed a multi-jittered grid to improve the uniformity of a jittered grid, and Kensler [2013] proposed correlated multi-jittering to improve the spacing between the points. Our method is also stratified.

2.3 Latin Hybercube Sampling

Another approach to generating a uniform distribution of samples is the Latin hybercube, also known as N-Rook sampling [Shirley 1991]. The idea is that the projections of the samples on each axis are uniformly distributed. Saka et al. [2007] studied *Latinization* as a post-processing of an optimized set, whereas Reinert et al. [2016] integrated the idea within the optimization process. Our technique also produces uniform distributions of projection, but with provable LD.

2.4 Look-up Methods

The generation of high-quality blue noise sets is computationally expensive, and could be prohibitive for high sampling rates [Lagae and Dutré 2008]. Many lookup methods were therefore developed for distributing pre-computed blue noise sets. These methods store the offsets of the points relative to a stochastic regular-lattice tiling [Cohen et al. 2003; Kopf et al. 2006; Lagae and Dutré 2006], a complex recursive tiling [Ostromoukhov et al. 2004; Ostromoukhov 2007; Wachtel et al. 2014], or AA Patterns [Ahmed et al. 2015]. Our method is also look-up based, but we store the offsets of the points relative to an LD point set. The AA Patterns method was an important inspiration to us, but here we solve a different problem.

2.5 Low-Discrepancy Sets and Sequences

LD sets (or sequences) use deterministic mathematical formulas to calculate the coordinates of the sample points directly from the ordinal sequence numbers of the samples in the set (or sequence). They are provably more uniformly distributed than random numbers can be [Kuipers and Niederreiter 1974]. In fact, most LD sequences exhibit both stratification (Section 2.2) and Latin Hypercube properties (Section 2.3). Shirley was among the pioneers and the best advocates for the use of discrepancy as a quality measure of computer graphics samplers [Shirley 1991]. Keller and his collaborators further promoted the use of LD sequences in computer graphics [Keller 2012]. Even though there are many known constructions for two-dimensional LD sets, we develop our own construction, and prove its discrepancy bounds from the first principles. The inherent 2D indexing of our construction is crucial for our optimization.

2.6 Scrambled LD Sequences

Random scrambling techniques [Tezuka 1994; Owen 1995; Matoušek 1998; Owen 2003] were introduced to improve uniformity, and to allow for error estimation, resulting in so-called randomized quasi-Monte Carlo techniques. Keller and Kollig [2002] use random scrambling to pad low dimensional LD sequences for sampling in higher dimensions. However, these techniques do not enable spectral control; rather, the typical resulting power spectrum resembles a jittered set; see Figure 7 on page 8. Our LD blue noise construction is based on scrambling as well, but in a more general way (Section 3.1), as it does not necessarily preserve the structure of the elementary intervals [Niederreiter 1992].

3 Optimized Low-Discrepancy Sets

In order to build an LD blue noise set, we start with a *template* low discrepancy set, and optimize it to acquire the desired blue noise profile (BNOT [de Goes et al. 2012], Step [Heck et al. 2013], etc). There are many LD constructions to choose from (Sobol, Halton, Hammersley, rank-1, etc.), but they all suffer from one fundamental problem when it comes to optimization: the topology of these sets changes with the size of the set, making it difficult to identify the neighborhoods of the points; whereas, on the other hand, the spectral properties are sensitive to the local neighborhoods of the points [Wei and Wang 2011]. Therefore, we devise a new LD construction that suits our needs, based on stratification.

Suppose that we want to generate $N := n^2$ samples. A stratified point set can be written as

$$P := \Big\{ (X + u_{X,Y}, Y + v_{X,Y}) : X, Y \in \{0 \dots n-1\}; \quad u_{X,Y}, v_{X,Y} \in [0,1) \Big\} . \quad (1)$$

The ordered pair (X, Y) selects a specific stratum, and an associated ordered pair (u, v) specifies the location of the sample point in that stratum. Note, that u and v are not necessarily functions of X and Y; we only indicate here that each (u, v) pair is associated with a specific stratum (X, Y). To sample the unit domain, $[0, 1)^2$, one can simply scale the point set by $1/n$.

The essence of stratification is that the discrepancy is zero for all the strata that are completely enclosed by the considered rectangle; hence, evaluating the star discrepancy of a stratified point set reduces to evaluating the discrepancy along a column and a row. As we show in Appendix A, we obtain a low-discrepancy 2D stratified point set if for every column, X, the sequence $\{u_{X,0}, u_{X,1}, \dots, u_{X,n-1}\}$ is an LD sequence, and for every row, Y, the sequence $\{v_{0,Y}, v_{1,Y}, \dots, v_{n-1,Y}\}$ is an LD sequence. To transform this condition into a construction, we pick up a 1D low-discrepancy sequence, S, and use it to supply the horizontal offsets along the columns:

$$u_{X,Y} := S_Y ,$$

and the vertical offsets along the rows:

$$v_{X,Y} := S_X .$$

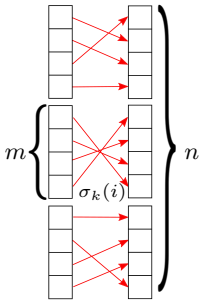
The resulting LD point set is then defined by:

$$P_S := \Big\{ (X + S_Y, Y + S_X) : X, Y \in \{0 \dots n-1\} \Big\} . \quad (2)$$

Unlike existing LD constructions, ours is inherently indexed in two dimensions, using a pair of indices. This difference is important for optimization, because the local neighborhood is fixed around any given index (X, Y). Our new construction allows to align the LD set with any stratified set, on a one-to-one basis. As we will discuss in Section 3.2, this is the key element to control the spectral properties of the set.

The set P_S will serve as our template LD set, and we will subsequently discuss how to enhance its spectral properties.

3.1 Randomized Permutation



A random element can be incorporated into our construction by applying a discrepancy-preserving rearrangement of the sequence S for each row and column. Unlike the global bit-wise scrambling techniques used with common LD constructions (Section 2.6), we randomly permute the entries of the constituent sequences (in rows and columns) locally within small chunks. We choose a chunk size, m, and re-order every subsequent m entries of the input sequence S (we assume that m divides n):

$$S'_{k \cdot m + i} = S_{k \cdot m + \sigma_k(i)} ; \quad k \in \{0 \dots n/m\} ; \; i \in \{0 \dots m-1\} , \quad (3)$$

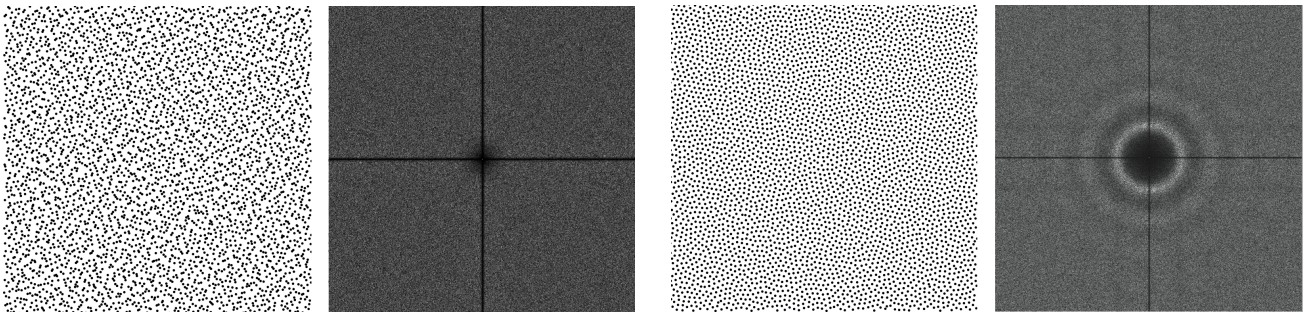
where each σ_k is a permutation over $\{0 ... m-1\}$. To incorporate this 1D chunk-wise permutation into P_S, we apply two passes; first along the columns, then along the rows; see Algorithm 1.

What makes this chunk-wise permutation have little impact on the discrepancy is that any subset of S' differs from the corresponding subset of S in at most m points. According to [Kuipers and Niederreiter 1974, Chapter 2, Theorem 4.1][1], the difference in discrepancy of any subsets of length $n > m$ of the two sequences is then bounded by m/n. Since m is fixed, the difference,

$$|D^*_n(S') - D^*_n(S)|$$

in discrepancy between the two sequences, is $\mathcal{O}(1/n)$. Consequently, if S is an LD sequence, with $\mathcal{O}(\log(n)/n)$ discrepancy bound, then the shuffled sequence S' is also an LD sequence. The parameter m only alters the constant in the $\mathcal{O}(\cdot)$ notation, but we want to keep m small so that we stay close to the discrepancy of S.

[1] The statement of the mentioned theorem compares corresponding entries in two "finite sequences". But since we are referring to subsets, these can be arranged in any order without altering their discrepancy. To give an example, the sets $\{0, 0.25, 0.5, 0.75\}$ and $\{0, 0.5, 0.25, 0.75\}$ have different discrepancies as sequences, but the same discrepancy as fixed sets. Alternatively, it is easy to show from the first principles that the difference in discrepancy between S' and S is bounded by m/n, since the point-count of the two sequences in any intervals differs by no more than m points.



(a) (b)

Figure 2: *Spatial and spectral plots of a point set (4096 points) obtained from (a) Algorithm 1 ($m = 8$), and (b) Algorithm 2 (BNOT target, $m = 16$, $t = 64$).*

Algorithm 1: Randomized permutation of P_S.

for *each column* X **do**
  for *each chunk* k **do**
    Randomly permute the vector $\{u_{X,Y}\}_{Y=k\cdot m}^{k\cdot m+m-1}$;
    Write the permuted vector back into the set;
  end
end
Repeat with the rows by exchanging the roles of u, X and v, Y, respectively;

Discussion: At this point, the resulting point set of Algorithm 1 (Figure 2a) closely resembles a scrambled LD set like Owen's [Owen 1995] (Figure 7 on page 8), which is the current state-of-the-art. Our localized permutation, however, admits much more control over the placement of individual points than Owen's axis-wise scrambling, and it enables real optimization, as we discuss subsequently.

3.2 Optimized Permutation

Instead of considering random permutations, we can construct the permutations such that the resulting point set, $P_{S'}$, is close to a given reference point set. Consider a stratified $t \times t$ point set, optimized to a specific blue noise profile (BNOT, Step, etc; see Section 4). We also assume that t is a multiple of m, so that each row and column of the point set can be decomposed into chunks of size m. The main idea is to start with a $t \times t$ slice of the P_S template set, and find the permutations, for all 1D chunks, that would transform this point set so that it approximates the reference set. The permutations are then stored in a look-up table that can be used to reconstruct the approximate set. For a given point, (X_t, Y_t), in the $t \times t$ domain, the look-up table entry, $LUT(X_t, Y_t)$, contains the offset, $(X_m, Y_m) \in m \times m$, of its new sequence numbers within the $m \times m$ block.

Technically, we minimize the distance between the two sets in two passes: first along the horizontal axis, then along the vertical axis; see Algorithm 2. Note that the two passes are completely independent, and can therefore be parallelized. Our algorithm solves a discrete assignment problem similarly to the Hungarian algorithm [Kuhn 1955]. Figure 3 illustrates the process for a single chunk, and Figure 4 illustrates it inside a single $m \times m$ block. We apply a similar rearrangement to each $m \times m$ block in the point-set. Within each block, the process is very similar to Latinization, but instead of snapping the points to a regular grid, the coordinates are taken from the underlying LD sequences. Thanks to this Latinization, the discrepancy is improved even within the blocks, cf. [Kensler 2013]. Figure 2b shows an example set generated by Algorithm 2.

Algorithm 2: Generating an optimized LD set from a reference stratified point set and our template LD point set.

Initialize the output set to P_S from Eq. (2);
for *each column,* X, **do**
  for *each chunk,* k, **do**
    Retrieve the vector $\mathbf{L_x} = \{u_{X,Y}\}_{Y=k\cdot m}^{k\cdot m+m-1}$ from the reference set;
    Retrieve the corresponding vector, $\mathbf{Q_x}$, from the output set;
    Rearrange $\mathbf{Q_x}$ so that the smallest entry aligns with the smallest entry in $\mathbf{L_x}$, the second smallest with the second smallest, and so on;
    Write the permuted vector, $\mathbf{Q_x}$, back to the output set;
  end
end
Repeat with the rows by exchanging the roles of u, X and v, Y, respectively;

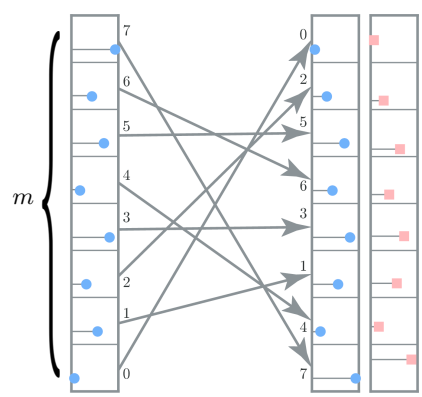


Figure 3: *A visual illustration of step 6 in Algorithm 2, using $m = 8$. For each chunk of size m from the template LD set (blue dots) and the reference set (orange squares), we rearrange the points of the template set so that the smallest u aligns with the smallest u of the reference, the second smallest with the second smallest, and so on. The new sequence of indices, $\{7, 4, 1, 3, 6, 5, 2, 0\}$, is appropriately encoded and stored in the look up table. Note that v is not relevant at this stage.*

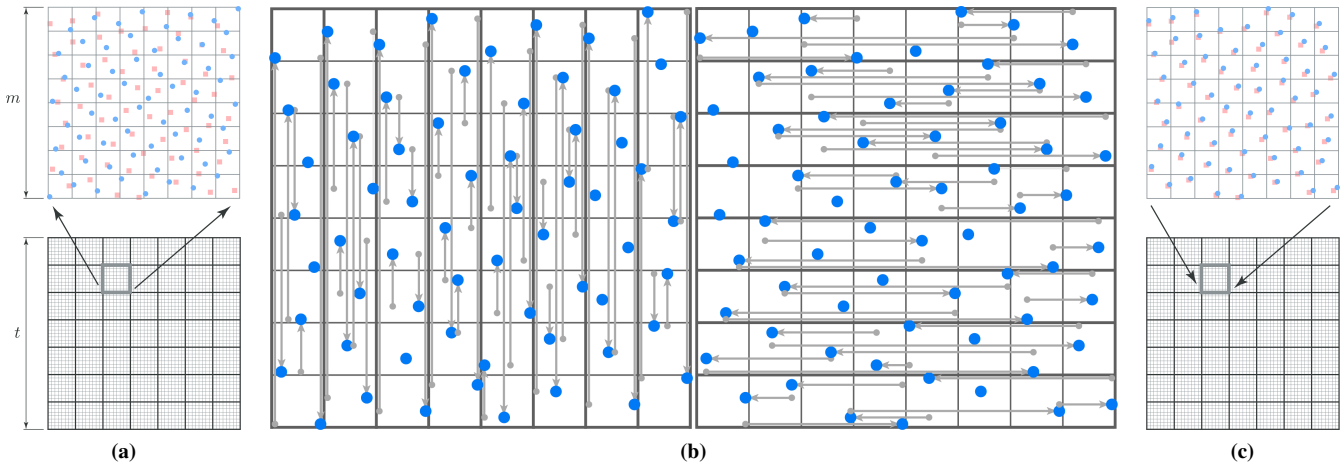


Figure 4: *Reordering the template LD set (blue circles) to match a given reference (orange squares), without loosing its LD property. (a) An $m \times m$ block is extracted from the LD set, along with the corresponding block from the reference. (b) The horizontal offsets are rearranged along the columns first, as shown in Figure 3 for one column, and then the vertical offsets along the rows. (c) The final block — now closely matching the reference — is written back to the set.*

The resulting point set of Algorithm 2 inherits the LD property from the template set, P_S, and the spectral profile from the reference set. The parameter m controls the blending ratio between the two. Setting $m = 1$ reproduces P_S, whereas $m = t$ produces a merely Latinized copy of the reference set, with limited improvement of its discrepancy. From our experiments, $m = 16$ is a good compromise. The impact of m on the discrepancy was discussed earlier in Section 3.1. The effect of m on how the rearranged LD set comes close to the reference is discussed in Appendix B.

The size of the reference point set, t^2, is a trade-off between the memory footprint of the look-up table, and the quality of the spectral profile (compared to the reference set) when the size of the final point set, N, is larger than t^2; see Section 3.4. In our experiments, we have considered $t = 128$; evidence for this is given in the supplementary materials, along with a complete evaluation of both the t and m parameters.

3.3 Choosing a Low Discrepancy Set

Any one-dimensional LD sequence can be used to construct the template LD point set in Eq. (2), but the inherent binary nature of the van der Corput sequence [van der Corput 1935] makes it an excellent candidate for an efficient implementation. This sequence is obtained by mirroring the binary digits of each sequence number around the binary point. That is, if ϕ is a binary bit-reversal function:

$$\phi\left(\sum_{j=0}^{\infty} a_j 2^j\right) = \sum_{j=0}^{\infty} a_j 2^{-j-1}\,;\; a_j \in \{0,1\}\,, \tag{4}$$

then the ith entry in the sequence is $\phi(i)$. If the chunk size, m, is a power of 2, then permutation reduces to permuting the $\log_2(m)$ least significant bits of the sequence number. We denote by P_{vdC} the 2D point set obtained from this 1D sequence:

$$P_{\mathrm{vdC}} := \Big\{\, (X + \phi(Y), Y + \phi(X)) \,\Big\}\,. \tag{5}$$

As shown in Appendix A, P_{vdC} is an LD point set, with discrepancy bound in $\mathcal{O}\left(\log(N)/N\right)$. It is worth noting that P_{vdC} coincides with the Hammersley construction when the generated number of points is a power of two, and is different otherwise.

3.4 Indexing and the Overall Sampler Construction

Our optimized permutation described in Section 3.2 is an offline process to construct the $t \times t$ look-up table. For a desired number of samples N ($N > t^2$), we need to retrieve the offsets (u, v) associated with the strata (X, Y). One solution could be to replicate the $t \times t$ permuted point set [Dippé and Wold 1985; Pharr and Humphreys 2010], assuming that the set can be tiled seamlessly. However, replication would break the low discrepancy property, since several points would project to the same 1D point on either of the two axes. To overcome this issue, we define an indexing scheme tightly related to the underlying 1D LD sequence and the bit-reversal principle.

For a given abscissa X, we define $X_t := X \bmod t$, its position in a $t \times t$ domain, and $X_m := X \bmod m$, its position in a chunk of size m within the $t \times t$ domain (we assume similar definitions for Y_t and Y_m). The final offsets are given by:

$$u := \phi(Y - Y_m + L_Y) \tag{6}$$

$$v := \phi(X - X_m + L_X)\,, \tag{7}$$

where $(L_X, L_Y) := LUT(X_t, Y_t)$.

Eq. (6) and (7) perform both the initial offset construction, using the van der Corput sequence, and the optimized permutation within the chunks of size m. More precisely, for a given X and Y, $\phi(X)$ is the vertical offset in the template LD point set (Eq. (2)). The optimized permutation is only performed within the size-m vertical and horizontal chunks the point (X, Y) belongs to. In a bitwise representation of X, the permutation only occurs in the least significant $\log_2(m)$ bits, as illustrated in Figure 5.

Using previous notations, we define our final Low-Discrepancy Blue Noise sampler as follows:

$$\mathrm{LDBN} := \Big\{ \big(X + \phi(Y - Y_m + L_Y),\; Y + \phi(X - X_m + L_Y)\big) \Big\}\,. \tag{8}$$

From Appendix A, LDBN is an LD set, and we demonstrate in Sec-

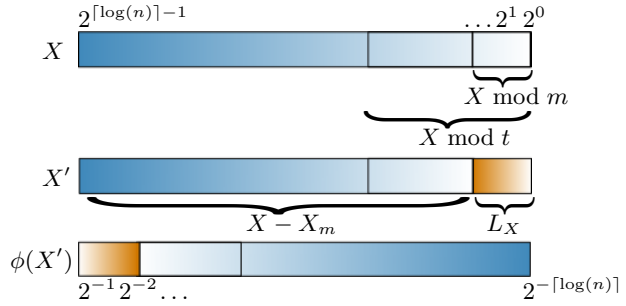


Figure 5: *Bitwise representation of Eq. (7) with $X' := X - X_m + L_X$. Note that the permutation in orange depends on the $X \bmod t$ range (bit reversal is illustrated with color gradient).*

tion 5 that when optimizing the look-up table with a reference point set with specific spectral properties (e.g, Step blue noise), LDBN preserves these properties.

On the practical side, LDBN is extremely computationally efficient, since it can be implemented using only bit manipulations and memory lookups. It is also considerably efficient in storage, since only integer permutations in $[0, m)$ have to be stored, rather than the actual real-valued (u, v) offsets. If $m = 16$, then the storage space reduces to only one byte per entry.

4 Producing the Input reference Sets

The only input we require to build an LDBN sampler is a stratified set optimized for the desired spectral profile. We now discuss the practical details of how to obtain such sets.

In order to create a stratified blue noise set, we start with a jittered grid, then apply a blue noise optimizer while ensuring that the displacements of the points are confined to the respective strata. The actual realization of these steps, however, is not as trivial as it may sound.

The first obstacle is that a regular grid is a stable minimum for the associated energy of many optimizers (Lloyd's algorithm [McCool and Fiume 1992] and kins [Balzer et al. 2009; de Goes et al. 2012; Chen et al. 2012; Xu et al. 2011]); hence, such optimizers tend to restore the grid structure of the jittered grid. Rather than employing direct optimization algorithms, we overcome this problem by employing a target-matching algorithm, such as Heck et al. [2013]. These algorithms consider much wider neighborhoods compared to Lloyd's variants, and are therefore less likely to form local regular structures. Once the point set is sufficiently close to the target profile, it becomes safe to use the original optimizer without risking convergence towards a grid structure. Alternatively, the target-matching algorithm can be resumed until convergence. This becomes the only available choice for producing noise profiles that are only defined analytically, such as Step blue noise [Heck et al. 2013].

The second obstacle concerns the treatment of points that escape their strata during optimization. We experimented with different policies, such as clamping the shifts suggested by the optimizer to the strata, moving the escaped points to a random location, or just skipping these point in the current iteration. We prefer the latter option, since it imposes less bias (over- or under-crowding) near the edges of the strata, and is also easier to implement. We also occasionally Latinized the point-set during the optimization process in the spirit of Reinert et al. [2016]. This process almost circumvents the bias to the edges.

Depending on the optimizer and the targeted quality, the optimization process may take a few cycles of jittering, optimizing, Latinizing, and evaluation. It requires certain skills, and may take hours. Fortunately, the end users would not need to implement the optimization by themselves. In our supplementary material, we provide the look-up tables for the BNOT and the Step blue noise profiles.

5 Results

In this section, we perform a series of experiments to demonstrate the low discrepancy and spectral properties of our sampler, leading to good results in quasi-Monte Carlo integration. First, we recall that integration error and low discrepancy of the point set are related thanks to the Koksma-Hlawka's [Hlawka 1961] inequality. In Section 5.1, we present a comparison between our sampler and state-of-the-art samplers with respect to star discrepancy. In Section 5.2, we compare spectral contents in addition to zoneplate experiments widely used to evaluate the anti-aliasing performances of the point sets. Finally, in Section 5.3, we present some integration results for classical integrand functions.

In all our experiments, we compare our sampler with samplers of different types, including LD samplers like Sobol [Sobol' 1967], blue noise samplers like BNOT [de Goes et al. 2012] and Step [Heck et al. 2013], stratified samplers like jittered grid, and rank-1 lattices with Fibonnacci sequences [Keller 2004]. In our supplementary material, we provide a more exhaustive analysis including CVT [Lloyd 1982], Wang tiles [Kopf et al. 2006], FPO [Schlömer et al. 2011], CapCVT [Chen et al. 2012], Polyhexes [Wachtel et al. 2014], AA patterns [Ahmed et al. 2015], Poisson disk, Halton [Halton 1960], Hammersley [Hammersley 1960], Faure [Faure 1982], Niederreiter [Niederreiter 1988], N-Rooks [Shirley 1991], rank-1 [Keller 2004; Dammertz 2009], Correlated Multi Jittered sampling [Kensler 2013], Owen's scrambling over Sobol [Owen 1995; Owen 2003; Kollig and Keller 2002], regular and hexagonal grid, jittered grid, and white noise.

All our tests were performed using $m = 16$ and $t = 128$; the influence of these parameters on the result is discussed in our supplementary material.

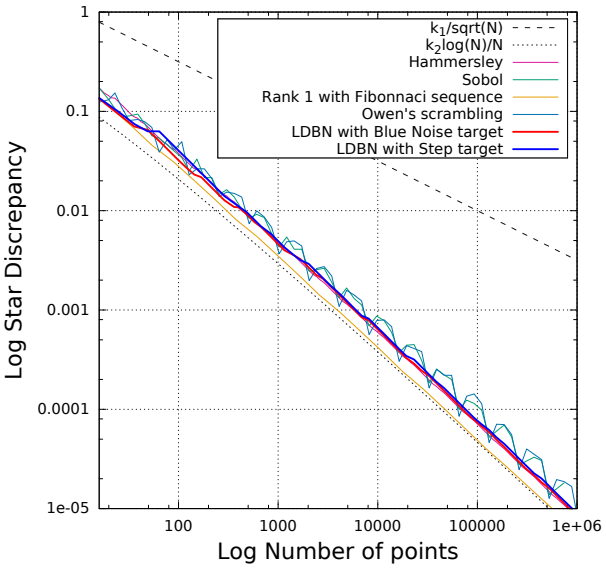
5.1 Discrepancy

In Figure 6, we present the evolution of star discrepancy as the number of points increases for our sampler and various others. As expected, our sampler is a low discrepancy point set and thus behaves like LD samplers in Figure 6a.

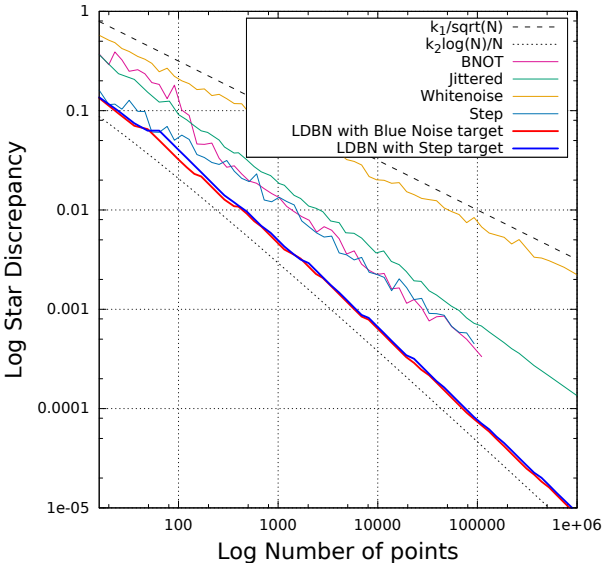
In Figure 6b, we also compare our sampler with stochastic samplers targeting particular spectral content such as BNOT [de Goes et al. 2012] or Step [Heck et al. 2013]. Since these samplers do not optimize specifically the discrepancy, LD samplers outperform them in this discrepancy test. Note that BNOT and Step optimizer are time consuming compared to LD samplers or to our sampler, which limits the size of the point sets considered in this experiment. Furthermore, the star-discrepancy measurements have been obtained using an exact $\mathcal{O}(N^2)$ algorithm.

5.2 Spectral and Anti-Aliasing Analysis

In Figure 7, we detail spectral content of various samplers and zoneplate aliasing tests. This test reconstructs the function $\sin(x^2 + y^2)$ using a single sample per pixel and a Mitchell reconstruction filter



(a) *Discrepancies of LD Samplers*



(b) *Discrepancies of other samplers*

Figure 6: *Discrepancy comparison between our sampler and various others. Samplers such as Step or BNOT require tens of hours to compute a point set of size* 10^5*, since this graph requires between* 30 *to* 100 *point sets for each sampler with sizes ranging from* 16 *to* 10^6*, it is impracticable to generate it up to* 10^6 *if a single point set requires days to compute. Samplers such as Low Discrepancy ones on the other hand are much faster to compute and we could therefore analyze their behaviors up to* 10^6 *points (*k_1 *and* k_2 *are constants used to scale analytic curves).*

[Mitchell and Netravali 1988]. Since a whole range of frequencies is present, zoneplate reconstructed images are classical tools to evaluate the aliasing defects of sampling patterns.

First, you may note that apart from the black cross (resulting from our low discrepancy property), the fourier spectra of our sampler (BNOT and Step targets) match with, respectively, BNOT and Step spectra, illustrating its versatility.

Blue Noise Samplers	1k	10k	100k	1M
[de Goes et al. 2012]	10s	6m 5s	6h 27m	-
[Heck et al. 2013]	6s	7m 45s	17h 5m	-
Dunbar et al. [2006]	0.000s	0.015s	0.131s	1.33s
[Wachtel et al. 2014]	0.003s	0.032s	0.23s	2.43s
Ours	0.002ms	0.023ms	0.22ms	2.14ms

Other Samplers	1k	10k	100k	1M
[Sobol' 1967]	0.066ms	0.669ms	6.88ms	0.067s
[Keller 2004]	0.067ms	0.38ms	2.72ms	0.027s
Jittered	0.04ms	0.403ms	3.90ms	0.041s
Ours	0.002ms	0.023ms	0.22ms	2.14ms

Table 1: *Timings (on a bi-Intel Xeon E5-2650 2.60GHz) to generate different number of points with various method. BNOT [de Goes et al. 2012], Fast Poisson [Dunbar and Humphreys 2006], Polyhexes [Wachtel et al. 2014] and Step [Heck et al. 2013] implementations correspond to source code provided by the authors. Note that for our sampler and for [Wachtel et al. 2014], we did not include the timings for reading the LUT table since it needs to be read only once to generate multiple pointsets.*

Secondly, the zoneplate images show that the best reconstruction is obtained using blue noise samplers, such as BNOT or ours. However, whereas BNOT requires several hours to generate a $100k$ pointset (Table 1), our sampler requires less than a millisecond. We also compare timings with optimized fast approximation techniques of Poisson disk sampling [Dunbar and Humphreys 2006]. However, spectral profiles of Poisson disk point patterns lead to slower variance convergence rates ([Pilleboue et al. 2015] and Figure 8 and 9). The zoneplates also show how state-of-the-art LD samplers, such as Sobol or rank-1, perform poorly for this reconstruction, due to the spectral peaks in their Fourier spectra. Note that Owen's scrambling over a Sobol sequence is still an LD sequence, but does not exhibit significant peaks in the Fourier spectra; hence, very little aliasing is observed in the reconstruction. Nevertheless, its zoneplate is similar to a jittered grid, with only a narrow low-noise band.

5.3 Quasi-Monte Carlo Integration

In this work, point sets are used in a Monte Carlo context where we want to evaluate a complex integral by sampling and summing up integrand values. To predict the integration error with respect to the number of samples, the discrepancy is a first critical value, as discussed earlier. The shape of the point set power spectra can also be explicitly related to variance or integration error [Pilleboue et al. 2015].

We evaluate integration variance by integrating a scene with several realizations of a point set (1000 realizations for a fast sampler and between 10 to 100 for slower ones). All samplers considered here being unbiased, the integration error is only composed of the variance term. For deterministic samplers such as ours, we apply Cranley-Patterson rotations to achieve several realizations [Pharr and Humphreys 2010] similarly to stochastic samplers.

The integrand functions used in this test are of two kinds. First, we integrated over an analytical disk (see Figure 8), with a radial mean spectral profile in $\mathcal{O}(\rho^{-3})$, which corresponds to a difficult integration test [Pilleboue et al. 2015].

Secondly, we integrated HDR images (see Figure 9). The two HDR images used were chosen because they present very different fourier

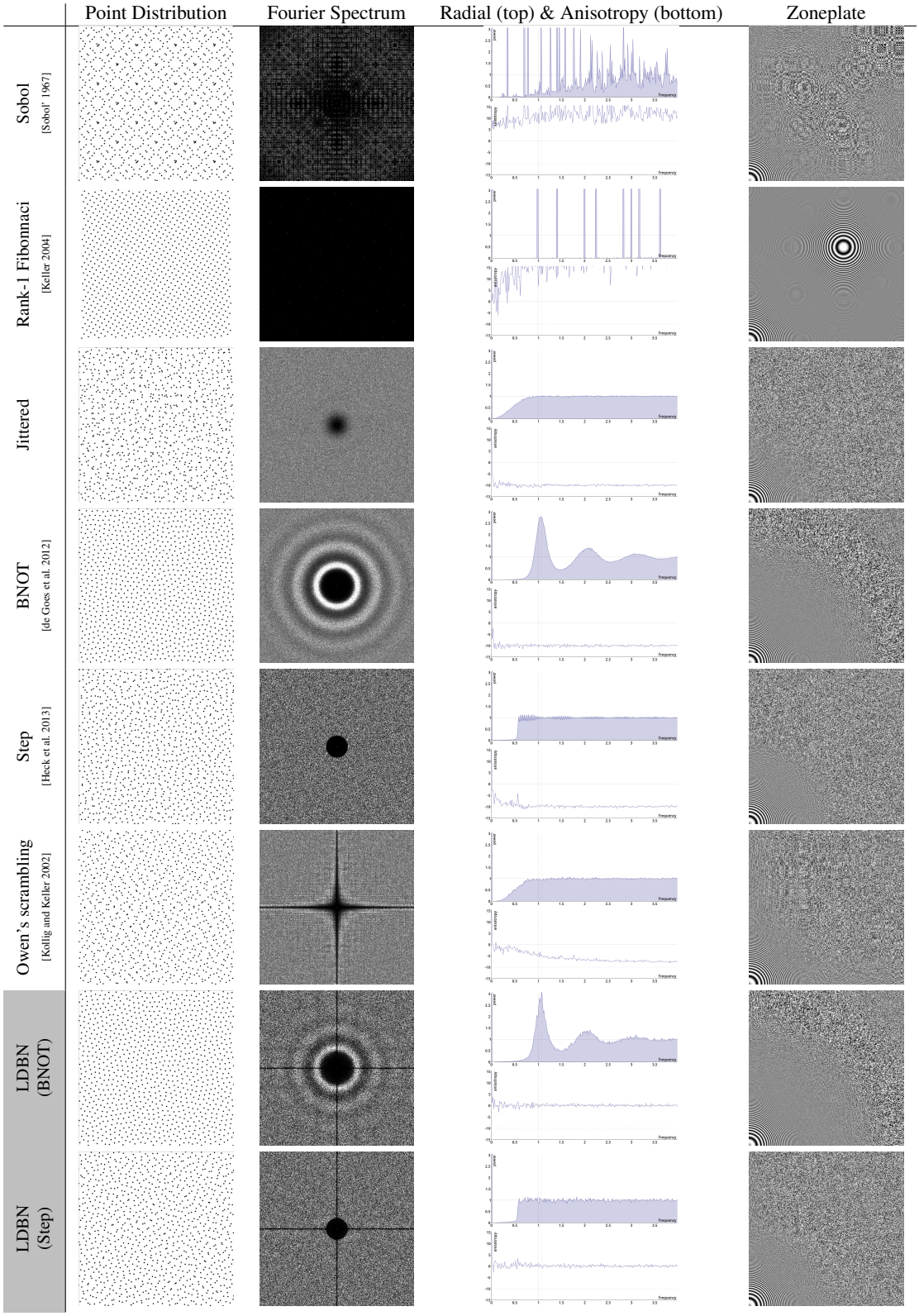


Figure 7: *The Fourier spectra of a single realization of 4096 points for deterministic samplers, and an average of 10 realizations for stochastic samplers. The zoneplate aliasing test uses a single sample per pixel. Please zoom in the Fourier spectra to see narrow peaks. Please refer to the HTML bundle in the supplementary material to visualize the same results without zooming.*

spectra profiles. Natural images spectra can be roughly categorized into two categories [Torralba and Oliva 2003]: The ones showing man made scenes (buildings, streets, etc.) and the ones showing natural scenes (forests, grass, etc.). Man made structures have a much more anisotropic spectra and present a mean spectral profile in $\mathcal{O}(\rho^{-1.8})$ while natural images are much more isotropic with a mean spectral profile in $\mathcal{O}(\rho^{-2})$. Integrating over an HDR image of each kind allows to test the behaviour of a sampler in very different situations. The integrated disk has a radius 0.25 with a sampling domain of $[0,1)^2$. The two gray scale HDR images are taken from the sIBL archive with a size of 1600x1600. In Figure 8 and Figure 9, analytic curves in $\mathcal{O}(1/N)$ and $\mathcal{O}(1/N\sqrt{N})$ correspond to expected behaviors of white noise and jittered grid or blue noise samplers, respectively [Pilleboue et al. 2015].

We observe that integration errors for functions with higher exponents (in absolute value) in their mean spectral profile are similar for all samplers. The differences appear when integrating over functions having a mean spectral profile with a lower exponent (due to the fact that all samplers try to be as close to 0 as possible in the low frequencies). Even though BNOT performs still very well, the importance of LD samplers can be seen more clearly. If one compares Owen's scrambling with jittered grid, even though they have a very similar anti-aliasing property, the low discrepancy property of the Owen's scrambled Sobol sequence leads to better results (Figure 9, bottom image). We can also note that our sampler performs very well, similarly to BNOT and LD samplers, but with the ability to efficiently generate very large point sets.

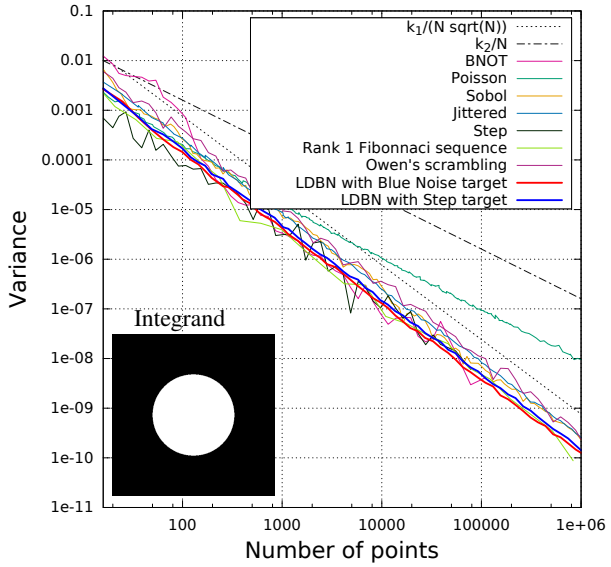


Figure 8: *Measured integration variance over an analytical disk. Samplers such as Step or BNOT require tens of hours to compute over 10^5 points, therefore we could not perform their analysis before 10^6 points.*

6 Conclusions, Limitations, and Discussion

In a certain sense, this work is a proof of concept: we have demonstrated that LD point sets with anti-aliasing features and almost blue-noise properties can be reliably constructed in 2D. Positive aspects of the system include its performance (very few calculations in the runtime, linear time complexity), its compactness, and conceptual simplicit. We also believe that the algorithm can be implemented on the GPU.

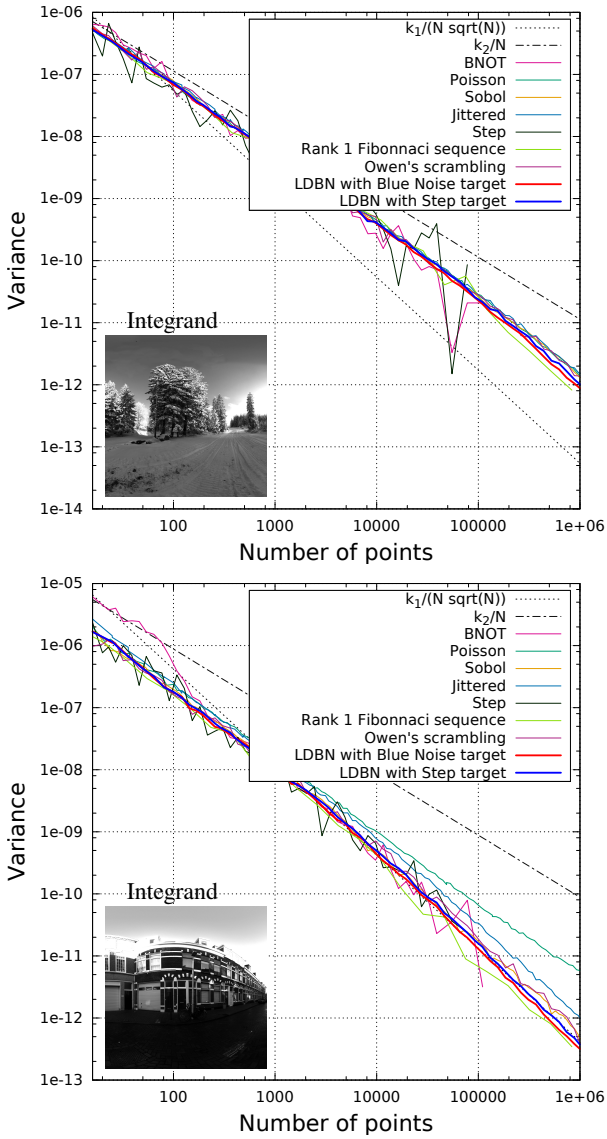


Figure 9: *Measured integration variance over two HDR images from the sIBL archives. Samplers such as Step or BNOT require tens of hours to compute over 10^5 points, therefore we could not perform their analysis until 10^6 points.*

Our construction carries some limitations, and can be improved in many ways. Namely, as we have already mentioned in Section 1, the current version does not yet support adaptive LD sampling. In the future, we would like to extend our system to support adaptivity.

This paper raises a number of important and challenging open problems. First, our current version of the sampling system produces LD point sets, not sequences. It would be very interesting to build LD sequences with blue-noise properties, based on simple and compact constructions. Another interesting direction for future research is to extend our approach to build higher-dimensional sampling systems. Multi-dimensional LD pointsets and sequences with spectral control may play an important role in computer graphics: they might provide excellent support for uniform sampling while being computationally efficient, and may bring satisfactory solutions to many problems that are inherently high-dimensional. We hope that this work will promote further research towards solving such open

problems.

While this work mainly focussed on the theoretical concepts and analysis, and even though we do not claim any optimality, there is still an interesting practical outcome. Indeed, our primary goal was to combine blue noise and low discrepancy, but we ended up with a sampler that combines blue noise, low discrepancy, stratification, look-up, and the Latin hybercube property; all in one framework. Thus, LDBN can readily replace any existing implementation of any of these sampling strategies, and bring the advantages of the others.

The proposed solution, including the code and lookup tables, will be made publicly available.

Acknowledgments

We thank the anonymous reviewers for their detailed feedback to improve the paper. Thanks to Jean-Yves Franceschi and Jonathan Dupuy for reviewing an earlier version of the paper. This project was supported in part by Deutsche Forschungsgemeinschaft Grant (DE-620/22-1), French ANR Excellence Chair (ANR-10-CEXC-002-01) and CoMeDiC (ANR-15-CE40-0006), 973 Program (2015CB352501), National Foreign 1000 Plan (WQ201344000169), National Natural Science Foundation of China (61372168, 61620106003, 61331018), GD Leading Talents Plan (00201509), GD Science and Technology Program (2014B050502009, 2014TX01X033, 2015A030312015, 2016A050503036), and SZ Innovation Program (JCYJ20151015151249564).

References

AHMED, A. G. M., HUANG, H., AND DEUSSEN, O. 2015. AA patterns for point sets with controlled spectral properties. *ACM Trans. Graph. 34*, 6, 212:1–212:8.

BALZER, M., SCHLÖMER, T., AND DEUSSEN, O. 2009. Capacity-constrained point distributions: A variant of Lloyd's method. *ACM Trans. Graph. 28*, 3, 86:1–8.

CHEN, Z., YUAN, Z., CHOI, Y.-K., LIU, L., AND WANG, W. 2012. Variational blue noise sampling. *IEEE Transactions on Visualization and Computer Graphics 18*, 10, 1784–1796.

COHEN, M., SHADE, J., HILLER, S., AND DEUSSEN, O. 2003. Wang tiles for image and texture generation. *ACM Trans. Graphics 22*, 3, 287–294.

COOK, R. L. 1986. Stochastic sampling in computer graphics. *ACM Trans. Graph. 5*, 1, 51–72.

DAMMERTZ, S. 2009. Rank-1 lattices in computer graphics. University of Ulm, PhD Thesis.

DE GOES, F., BREEDEN, K., OSTROMOUKHOV, V., AND DESBRUN, M. 2012. Blue noise through optimal transport. *ACM Trans. Graph. 31*, 6, 171:1–171:11.

DIPPÉ, M. A. Z., AND WOLD, E. H. 1985. Antialiasing through stochastic sampling. In *ACM SIGGRAPH*, 69–78.

DUNBAR, D., AND HUMPHREYS, G. 2006. A spatial data structure for fast Poisson-disk sample generation. *ACM Trans. Graph. 25*, 3, 503–508.

DURAND, F. 2011. A frequency analysis of Monte-Carlo and other numerical integration schemes. *MIT CSAIL Tech. rep. TR-2011-052*.

EBEIDA, M. S., DAVIDSON, A. A., PATNEY, A., KNUPP, P. M., MITCHELL, S. A., AND OWENS, J. D. 2011. Efficient maximal Poisson-disk sampling. *ACM Trans. Graph. 30*, 49:1–49:12.

FATTAL, R. 2011. Blue-noise point sampling using kernel density model. *ACM Trans. Graph. 30*, 3, 48:1–48:12.

FAURE, H. 1982. Discrépances de suites associées à un système de numération (en dimension s). *Acta Arithmetica 41*, 4, 337–351.

HALTON, J. H. 1960. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik 2*, 1, 84–90.

HAMMERSLEY, J. M. 1960. Monte Carlo methods for solving multivariable problems. *Annals of the New York Academy of Sciences 86*, 3, 844–874.

HECK, D., SCHLÖMER, T., AND DEUSSEN, O. 2013. Blue noise sampling with controlled aliasing. *ACM Trans. Graph. 32*, 3, 25:1–25:12.

HLAWKA, E. 1961. Funktionen von beschränkter Variation in der Theorie der Gleichverteilung. *Annali di Matematica Pura ed Applicata 54*, 1, 325–333.

JIANG, M., ZHOU, Y., WANG, R., SOUTHERN, R., AND ZHANG, J. J. 2015. Blue noise sampling using an sph-based method. *ACM Trans. Graph. 34*, 6, 211:1–211:11.

KELLER, A. 2004. Stratification by rank-1 lattices. *Monte Carlo and Quasi-Monte Carlo Methods 2002*, 299–313.

KELLER, A. 2012. Quasi-Monte Carlo image synthesis in a nutshell. *Monte Carlo and Quasi-Monte Carlo Methods*, 213–252.

KENSLER, A. 2013. Correlated multi-jittered sampling. *Pixar Technical Memo 13-01 7*, 86–112.

KOLLIG, T., AND KELLER, A. 2002. Efficient multidimensional sampling. In *Computer Graphics Forum*, vol. 21, 557–563.

KOPF, J., COHEN-OR, D., DEUSSEN, O., AND LISCHINSKI, D. 2006. Recursive Wang tiles for real-time blue noise. *ACM Trans. Graph. 25*, 3, 509–518.

KUHN, H. W. 1955. The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly 2*, 83–97.

KUIPERS, L., AND NIEDERREITER, H. 1974. *Uniform Distribution of Sequences*. Pure and applied mathematics. John Wiley & Sons, New York.

LAGAE, A., AND DUTRÉ, P. 2006. An alternative for Wang tiles: Colored edges versus colored corners. *ACM Trans. Graph. 25*, 4, 1442–1459.

LAGAE, A., AND DUTRÉ, P. 2008. A comparison of methods for generating Poisson disk distributions. *Computer Graphics Forum 27*, 1, 114–129.

LEMIEUX, C. 2009. *Monte Carlo and Quasi Monte Carlo Sampling*. Springer-Verlag New York.

LLOYD, S. 1982. Least squares quantization in PCM. *IEEE Transactions on Information Theory 28*, 2, 129–137.

MATOUŠEK, J. 1998. On the L_2-discrepancy for anchored boxes. *Journal of Complexity 14*, 4, 527–556.

MCCOOL, M., AND FIUME, E. 1992. Hierarchical Poisson disk sampling distributions. In *Proc. Graphics Interface '92*, 94–105.

MITCHELL, D. P., AND NETRAVALI, A. N. 1988. Reconstruction filters in computer-graphics. In *ACM Siggraph Computer Graphics*, vol. 22, ACM, 221–228.

MITCHELL, D. 1991. Spectrally optimal sampling for distributed ray tracing. In *Proc. SIGGRAPH '91*, vol. 25, 157–164.

NIEDERREITER, H. 1988. Low-discrepancy and low-dispersion sequences. *Journal of Number Theory 30*, 1, 51–70.

NIEDERREITER, H. 1992. *Random Number Generation and quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

OSTROMOUKHOV, V., DONOHUE, C., AND JODOIN, P.-M. 2004. Fast hierarchical importance sampling with blue noise properties. *ACM Trans. Graph. 23*, 3, 488–495.

OSTROMOUKHOV, V. 2007. Sampling with polyominoes. *ACM Trans. Graph. 26*, 3, 78:1–78:6.

OWEN, A. B. 1995. Randomly permuted (t,m,s)-nets and (t, s)-sequences. *MCQMC in Scientific Computing. Lecture Notes in Statistics, Vol. 106*, 299–317.

OWEN, A. 2003. Variance and discrepancy with alternative scrambling. *ACM Trans Model Comput Simul 13*, 363–378.

ÖZTIRELI, A. C., AND GROSS, M. 2012. Analysis and synthesis of point distributions based on pair correlation. *ACM Trans. Graph. 31*, 6, 174:1–174:6.

PHARR, M., AND HUMPHREYS, G. 2010. *Physically Based Rendering, Second Edition: From Theory To Implementation*, 2nd ed. Morgan Kaufmann Publishers Inc.

PILLEBOUE, A., SINGH, G., COEURJOLLY, D., KAZHDAN, M., AND OSTROMOUKHOV, V. 2015. Variance analysis for Monte Carlo integration. *ACM Trans. Graph. 34*, 4, 124:1–124:14.

RAMAMOORTHI, R., ANDERSON, J., MEYER, M., AND NOWROUZEZAHRAI, D. 2012. A theory of monte carlo visibility sampling. *ACM Trans. Graph. 31*, 5, 121.

REINERT, B., RITSCHEL, T., SEIDEL, H.-P., AND GEORGIEV, I. 2016. Projective blue-noise sampling. *Computer Graphics Forum 35*, 1, 285–295.

SAKA, Y., GUNZBURGER, M., AND BURKARDT, J. 2007. Latinized, improved LHS, and CVT point sets in hypercubes. *International Journal of Numerical Analysis and Modeling 4*, 3-4, 729–743.

SCHLÖMER, T., HECK, D., AND DEUSSEN, O. 2011. Farthest-point optimized point sets with maximized minimum distance. In *Symp. on High Performance Graphics*, 135–142.

SCHMALTZ, C., GWOSDEK, P., BRUHN, A., AND WEICKERT, J. 2010. Electrostatic halftoning. *Comput. Graph. Forum 29*, 8, 2313–2327.

SHIRLEY, K. C. P., AND WANG, C. 1994. Multi-jittered sampling. *Graphics Gems IV 4*, 370.

SHIRLEY, P. 1991. Discrepancy as a quality measure for sample distributions. In *Proc. Eurographics '91*, 183–194.

SOBOL', I. M. 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki 7*, 4, 784–802.

TEZUKA, S. 1994. A generalization of Faure sequences and its efficient implementation. Tech. rep., RT0105, IBM Research, Tokyo.

TORRALBA, A., AND OLIVA, A. 2003. Statistics of natural image categories. *Network (Bristol, England) 14*, 3, 391–412.

ULICHNEY, R. 1988. Dithering with blue noise. *Proceedings of the IEEE 76*, 1, 56–79.

VAN DER CORPUT, J. 1935. Verteilungsfunktionen. *Proceedings of the Nederlandse Akademie van Wetenschappen*, 38, 813–821.

WACHTEL, F., PILLEBOUE, A., COEURJOLLY, D., BREEDEN, K., SINGH, G., CATHELIN, G., DE GOES, F., DESBRUN, M., AND OSTROMOUKHOV, V. 2014. Fast tile-based adaptive sampling with user-specified Fourier spectra. *ACM Trans. Graph. 33*, 4, 56:1–56:11.

WEI, L.-Y., AND WANG, R. 2011. Differential domain analysis for non-uniform sampling. *ACM Trans. Graph. 30*, 4, 50:1–50:10.

XU, Y., LIU, L., GOTSMAN, C., AND GORTLER, S. J. 2011. Capacity-constrained Delaunay triangulation for point distributions. *Computers & Graphics 35*, 3, 510 – 516.

YUKSEL, C. 2015. Sample elimination for generating Poisson disk sample sets. *Comput. Graph. Forum 34*, 2, 25–32.

ZHOU, Y., HUANG, H., WEI, L.-Y., AND WANG, R. 2012. Point sampling with general noise spectrum. *ACM Trans. Graph. 31*, 4, 76:1–76:11.

A Discrepancy Bounds of Stratified Sets

In this section, we demonstrate that if a 2D stratified pointset, P, as defined in Eq. (1), is viewed as an ensemble of 1D sequences of u's along the columns, and an ensemble of 1D sequences of v's along the rows, then the discrepancy of the pointset is related to the discrepancy of these 1D sequences.

Without loss of generality, let P be scaled to $[0,1)^2$. λ_2 (respectively λ_1) denotes the Lebesgue measure of a subset of $\mathbb{R}^2$ (respectively $\mathbb{R}$). $A(\Omega;P)$ is number of points of P in $\Omega \subset [0,1)^d$ ($d = \{1,2\}$). The discrepancy of a pointset P with respect to a set $\Omega \subset [0,1)^2$ is defined as

$$D_N(\Omega;P) := \left|\lambda_2(\Omega) - \frac{A(\Omega;P)}{N}\right| .$$

The discrepancy of a pointset is defined as the maximum discrepancy for all subsets of $[0,1)^2$:

$$D_N(P) := \max_{\Omega\subset[0,1)^2} \{D_N(\Omega;P)\} .$$

If R_p denotes the rectangle defined by the origin and $p \in [0,1)$, the star-discrepancy of P is defined by

$$D_N^*(P) := \max_{p\in[0,1)^2} \{D_N(R_p;P)\} .$$

Additivity of the Lebesgue measure and $A(\Omega,P)$ leads to the following property: Let A and B be disjoint subset of $[0,1)$, then

$$D_N(A\cup B;P) \le D_N(A;P) + D_N(B;P) . \tag{9}$$

We denote by $\mathbf{u} := \{u_i\}$ a one-dimensional sequence of points in $[0,1)$. $\mathbf{u}(k)$ denotes the set of the first k elements of the sequence. We consider a collection of n such sequences $\mathbf{u}^{(X)} := \left\{u_i^{(X)}\right\}$, indexed by an integer X; that is, $\mathbf{u}^{(X)}$ is the Xth sequence in the collection, while $u_i^{(X)}$ is the i-th entry in that sequence. We consider a similar setup for $\mathbf{v}$, v, and Y, respectively.

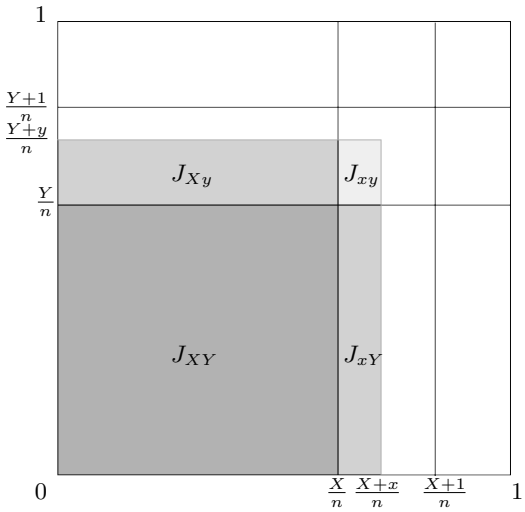


Figure 10: *Notations for Lemma 1.*

In Eq. (2) and our LDBN sampler, we have considered van der Corput LD sequences for each column and row offsets. The results stated below are more general. We consider a more generic two-dimensional stratified point set, with $N = n^2$ points, defined as:

$$P := \left\{ \left(\frac{X + u_Y^{(X)}}{n}, \frac{Y + v_X^{(Y)}}{n} \right) ; X, Y \in \{0 \dots n-1\} \right\} . \tag{10}$$

In other words, the abscissa $\frac{X}{n}$ of a given grid point is shifted by $\frac{u_Y^{(X)}}{n}$, whose numerator is the Yth element of the sequence $\mathbf{u}^{(X)}$. Same for the ordinate. In the following, we simplify the notations of these sequences to $\mathbf{u}$ and $\mathbf{v}$ (and their points $\{u_i\}$ and $\{v_i\}$) in expressions whenever there is no ambiguity on X and Y values.

A bound for the star discrepancy of such a set is obtained in terms of the discrepancy of the constituent sequences $\mathbf{u}$ and $\mathbf{v}$. More formally, we have the following lemma:

Lemma 1. *If $N = n^2$ denotes the total number of points in P, we have*

$$D_N^*(P) \leq \frac{1}{N} \left(\max_{X,Y \in \{0 \dots n-1\}} (X D_X^*(\mathbf{v}) + Y D_Y^*(\mathbf{u})) + 1 \right) . \tag{11}$$

Proof. Let

$$J := [0, (X+x)/n] \times [0, (Y+y)/n] \; ; \quad x, y \in [0, 1) , \tag{12}$$

be an arbitrary interval in I^2. It can be partitioned into non-overlapping sub-intervals (see Figure 10):

$$J = J_{XY} + J_{Xx} + J_{xY} + J_{xy} \tag{13}$$

where

$$\begin{aligned} J_{XY} &:= [0, X/n) \times [0, Y/n) , \\ J_{Xy} &:= [0, X/n) \times [Y/n, (Y+y)/n] , \\ J_{xY} &:= [X/n, (X+x)/n] \times [0, Y/n) , \\ J_{xy} &:= [X/n, (X+x)/n] \times [Y, (Y+y)/n] . \end{aligned}$$

Thanks to the additive property of discrepancy, we have:

$$\begin{aligned} D_N(J; P) \quad \leq \quad & D_N(J_{XY}; P) + D_N(J_{Xy}; P) + \\ & D_N(J_{xY}; P) + D_N(J_{xy}; P) . \end{aligned} \tag{14}$$

Now

$$\begin{aligned} D_N(J_{XY}; P) &= \left| \frac{A(J_{XY}; P)}{N} - \lambda_2(J_{XY}) \right| \\ &= \left| \frac{X \cdot Y}{N} - \frac{X}{n} \cdot \frac{Y}{n} \right| = 0 . \end{aligned} \tag{15}$$

For J_{Xy} we have

$$\left(\frac{X + u_Y}{n}, \frac{Y + v_X}{n} \right) \in J_{Xy} \implies v_X \leq y . \tag{16}$$

If we write J_y to denote the one-dimensional interval $[0, y]$, we observe that

$$A(J_{Xy}; P) = A\left(J_y; \mathbf{v}(X)\right) , \tag{17}$$

and

$$\lambda_2(J_{Xy}) = \frac{X}{n} \cdot \frac{y}{n} = \frac{X \lambda_1(J_y)}{N} , \tag{18}$$

Therefore:

$$\begin{aligned} D_N(J_{Xy}; P) &= \left| \frac{A(J_{Xy}; P)}{N} - \lambda_2(J_{Xy}) \right| \\ &= \left| \frac{A(J_y; \mathbf{v}(X))}{N} - \frac{X \cdot \lambda_1(J_y)}{N} \right| \\ &= \frac{X}{N} \left| \frac{A(J_y; \mathbf{v}(X))}{X} - \lambda_1(J_y) \right| \\ &= \frac{X}{N} D_X(J_y; \mathbf{v}(X)) \quad \leq \frac{X}{N} D_X^*(\mathbf{v}) . \end{aligned} \tag{19}$$

Similarly

$$D_N(J_{xY}; P) \leq \frac{Y}{N} D_Y^*(\mathbf{u}) . \tag{20}$$

Finally, $A(J_{xy}; P)$ is either 0 or 1, and $\lambda_2(J_{xy})$ is at most $1/N$, therefore

$$D_N(J_{xy}; P) \leq \frac{1}{N} . \tag{21}$$

Substituting Eqs. (15, 19, 20, and 21) into Eq. (14):

$$D_N(J; P) \leq \frac{1}{N} \left(X D_X^*(\mathbf{v}) + Y D_Y^*(\mathbf{u}) + 1 \right) \tag{22}$$

and Eq. (11) follows for the upper bound of the star discrepancy of the set P. □

A regular grid uses a fixed value (typically 0 or $\frac{1}{2}$) for all the entries in $\mathbf{u}$ and $\mathbf{v}$, hence $D_X^*(\mathbf{v})$ and $D_Y^*(\mathbf{u})$ are $\mathcal{O}(1)$ and as such $D_N^*(P)$ is $\mathcal{O}\left(\frac{\sqrt{N}}{N}\right)$.

If all of the $\mathbf{u}$'s and $\mathbf{v}$'s are low-discrepancy sequences, that is, $D_X^*(\mathbf{v})$ is $\mathcal{O}(\log(X))$ and $D_Y^*(\mathbf{u})$ is $\mathcal{O}(\log(Y))$, both bounded by $\mathcal{O}(\log(\sqrt{N}))$, then the resulting point-set is a low-discrepancy point-set, with discrepancy in $\mathcal{O}\left(\frac{\log(N)}{N}\right)$. Note, however, that such a construction does not extend automatically to higher dimensions.

Finally:

Corollary 1. *The LDBN point set defined from van der Corput sequences in Eq.(8) has a discrepancy in $\mathcal{O}\left(\frac{\log(N)}{N}\right)$.*

B Approximation Error in Target Matching

The differences between corresponding entries (in assending order) of the vectors $\mathbf{L}$ and $\mathbf{Q}$ in Algorithm 2, in all chunks, determine, along the respective axes, the distances between corresponding points in the given target and its LD approximation constructed from the template LD set. Let us refer to these distances as the "approximation error". To get an estimate of this error we compare the entries to a common reference: the uniform set $\mathbf{A} = \left\{ \frac{2i+1}{2m} \right\}_{i=0}^{m-1}$. The largest difference between an entry $l_i \in \mathbf{L}$ and the corresponding entry $q_i \in \mathbf{Q}$, both sorted in an ascending order, is bounded by

$$|l_i - q_i| \leq |l_i - a_i| + |q_i - a_i| \ ,$$

where a_i is the i'th entry in $\mathbf{A}$. But for any finite set, s, of m points,

$$|s_i - a_i| \leq D_m^*(s) - \frac{1}{2m} \ ;$$

see [Kuipers and Niederreiter 1974, Chapter 2, Theorem 1,4]; therefore:

$$|l_i - q_i| \leq D_m^*(\mathbf{L}) + D_m^*(\mathbf{Q}) - \frac{1}{m} \ .$$

For the van der Corput sequence, and when m is a power of 2, then

$$D_m^*(\mathbf{Q}) = \frac{1}{m} \ ;$$

hence

$$|l_i - q_i| \leq D_m^*(\mathbf{L}) \ . \tag{23}$$

Thus, the approximation error is bounded by the discrepancy of the u's and v's in the chunks. This has been shown in Appendix A to be closely connected with the discrepancy of the whole stratified set. We conclude that the approximation error is directly proportional to discrepancy of the target set, and inversely proportional to the chunk size, since the discrepancy normally decreases with the number of points. Thanks to the inherent stochastic nature of blue-noise sets, this approximation error manifests as random jittering of the points, sized by the estimate in Eq. (23).

Blue-noise point sets attain relatively low discrepancies; cf. Figure 6b. Not so low as to qualify them as LD sets, but low enough to make them easily matchable by rearranging the template LD set. At some point the approximation error becomes small relative to the Poisson disk radius of the set. In our experiments, the error becomes barely noticeable with $m \geq 16$.