

# 一种高效的最大化泊松圆盘采样算法

胡梓榆

## 摘要

本文提出一种在有界非凸区域上生成均匀泊松圆盘采样结果的算法，且该采样满足最大化采样和无偏差采样的特性，生成符合蓝噪声特性的采样点集。该算法的时间效率和空间效率仅依赖于输出采样点的个数。本文方法由两步骤组成，两步骤均基于常规的泊松圆盘采样算法，首先，第一步生成完全覆盖采样区域的由规整的方格组成的网格，基于网格进行泊松圆盘采样，得到初步的无偏差且接近最大化的泊松圆盘采样结果。第二步，生成最大化覆盖采样区域的泊松圆盘采样结果，用多边形拟合网格中未被圆盘覆盖的空白圆弧区域，在空白多边形区域进行迭代式的泊松圆盘采样，直到无法加入新的采样圆盘。该算法的空间效率为  $O(n)$ ，时间效率为  $O(n \log n)$ ，其中  $n$  代表输出采样点的个数。

**关键词：**泊松圆盘采样；蓝噪声采样

## 1 引言

本次选题实现最大化泊松圆盘采样算法，对于给定空间区域内生成随机且均匀的点集，该算法生成的采样点集具有很好的蓝噪声特性，该采样满足无偏差采样、最小距离以及最大化性质，即任意两个采样点之间的距离大于给定的采样半径，采样圆盘可以最大化覆盖采样区域，新的采样点可以无偏差地采样在区域内没有被覆盖的点上。在计算机图形学领域，采样技术有很多应用场景，比如图形渲染、物体分布以及仿真等。最大化泊松圆盘采样算法生成的采样点具有很好的蓝噪声特性，生成的结果在曲面重新网格化、二维或三维的网格生成等问题中发挥重要作用<sup>[1][2]</sup>，基于泊松圆盘采样结果生成的三角网格边长度和角度范围能很好地满足约束条件，因此生成的三角网格具有良好的几何特性。

## 2 相关工作

目前有多种方式对现存的蓝噪声采样技术进行分类<sup>[3]</sup>，比如依据采样区域进行分类，采样区域可分为二维平面、三维空间或者三维平面；依据采样基元进行分类，采样基元可分为点、线或者球等；依据采样结果的特性可以分类为各向同性和非各向同性；此外，还可以根据采样算法的种类分类为基于泊松圆盘采样的方法、基于优化的蓝噪声采样方法等。

### 2.1 基于泊松圆盘采样的蓝噪声采样

泊松圆盘采样是一种经典的生成随机均匀分布的点集的算法。一个理想的泊松圆盘采样方法要求满足以下三个特性：（1）最小距离特性，要求任意两个采样圆盘的中心距离大于采样半径；（2）无偏差采样特性，要求新的采样点落在某一子区域内的概率由子区域的面积与可采样的空白区域的比例确定；（3）最大化采样特性，要求采样圆盘的并集最大化覆盖采样区域。传统的泊松圆盘采样方法由Cook<sup>[4]</sup>提出，给定采样区域和采样半径，该方法生成随机分布的采样圆盘，如果新生成的采样圆盘与现有的采样圆盘冲突，则撤销此次采样，否则保留此次采样。重复上述采样过程，直至连续多次撤销采样，采样过程终止。此方法的局限性在于无法高效地实现采样结果的最大化特性。因此，很多研究

致力于提高泊松圆盘采样算法的时间效率。很多研究通过构建新的数据结构来存储空白待采样区域实现高效的泊松圆盘采样算法，比如四叉树。

White<sup>[5]</sup>等人首先提出使用四叉树提升算法时间效率。网格的边长设置为  $r/\sqrt{2}$ ，使得每个网格单元最多容纳一个半径为  $r$  的采样圆盘。在采样过程中，被部分覆盖的网格单元，使用四叉树的方式将其分割为更小的单元。

Jones<sup>[6]</sup>等人首先提出无偏差的最大化的泊松圆盘采样算法，通过使用维诺图提取网格中未被圆盘覆盖的区域，并在未覆盖的区域进行无偏差的采样最终得到最大化的采样结果。其核心思想是当且仅当维诺图划分的单元格被其顶点上圆盘完全覆盖时，则生成的采样结果满足最大化特性。在采样过程中，每次生成新的采样点首先依据单元格中空白区域的面积选取单元格进行采样，采样过程迭代多次直至满足最大化采样结果的特性。上述的方法仅能生成均匀采样的结果，Yan<sup>[7]</sup>等人提出一种生成不同尺寸采样圆盘的泊松圆盘采样方法。

## 2.2 基于优化的蓝噪声采样

基于迭代式优化的采样方法一般分为两步骤：（1）生成初始点集合  $X = \{\mathbf{x}_i\}_{i=1}^n$ ，（2）优化  $X$  中点的位置直至收敛。这类方法设置的目标优化方程各不相同，最原始的优化算法设置的能量方程如下式所示：

$$E_{\text{CVT}}(X) = \sum_{i=1}^n \int_{V_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 d\mathbf{x}$$

其中， $\{V_i\}_{i=1}^n$  代表在采样区域  $\Omega$  内由采样点构造的维诺图， $\rho(\mathbf{x})$  代表在区域  $\Omega$  上定义的密度函数。这一方法的局限性在于生成的采样点具有规整的分布模型，蓝噪声特性不明显，后续一些研究<sup>[8][9][10]</sup>对该模型进行改进得到更好的符合蓝噪声特性的采样结果。Balzer<sup>[8]</sup>等人提出一种改进的模型使得生成的采样点集合具有很好的蓝噪声特性，该方法要求维诺图中的单元格需要满足以下约束  $\|V_i\| = \int_{V_i} \rho(\mathbf{x}) d\mathbf{x} = c_i$ ，其中  $c_i$  代表容量约束项且  $c_i > 0$ 。该方法可以获得很好的符合蓝噪声特性的采样点结果但是其收敛速度较慢，在生成大规模采样结果时时间效率不高。

## 3 本文方法

### 3.1 本文方法概述

本文提出一种高效的算法解决最大化泊松圆盘采样问题。对于给定的采样区域  $D$ ，最大化泊松圆盘采样算法生成随机的采样点集合  $X = \{\mathbf{x}_i\}$ ，且要求采样点满足三个性质：（1）最小距离限制，即采样点之间的距离均大于最小距离，如下式所示

$$\forall \mathbf{x}_i, \mathbf{x}_j \in X, \mathbf{x}_i \neq \mathbf{x}_j : \|\mathbf{x}_i - \mathbf{x}_j\| \geq r$$

设定圆盘半径为  $r$ ，生成采样点后，可采样区域  $D_i$  为  $D$  的子区域。（2）最大化性质，生成的采样圆盘最大化覆盖采样区域，即最终采样区域无法再加入新的采样点，如下式所示

$$\forall \mathbf{x} \in D, \exists \mathbf{x}_i \in X : \|\mathbf{x} - \mathbf{x}_i\| < r$$

（3）无偏差性质，新的采样点落在某一子区域内的概率由子区域的面积与可采样的空白区域的比例确定，如下式所示

$$\forall \mathbf{x}_i \in X, \forall \Omega \subset D_{i-1} : P(\mathbf{x}_i \in \Omega) = \frac{\text{Area}(\Omega)}{\text{Area}(D_{i-1})}$$

本论文提出的最大化泊松圆盘采样算法由四步骤组成：（1）生成背景网格，对于采样区域生成规整的背景网格用于引导采样点的生成，（2）基于网格的泊松圆盘采样，在网格中进行迭代式的泊松圆盘采样，未被覆盖的采样区域在下一步骤进行处理，（3）计算空白区域的多边形近似，对于采样圆盘未覆盖的区域计算其多边形近似，（4）多边形区域内的泊松圆盘采样。

### 3.2 生成背景网格

此步骤使用泛洪算法生成规整的完全覆盖采样区域  $D$  的背景网格。采样区域  $D$  作为算法的输入，使用边界线段集合表示。网格的边长设置为  $r/\sqrt{2}$ ，即网格的对角线长度为  $r$ ，因此每个网格最多容纳一个采样点。设置网格的有效性属性，若网格可以容纳采样点，则有效性属性为 1，否则为 0。依据采样区域的边界线段，将背景网格划分为内部网格即完全在采样区域内部的网格、边界网格即与采样区域边界有交点的网格和外部网格即完全位于采样区域外部的网格。

### 3.3 基于网格的泊松圆盘采样

基于步骤一生成的背景网格对采样区域  $D$  进行初步泊松圆盘采样，首先均匀随机地选取一个网格，且网格的有效属性为真，其次在选取的网格内进行均匀随机采样得到采样点，若采样点不满足最小距离限制或者采样点落在边界外部，则此次采样视为落空，否则此次采样成功，保留采样点并设置网格的有效性属性为 0。上述过程迭代进行多次，迭代次数的下限为  $|C|/16$  次，其中  $|C|$  为初始的有效网格的个数，上限为  $5|C|$  次，且若落空的次数达到 400 次，则迭代终止。经过迭代式的采样后，采样区域中未被采样圆盘覆盖的子区域  $V$ ，即还可以容纳采样点的空白区域在后续步骤中进行处理。

### 3.4 计算空白区域的多边形近似

基于步骤二的采样结果，首先计算网格中未被采样圆盘覆盖的连通区域，即可以放置采样点的空白区域  $V$ ，由线段和圆弧构成的弧多边形  $V_r$ 。使用多边形  $V_p$  近似表示  $V$ 。计算空白区域的多边形近似步骤如下：（1）初始化  $V_p$  为网格方形，计算其与采样区域边界的交点，保留采样区域内的部分；（2）基于邻域网格的采样圆盘，从  $V_p$  减去采样圆盘覆盖的区域。对于邻域的采样圆盘  $d$ ，首先计算圆盘  $d$  和  $V_p$  的交点，如果没有边与  $d$  相交则被完全覆盖，如果两个交点构成的弦将  $V_p$  的顶点  $c$  与采样圆盘的中心分割，则将交点加入多边形的顶点列表，并将被覆盖的顶点移除顶点列表；（3）对新加入的顶点位置进行微调，调整为圆盘和弧多边形  $V_r$  的交点，得到更精确的近似。上述步骤的图示见图 1。若构成弧多边形  $V_r$  的两条弧相交点均落在  $V_p$  内部时，步骤（3）优化顶点位置后，将  $V_p$  分割成多个子区域，使得多边形  $V_p$  更好拟合原空白区域  $V_r$ ，如图 2 所示。

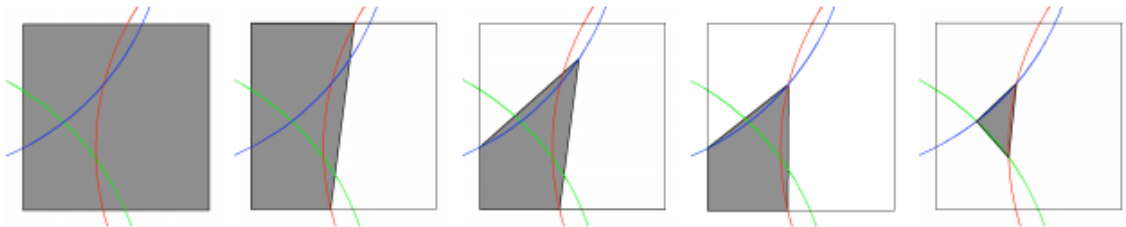


图 1: 计算空白区域的多边形近似（灰色）的过程示意

### 3.5 多边形区域内的泊松圆盘采样

此步骤的泊松圆盘采样过程与步骤二类似，但采样对象为多边形空白区域  $V_p$ 。首先选取一个空白多边形，为保证采样过程的无偏差性质，依据子区域面积与空白区域面积的比例选取空白多边形，其次在选取的空白多边形内部进行均匀随机采样得到采样点，并检测新的采样点是否满足最小距离限

制，若满足则保留采样点。若在每次采样之后对剩余空白区域进行立即更新，其时间开销很大，因此采用集中式的更新策略，即在一轮采样之后对剩余空白区域进行更新。初始的空白区域为  $\mathcal{V}_0$ ，步骤  $i$  时刻，空白区域为  $\mathcal{V}_i$ ，一轮采样的采样次数下限为  $|\mathcal{V}_i|/16$  次，上限为  $3|\mathcal{V}_i|$  次，且若落空的次数达到 100 次，则此轮采样终止。在一轮采样之后，依据新增的采样点，根据步骤三的策略更新空白区域，得到  $\mathcal{V}_{i+1}$ ，并更新剩余的空白多边形与总空白区域的面积占比。

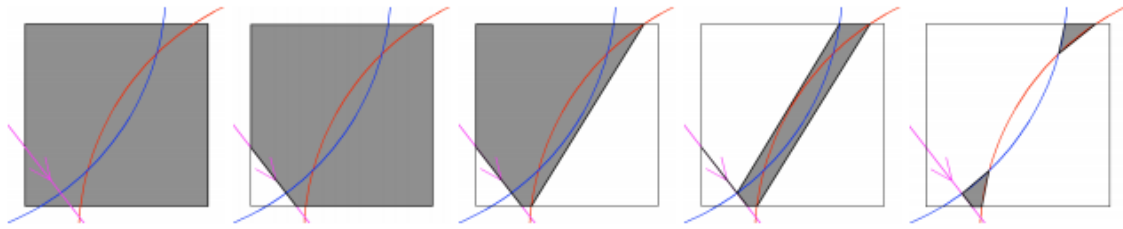


图 2: 计算空白区域的多边形近似（灰色），优化多边形的顶点位置过程示意

## 4 复现细节

### 4.1 与开源工作对比

此次复现的论文没有公开的源代码，本次复现工作无参考他人代码。本次复现完整实现了论文提出的最大化泊松圆盘采样算法，并实现交互式界面体现算法的采样过程和采样结果。算法的复现逻辑分为两大步骤：（1）根据背景网格进行初步泊松圆盘采样；（2）根据剩余的空白区域进行迭代式泊松圆盘采样，得到最大化的泊松圆盘采样结果。

步骤一，网格引导的泊松圆盘采样。首先，生成规整的背景网格，网格的边长为  $r/\sqrt{2}$ ，即网格的对角线长度为  $r$ ， $r$  代表采样圆盘的半径。基于背景网格进行初步采样，每次采样包含采样和检测两个步骤。首先随机选取一个待采样的网格并在采样网格中随机均匀采样得到采样点。对于新的采样点，检测其邻域网格中是否有采样点，若有采样点且距离小于最小距离限制，则此次采样落空即采样点无效，否则此次采样成功，设置采样网格的有效性为 0，并将采样点加入采样点集合  $P$  中。此步骤的迭代采样次数下限为  $|C|/16$  次，其中  $|C|$  为初始的有效网格的个数，上限为  $5|C|$  次，且若落空的次数达到 400 次，则迭代终止。此步骤得到初步的泊松圆盘采样结果，满足最小距离限制以及无偏差采样特性，但是采样结果并不是最大化的。此步骤输出采样点集合  $P$ ，记录采样点的位置信息；采样网格的属性数组  $flag$ ，记录采样网格的有效性属性。

步骤二，根据未被采样圆盘覆盖的空白区域  $V_r$  进行泊松圆盘采样，此步骤的伪代码如所示。在复现过程中，由三个代码块组成：（1）计算网格中未被采样圆盘覆盖的空白区域  $V_r$  的多边形近似，（2）优化  $V_r$  的多边形近似，（3）根据网格空白多边形进行泊松圆盘采样。

---

**Procedure 1** Dart throwing in polygon Voids.

---

**Input:** Sample domain, Valid cells  $V_i$ , Sampleradius  $r$ **Output:** Maximal sample result**repeat***GeneratePolygonVoid();**ComputePolygonVoidArea();***repeat***SampleInPolygonVoids();**CheckSampleValid();***until**  $Sample_{num}=3|V_i|$  or 100 miss;*UpdateNeighborVoid();***until**  $|V_i| = 0$ ;

---

计算网格中未被采样圆盘覆盖的空白区域  $V_r$  的多边形近似。 $V_r$  由线段和圆弧构成，使用圆弧上的弦代替圆弧，计算空白区域  $V_r$  的多边形近似  $V_p$ ，使用逆时针排序的顶点序列  $P$  表示  $V_p$ ， $V_p$  初始为所在网格的整个方形。遍历所有网格，记录网格的邻域网格中采样圆盘的圆心位置，对于每个邻域采样圆盘  $c$ ，根据采样圆盘  $c$  和网格的交点对  $P$  进行更新。对于网格的每条边  $e$ ，判断边  $e$  与圆  $c$  的位置关系，完全位于圆内部，即边  $e$  与圆的交点个数为 0 且端点都位于圆内；与圆相交，即边  $e$  与圆有一个交点；或者完全位于圆外部，即边  $e$  与圆没有交点且端点均位于圆外部。基于边和圆  $c$  的位置关系推导出网格和圆盘  $c$  的位置关系，完全位于圆内部，与圆相交，或者完全位于圆外部。若网格完全位于圆内部，设置网格的有效性属性为 0，若网格完全位于圆外部，设置  $V_p$  为原始网格方形。若网格与圆相交，则根据交点所在的边的端点信息，将交点加入  $V_p$  的顶点序列  $P$  中合适的位置，并删除位于圆内部的顶点，保证  $P$  中顶点按逆时针顺序排列。

优化  $V_r$  的多边形近似。目前更新后的  $P$  使用边与圆弧的交点表示，为更精准地拟合  $V_r$ ，使用圆弧和圆弧的交点对  $P$  进行更新。对于圆弧  $c_1$  和圆弧  $c_2$ ，计算交点信息，若交点个数为 0，则  $P$  不需要更新，若交点个数为 1，且交点位于  $V_p$  内，即两圆相切，将交点加入  $P$ ，若交点个数为 2，将位于  $V_p$  内部的交点加入  $P$ ，基于弧交点对顶点序列  $P$  进行更新的操作，需要考虑圆弧  $c_1$  和  $c_2$  对  $V_p$  进行分割的多种情况，插入  $P$  的合适位置，并删除替代的弧-边交点。

根据网格空白多边形进行泊松圆盘采样。首先，计算网格中空白多边形的面积  $S_i$ ，以及与总空白区域面积  $S$  的占比。其次，在空白区域内进行泊松圆盘采样，与步骤一类似，首先选取空白多边形  $V_p$ ，再在空白多边形  $V_p$  内部进行均匀随机采样，根据的重心将  $V_p$  划分为多个三角形，将采样对象由多边形转化为三角形，实现无偏差采样。

## 4.2 实验环境搭建

本次复现工作在 Windows 系统下使用 C++ 编程实现，可视化界面使用 QT 框架，具体版本为 QT5.12.5。

## 4.3 界面分析与使用说明

本次复现工作实现的操作界面如图 3 所示，界面的主体显示背景网格以及泊松圆盘采样的结果，界面的右侧信息栏实现参数的输入和输出，以及操作按钮。输入网格边长参数，设置背景网格的边长为  $l_1$ ，即设置采样点之间的最小距离  $r = \sqrt{2}l_1$ ，输出参数显示采样过程中采样点的个数。右侧的按钮



从上到下，按钮 1 控制进行初步泊松圆盘采样，按钮 2 控制在待采样区域即灰色区域进行泊松圆盘采样。按钮 3 设置采样圆盘的显示半径  $r' = \frac{1}{2}r$ ，按钮 4 设置采样圆盘的显示半径为  $r' = r$ 。如图 3 所示，当显示的圆盘半径为采样点的最小距离即  $r' = r$  时，界面上显示的灰色区域为计算得到的待采样区域的多边形近似。

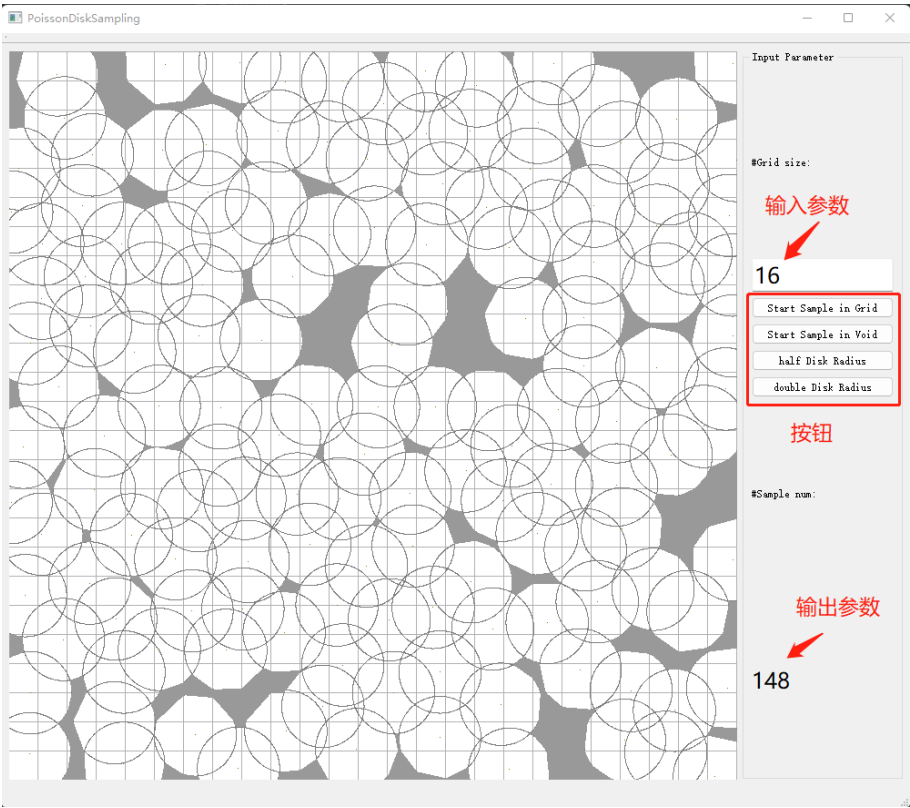


图 3: 操作界面示意

### 5 实验结果分析

本部分对实验所得结果进行分析，详细对实验内容进行说明，实验结果进行描述并分析。首先，算法第一步骤基于网格进行初步的泊松圆盘采样，得到的采样结果如图 4 所示，图的左侧显示当圆盘的显示半径等于最小距离时，采样区域中可接收采样点的区域，用灰色表示，图的右侧显示初步采样的结果。

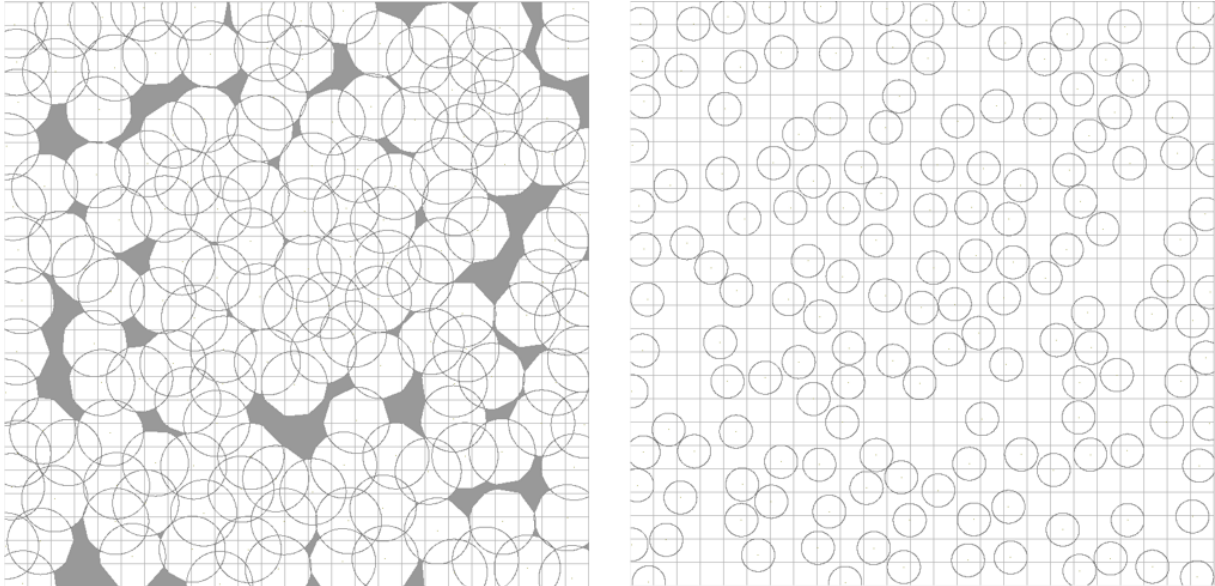


图 4: 算法初步采样结果示意

算法的第二步骤基于计算得到的可采样区域的多边形进行最大化泊松圆盘采样，得到的采样结果如图 5 所示，图的左侧显示当圆盘的显示半径等于最小距离时，此时采样区域中不存在可接收采样点的区域（灰色区域），即泊松圆盘采样结果满足最大化性质，图的右侧显示最终的采样结果。

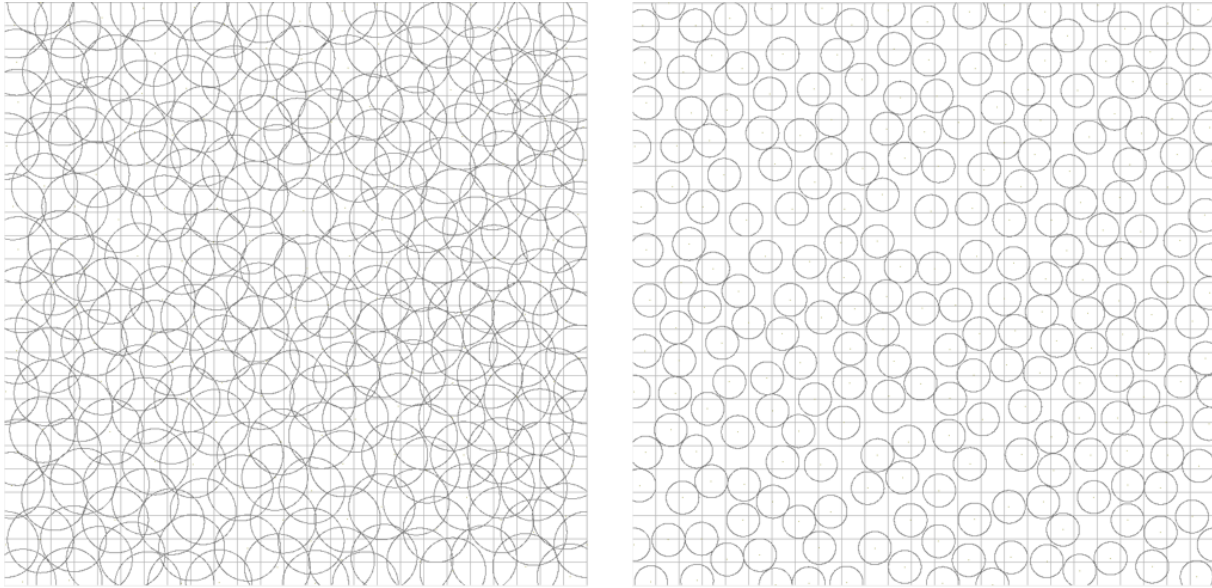


图 5: 算法最终采样结果示意

为验证该采样算法在大规模生成蓝噪声点集时的效果，测试算法在输入的采样半径参数较小的情况下生成的结果，如图 6 所示。由实验结果可知，本次复现的工作可以实现大规模的蓝噪声采样点集生成。

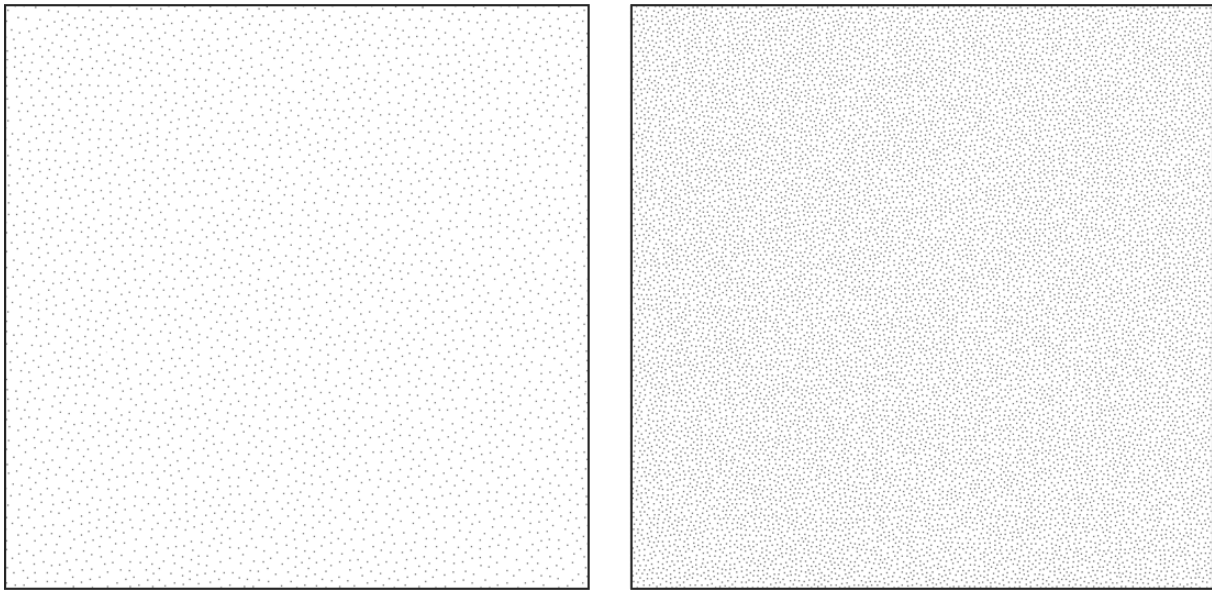


图 6: 方形区域内生成大规模蓝噪声点集的结果图，左图展示当采样半径  $r = 4\sqrt{2}$  时 3K 个采样点，右图当采样半径  $r = 2\sqrt{2}$  时 10K 个采样点

## 6 总结与展望

本文介绍了一种高效的最大化泊松圆盘采样算法，对于给定的二维区域生成随机且均匀的符合蓝噪声特性的点集，该方法的优势在于可以同时满足以下要求：（1）算法可以生成最大化的采样点集，（2）无偏差采样，（3）算法的时间效率高，为  $O(n \log n)$ ，且空间效率为  $O(n)$ 。因此该算法可以生成大规模的蓝噪声点集。本次复现工作基于论文中对算法的描述，算法的核心步骤均已实现，并实现交互式界面展示算法运行过程和最终结果。目前复现工作的不足在于，算法的时间效率不能达到论

文中描述的每秒生成 100,000 个采样点的高效率状态。目前,该算法的采样对象为二维平面区域,后续研究可以考虑将算法扩展到三维空间或者三维曲面上,或者研究在圆盘半径自适应变化的情况下,生成蓝噪声点集。

## 参考文献

- [1] EBEIDA M S, MITCHELL S A, DAVIDSON A A, et al. Efficient and good Delaunay meshes from random points[J]. *Comput. Aided Des.*, 2011, 43: 1506-1515.
- [2] GUO J, YAN D M, BAO G, et al. Efficient triangulation of Poisson-disk sampled point sets[J]. *The Visual Computer*, 2014, 30: 773-785.
- [3] YAN D M, GUO J, WANG B, et al. A Survey of Blue-Noise Sampling and Its Applications[J]. *Journal of Computer Science and Technology*, 2015, 30: 439-452.
- [4] COOK R L. Stochastic sampling in computer graphics[J]. *ACM Trans. Graph.*, 1988, 5: 51-72.
- [5] WHITE K B, CLINE D, EGBERT P K. Poisson Disk Point Sets by Hierarchical Dart Throwing[J]. 2007 IEEE Symposium on Interactive Ray Tracing, 2007: 129-132.
- [6] JONES T R. Efficient Generation of Poisson-Disk Sampling Patterns[J]. *Journal of Graphics Tools*, 2006, 11: 27-36.
- [7] YAN D M, WONKA P. Gap processing for adaptive maximal poisson-disk sampling[J]. *ArXiv*, 2012, abs/1211.3297.
- [8] BALZER M, SCHLÖMER T, DEUSSEN O. Capacity-constrained point distributions: a variant of Lloyd's method[J]. *ACM Trans. Graph.*, 2009, 28: 86.
- [9] XU Y, LIU L, GOTSMAN C, et al. Capacity-Constrained Delaunay Triangulation for point distributions [J]. *Comput. Graph.*, 2011, 35: 510-516.
- [10] CHEN Z, YUAN Z, CHOI Y K, et al. Variational Blue Noise Sampling[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2012, 18: 1784-1796.