

Université de Montréal

Échantillonnage basé sur les Tuiles de Penrose
et applications en infographie

par

Charles Donohue

Département d'informatique et de recherche opérationnelle

Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en informatique

Août 2004

© Charles Donohue, 2004



QA

76

U54

2004

V.034



Direction des bibliothèques

AVIS

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

NOTICE

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé

Échantillonnage basé sur les Tuiles de Penrose
et applications en infographie

présenté par

Charles Donohue

a été évalué par un jury composé des personnes suivantes:

Pierre Poulin, président du jury
Victor Ostromoukhov, directeur de recherche
Philippe Langlais, membre du jury

Mémoire accepté le
24 septembre 2004

Sommaire

L'échantillonnage est un processus omniprésent dans le domaine de l'infographie. Notamment, il joue un rôle central dans l'estimation d'intégrales servant à générer des images de synthèse. Il peut aussi servir dans le rendu en demi-ton, en traitement d'images, ainsi qu'en modélisation géométrique. Il n'existe pas de méthode définitive qui assure les meilleurs résultats dans tous les cas ; l'utilisation d'un échantillonnage régulier s'avère inefficace et souvent détrimental, car il peut introduire un biais dans la solution qui se manifeste par exemple en aliassage. Un échantillonnage purement stochastique comporte aussi ses problèmes, notamment dans la variance des résultats. Plusieurs chercheurs se sont donc penchés sur le sujet, et il existe d'ailleurs une multitude de stratégies d'échantillonnage utilisées en infographie.

Nos travaux de recherche portent sur l'utilisation des tuiles de Penrose pour aborder le problème de l'échantillonnage. Il s'avère que ce jeu de tuiles comporte certaines propriétés qui peuvent être exploitées dans ce contexte. Ce mémoire présente deux articles qui sont les fruits de cette recherche.

Le premier article présenté, *Fast Hierarchical Importance Sampling with Blue Noise Properties*, propose un système d'échantillonnage basé sur les tuiles de Penrose. À partir d'une fonction de densité en deux dimensions, le système peut générer un ensemble de points d'échantillonnage avec une bonne distribution spatiale. Notre méthode donne des résultats d'une très bonne qualité tout en se classant parmi les plus rapides.

Le second article, *Fast Triangulated Importance Sampled Point Sets*, propose un système qui permet de générer un jeu de points tel que dans l'article précédent, mais augmenté d'une triangulation Delaunay de ces points. Notre méthode exploite certaines propriétés de l'échantillonnage avec les tuiles de Penrose de façon à obtenir un temps de triangulation plus rapide que les meilleures techniques existantes.

Mots Clés : Échantillonnage, Bruit bleu, Tuiles de Penrose, Numérotation de Fibonacci, Rendu, Cartes d'environnement, Triangulation de Delaunay.

Abstract

Sampling is a process that is omnipresent in computer graphics. Specifically, it plays an important role in the estimation of integrals used in digital image synthesis. It is also used in digital halftoning, image processing, as well as geometric modelling. There is no definitive sampling strategy that can ensure the best results in all cases; regular sampling proves to be inefficient and often detrimental to the quality of the results, because it can introduce a bias that can manifest itself as aliasing. Pure stochastic sampling also has its problems, notably due to the variance in the results. Many researchers have thus studied this problem, and there are many sampling strategies used in computer graphics.

Our research focuses on using the Penrose tiles to address the sampling problem. It so happens that this set of tiles harbors certain properties that can be exploited in this context. This thesis presents two contributions that stem from this research.

The first article presented, *Fast Hierarchical Importance Sampling with Blue Noise Properties*, proposes a sampling system based on Penrose tiles. Given an importance density function in two dimensions, the system can generate a discrete sample distribution, in which the local point density is proportional to the given function, with a local blue noise distribution. Our technique is amongst the fastest, yet it is also amongst the best in terms of quality.

The second article presented, *Fast Triangulated Importance Sampled Point Sets*, proposes a system that can not only generate point sets as in the prior system, but can also build a Delaunay triangulation of these points. Our method exploits certain properties of sampling with Penrose tiles in order to obtain an efficiency greater than all known Delaunay triangulation algorithms.

Keywords : Sampling, Blue noise, Penrose tiles, Fibonacci numbers, Rendering, Environment mapping, Delaunay triangulations.

Table des matières

| | |
|---|------------|
| Remerciements | vii |
| 1 Introduction | 1 |
| 2 Articles | 3 |
| 2.1 <i>Fast Hierarchical Importance Sampling with Blue Noise Properties</i> | 4 |
| 2.1.1 Résumé | 4 |
| 2.1.2 L'Article | 5 |
| 2.1.3 Discussion | 13 |
| 2.1.4 Contribution Personnelle | 13 |
| 2.1.5 Droits de publication | 14 |
| 2.2 <i>Fast Triangulated Importance Sampled Point Sets</i> | 15 |
| 2.2.1 Résumé | 15 |
| 2.2.2 L'Article | 16 |
| 2.2.3 Discussion | 24 |
| 2.2.4 Contribution Personnelle | 25 |
| 2.2.5 Droits de publication | 25 |
| 3 Conclusion | 26 |
| Bibliographie | 27 |

Remerciements

Je tiens à remercier mon directeur de recherche, Victor Ostromoukhov. Non seulement m'a-t-il convaincu de faire une maîtrise à l'Université de Montréal, il m'a aussi initié au monde de la recherche et m'a donné l'opportunité d'explorer des sujets captivants. Je lui en serai toujours reconnaissant.

Je tiens aussi à remercier Pierre Poulin et Neil Stewart, dont les conseils perspicaces m'ont été indispensables, ainsi que Philippe Langlais, Pierre-Marc Jodoin, et Kari Pulli. Merci au CRSNG et au FCAR équipe pour leur soutien au laboratoire d'infographie.

Chapitre 1

Introduction

Introduction

Quand les pavages apériodiques de Penrose ont été introduits à la fin des années 1970, Roger Penrose lui-même les catégorisait dans le domaine des ‘mathématiques récréatives’. À part le fait que ses tuiles aient inspiré la découverte d’une nouvelle classe de cristaux, les *quasicristaux*, elles ont toujours été considérées comme des curiosités.

Toutefois, certains parallèles ont déjà été observés entre ce genre de pavage et la théorie de l’échantillonnage, ce qui porte à croire qu’il pourrait y avoir des applications utiles aux tuiles de Penrose, mais cette piste n’avait pas encore été explorée pleinement. Dans les travaux de recherche présentés dans ce mémoire, nous avons entrepris de trouver et d’exploiter ces propriétés des tuiles de Penrose, afin d’élaborer de nouveaux outils qui traitent de certains problèmes en échantillonnage.

En infographie, on doit souvent faire appel à des techniques reliées à l’échantillonnage, que ce soit en rendu, en traitement d’images, en traitement de géométrie, ou en vision. Il nous semble donc clair que les méthodes qui sont présentées dans les articles suivants seront d’une grande utilité pour le domaine de l’infographie.

La science ne consiste pas seulement à savoir ce qu'on doit ou peut faire, mais aussi à savoir ce qu'on pourrait faire quand bien même on ne doit pas le faire.

Umberto Eco, *Le nom de la Rose*

Chapitre 2

Articles

Ce mémoire de maîtrise est présenté sous la forme *par article*, et est composé de deux travaux : Le premier article, *Fast Hierarchical Importance Sampling with Blue Noise Properties*, introduit un système d'échantillonnage original basé sur les tuiles de Penrose. Le second article, *Fast Triangulated Importance Sampled Point Sets*, présente une extension au système précédent, en s'attaquant au problème de la connectivité des échantillons, au sens de Delaunay.

Ces deux articles synthétisent bien l'axe principal de mes recherches avec le professeur Ostromoukhov au long de ma maîtrise. Ces travaux sont le fruit d'un travail de longue haleine, et nous espérons qu'ils seront bien reçus par la communauté scientifique en infographie.

2.1 *Fast Hierarchical Importance Sampling with Blue Noise Properties*

Ce premier article a été publié dans ACM SIGGRAPH 2004, et ses auteurs sont Victor Ostromoukhov, Charles Donohue et Pierre-Marc Jodoin.

Une séquence vidéo qui accompagne l'article peut être téléchargée à partir du site suivant :
<http://www.iro.umontreal.ca/~ostrom/ImportanceSampling>

2.1.1 Résumé

Cet article présente une nouvelle méthode pour générer efficacement une bonne distribution d'échantillons à partir d'une fonction de densité d'importance sur un domaine à deux dimensions. Un jeu de tuiles de Penrose est subdivisé hiérarchiquement de manière à générer un nombre suffisamment important de points d'échantillonnage. Ces points sont comptés à l'aide du système de numérotation de Fibonacci, et ces numéros servent à seuiller les échantillons contre la valeur locale de la fonction de densité d'importance. Des vecteurs de correction pré-calculés, obtenus par relaxation, sont utilisés afin d'améliorer le comportement spectral de la distribution de points. La technique est déterministe et très rapide. Les temps d'échantillonnage sont linéairement proportionnels au nombre d'échantillons exigés. Nous illustrons notre technique avec le cas de l'échantillonnage de cartes d'environnement pour l'illumination de scènes synthétiques, mais la technique se prête bien à une grande variété d'applications en infographie, comme par exemple le traitement de géométrie, le rendu en demi-tons, ainsi qu'une variété d'autres problèmes de rendu.

Mots Clés : Échantillonnage d'importance, Bruit bleu, Tuiles de Penrose, Numérotation Fibonacci, Rendu, Cartes d'environnement.

2.1.2 L'Article

Fast Hierarchical Importance Sampling with Blue Noise Properties

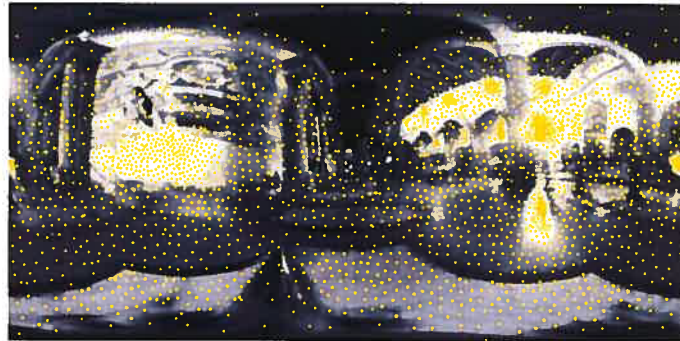
Victor Ostromoukhov*
University of MontrealCharles Donohue †
University of MontrealPierre-Marc Jodoin †
University of Montreal

Figure 1: A high dynamic range 1024×512 environment map [Debevec 98] sampled with 3000 point lights. In this image, importance density is represented by the lightness of the background. It took 0.064 seconds on a 2.6 GHz P4 to generate this point set. Similar results using a hardware accelerated Lloyd relaxation [Hoff et al. 1999] required 1 second, while *Structured Importance Sampling* [Agarwal et al. 2003] took 1393 seconds.

Abstract

This paper presents a novel method for efficiently generating a good sampling pattern given an importance density over a 2D domain. A Penrose tiling is hierarchically subdivided creating a sufficiently large number of sample points. These points are numbered using the Fibonacci number system, and these numbers are used to threshold the samples against the local value of the importance density. Pre-computed correction vectors, obtained using relaxation, are used to improve the spectral characteristics of the sampling pattern. The technique is deterministic and very fast; the sampling time grows linearly with the required number of samples. We illustrate our technique with importance-based environment mapping, but the technique is versatile enough to be used in a large variety of computer graphics applications, such as light transport calculations, digital halftoning, geometry processing, and various rendering techniques.

CR Categories: I.3.3 [Picture/Image Generation]: Anti-aliasing; I.3.m [Miscellaneous]: Sampling.

Keywords: Rendering, Importance Sampling, Deterministic Sampling, Hierarchical Representation, Environment Mapping, Digital Halftoning, Blue Noise, Lookup Table-based Techniques, Penrose Tiling, Fibonacci Number System.

1 Introduction

Sampling is ubiquitous in computer graphics. Many researchers have studied how the properties of sampling may affect the quality of the achieved results in applications such as ray tracing, Monte Carlo path tracing, motion blur, geometry processing, digital halftoning, etc. Nowadays, it is generally accepted that isotropic two-dimensional sampling with blue noise Fourier spectrum is well suited for a large range of applications – see [Cook 1986], [Ulichney 1988], [Shirley 1991], [Mitchell 1991], [McCool and Fiume 1992], [Glassner 1995], [Hiller et al. 2001], [Kollig and Keller 2002], [Kollig and Keller 2003].

Often, these graphics applications need distributions of samples proportional to an importance that results from a prior treatment (e.g., BRDF of a surface, distribution of light energy, and geometrical properties). The problem of 2D importance sampling with blue noise can be stated as follows:

- **Given** the importance density I on a domain D , as an analytical function or in the form of an array of discrete values. Without loss of generality, I can be normalized in such a way that $0 \leq I(x,y) \leq 1 \quad \forall (x,y) \in D$.
- **Find** a set of discrete samples, whose local density of samples (the number of samples per unit area, calculated locally) is proportional to the importance density I , and whose Fourier spectrum exhibits the following properties: (a) low angular anisotropy, and (b) characteristic blue noise profile of the radial component, i.e., a low-magnitude disk around the DC term, a high-magnitude annulus that corresponds to the mean distance between the samples, and a surrounding medium-magnitude background exterior to the annulus (see more details in [Ulichney 1987], [Hiller et al. 2001]).

Many different techniques have been developed in order to solve this problem. Some of them, known as relaxation techniques, can produce solutions of remarkable quality. In particular, Lloyd's relaxation [Lloyd 1983] and its variants lead to centroidal Voronoi tessellations [Du et al. 1999]. Unfortunately, the price paid for this

quality is high: relaxation techniques are fundamentally slow because they have to solve, often iteratively, the problem of neighborhood determination of each point with respect to all others. Even the most advanced and optimized implementations remain slow. The hardware-assisted implementation of Lloyd's relaxation [Hoff et al. 1999] is faster but is limited by the resolution of the frame buffer. Some techniques use a form of stochastic sampling (dart throwing), such as the method proposed in [McCool and Fiume 1992], where random points are added or rejected according to the proximity to previous points. Due to the low convergence rate of these methods, their running times are at best in the same order as Lloyd's. These methods are strictly descending and can be very sensitive to the initial point set.

Other approaches employed in digital halftoning, known as error-diffusion techniques (see [Ulichney 1987], [Ostromoukhov 2001], [Zhou and Fang 2003]), are considerably faster because only a very limited neighborhood of each point is examined. An example of efficient usage of error-diffusion in geometry processing has been exploited in [Alliez et al. 2002]. The main drawback of error-diffusion is the discrete nature of the elements on which it operates: they must be rectangular tiles with fixed spatial resolution. This limits considerably the use of error-diffusion as a general-purpose sampling technique for computer graphics, where multi-resolution sampling is often needed. This drawback has been explicitly mentioned in [Surazhsky et al. 2003] where Lloyd's relaxation was preferred to error-diffusion.

Another fast sampling technique that could compare to ours in terms of running times would be to use a cumulative density function (CDF), generated from the probability density, and to sample it with a stratified Monte-Carlo technique. Although such an approach can generate points that reflect the required local density, they do not follow the desired blue-noise distribution, as shown in Figure 13. Recently, [Secord et al. 2002] have used a similar approach, using some well-known low-discrepancy sequences such as Halton and Sobol sequences (see [Niederreiter 1992] combined with CDF, in order to distribute graphics primitives at interactive rates, in an NPR context. Although this deterministic approach is very promising, the convincing multi-purpose results have yet to be seen (see Figure 13).

In this paper, we introduce a novel Penrose tiling-based importance sampling technique that presents certain advantages over existing techniques. It belongs to the family of point sampling, that is, each point is processed independently of other points. The treatment for each point is simple and computationally inexpensive, which guarantees the very high speed of our algorithm. Moreover, thanks to off-line optimization and to a specially designed lookup table, the quality of the sampling is high, approaching the quality of centroidal Voronoi tessellations. The size of the lookup table is reasonably small (typically, less than 1K of data). No data-dependent pre-calculation is needed. Our technique is multi-resolution and can be successfully applied on high dynamic range images (we illustrate this in Section 5).

The rest of the paper is organized as follows. In Section 2, we recall some historical facts and properties of Penrose tiling. In Section 3 we describe the core of our sampling system. In Section 4, we enrich the basic technique with an advanced relaxation that produces an almost perfect blue noise Fourier spectrum at all importance levels. In Section 5, we apply, as a case study, the proposed technique to importance-based environment mapping. Finally, in Sections 6 and 7, we discuss future work and draw some conclusions.

2 Penrose Tiling

The history of Penrose tiling is fascinating. It goes back to the work of Johannes Kepler, a 17th-century astronomer and mathematician.

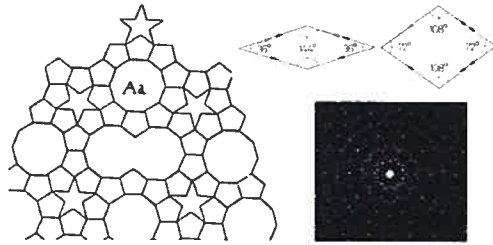


Figure 2: Left: Kepler's drawing from *Harmonice Mundi* published in 1619. This tiling inspired Roger Penrose to discover his aperiodic tiling composed of two marked rhombs (top right). Tiles from his original 1979 article contain arrows as matching rules that force the aperiodicity of the tiling. Bottom right: Optical Fourier transform of the vertices of Penrose tiling obtained in 1982 by Alan Mackay.

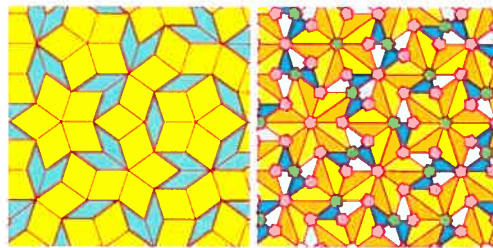


Figure 3: Left: Original Penrose tiling with two kinds of rhombs. Right: An alternative representation of the same tiling where the rhombs are split in two halves, and the pentagons of two kinds are placed at the vertices of the original tiling.

In his book *Harmonice Mundi*, he published an atlas of various tilings with regular polygons. One of them, shown in Figure 2 (left), excited the imagination of many mathematicians over a long period. Is it possible to tile the plane only with regular pentagons, decagons, and five-pointed stars? According to Kepler's drawing, it was possible if one permitted also strange peanut-shaped figures ("monsters"), such as the one visible underneath the label "Aa".

In the early 1970s, a modern physicist and mathematician, Roger Penrose, was mesmerized by Kepler's drawing. He modified it in such a way that he was able to tile the plane non-periodically with a similar set of tiles. And he did much more: he found that introducing special matching rules such as marks on the edges of the tiles will preclude any periodic arrangements of the tiles. Still, the tiling shows a clearly identifiable local order. This tiling belongs to the family of *aperiodic structures*, i.e., structures whose non-periodicity is forced by the matching rules. Penrose published a first account of his discovery in [Penrose 1974]. Later, Penrose published a paper where he presented three different but tightly related aperiodic tiling systems with matching rules [Penrose 1979]. One of them, shown in Figure 2 (top right), has only two extremely simple shapes, two different rhombs with matching rules. In 1977, Martin Gardner published in his column in *Scientific American*, an enthusiastic account of Penrose's discovery [Gardner 1977]. After Gardner's publication, Penrose tiling became well-known to a large number of mathematicians, physicists, and chemists.

The Penrose tiling with rhombs shown in Figure 2 (top right) and in Figure 3 (left) has attracted special attention, due to its simplicity. In the early 1980s, Alan Mackay generalized the Penrose tiling to

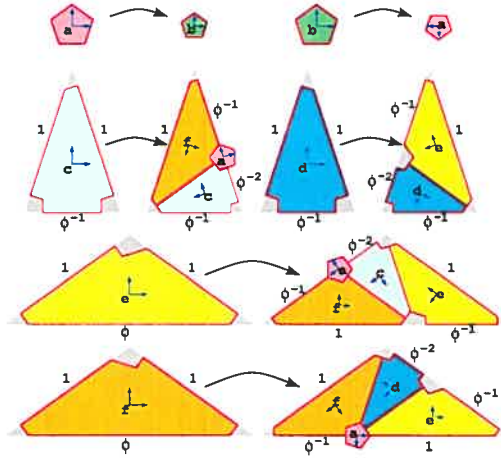


Figure 4: Subdivision rules for modified Penrose tiling, as shown in Figure 3 (right). The Golden Ratio $\phi = \frac{1+\sqrt{5}}{2} \approx 1.61803$. Pairs of orthogonal vectors form the basis for each tile.

three dimensions. He also performed an optical Fourier transform of the pattern of holes perforated at the vertices of Penrose tiling (see Figure 2 (bottom right), reproduced from [Mackay 1982]). This picture caused a sensation. In fact, it looks like a Bragg diffraction pattern, but it clearly violates a well-established principle, the so-called crystallographic restriction, which states that a diffraction pattern of a crystal may have only a two-, three-, four-, or six-fold rotational symmetry. The pattern obtained by Mackay in fact had ten-fold rotational symmetry! A new science of quasicrystallography was born. In 1984, Shechtman and his colleagues synthesized a new matter, a quasicrystal, that had a diffraction pattern close to that predicted by Mackay. It was the first time in human history that a new kind of matter was predicted and analyzed before being physically synthesized in the laboratory. Since then, hundreds of papers on quasicrystals have been published. A corpus of fundamental papers on the physics and mathematics of quasicrystals can be found in [Steinhardt and Ostlund 1987]. Grünbaum and Shephard, in their theory of tiling bible [Grünbaum and Shephard 1986], devote a whole section to Penrose tiling, where they provide a detailed analysis.

Alternatively, Penrose tiling with rhombs can be represented as a tiling with six shapes: both “fat” and “thin” rhombs are split into two triangular halves, as shown in Figure 3 (right). In addition, regular pentagons of two kinds, which we also call “sampling tiles” are placed at each vertex of the original Penrose rhombs. The pentagons play the role of matching rules that enforce aperiodicity. The tiling can be achieved by applying on each tile the subdivision process as shown in Figure 4 (Grünbaum and Shephard call this subdivision “inflations”). Geometrical proportions for all sides during the subdivision process are shown in Figure 4. Note that the size of the pentagons with respect to the triangles does not matter. Without loss of generality the pentagons can be taken to be infinitesimal, and the half-rhombs are triangular. The positions and orientations of tiles, schematically represented in Figure 4 by pairs of orthogonal vectors that form the basis for each tile, are important for our construction.

Penrose tiling has attracted our attention for several reasons. First, it is obvious from a glance at Mackay’s optical Fourier transform

shown in Figure 2 (bottom right) that it is surprisingly close to the blue noise spectrum, a goal we fixed for our sampling system. In fact, annuli of spectral peaks around the DC term are clearly visible. Second, it can be easily observed that the pentagons obtained at one level of subdivision are enriched by the pentagons of the next level, which are placed between the pentagons of the previous level (see Figure 5).

Although another famous aperiodic tiling, Wang tiling, has recently been successfully exploited in computer graphics for the generation of Poisson distributions of points [Hiller et al. 2001], [Cohen et al. 2003], Penrose tiling has been used only in the context of visualization of decorative properties of the tiling (see for example [Glassner 1998]).

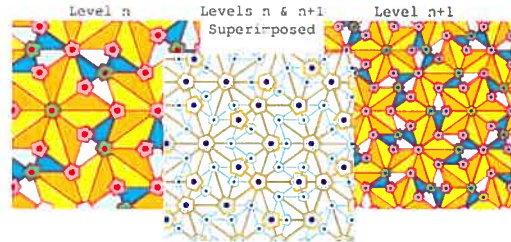


Figure 5: Two consecutive subdivision levels of Penrose tiling. Red dots mark sampling tiles of level n , blue dots – that of level $n + 1$.

3 Two-dimensional Penrose-based Hierarchical Importance Sampling

Let us consider the Penrose subdivision process shown in Figure 4 as a recursive subdivision process. A special binary code called F-code is assigned to each tile. This subdivision process can be described by the following production rules:

$$\mathcal{P}_{Penrose} := \begin{cases} a_* \mapsto \{b_{00*}\} \\ b_* \mapsto \{a_{00*}\} \\ c_* \mapsto \{f_{00*}, c_{10*}, a_{10*}\} \\ d_* \mapsto \{e_{00*}, d_{10*}\} \\ e_* \mapsto \{f_{00*}, c_{10*}, e_{01*}, a_{10*}\} \\ f_* \mapsto \{e_{00*}, d_{10*}, f_{01*}, a_{01*}\} \end{cases} \quad (1)$$

where x_y means a tile of type x having F-code y . The symbol ‘*’ replaces the F-code of a tile before subdivision. Each subdivision left-concatenates two symbols to the current F-code. Thus, after n subdivisions, the F-code will have the length of $2n$ symbols. F-codes can be interpreted as integer numbers in the Fibonacci number system as described in [Knuth 1997] and [Graham et al. 1994]. Appendix B provides some basic facts about the Fibonacci number system, together with the pseudo-code of the routine FIBTOTDECIMAL that converts F-codes to the conventional decimal representation.

Figure 6 shows the first three subdivisions applied to a pair of tiles of type ‘e’ and ‘f’ (top left).

Three important observations can be made:

- Decimal numbers that correspond to F-codes assigned to pentagonal sampling tiles of type ‘a’ and ‘b’ are all in the range $[1..(F_{2(n-1)} - 1)]$, where n is the subdivision level of the initial tiles, and F_i are the Fibonacci numbers.
- Successive subdivisions enrich sampling points obtained with the previous subdivisions, putting new sampling points in be-

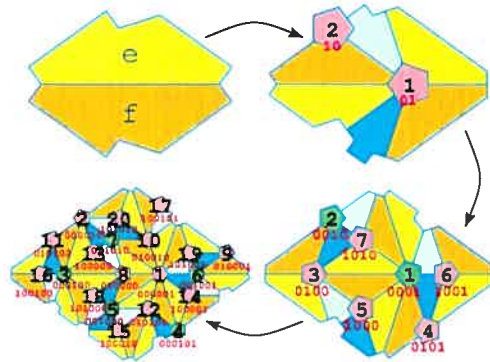


Figure 6: Three subdivisions according to the production rules (1). Only the F-codes of pentagonal sampling tiles are shown (in red), together with corresponding decimal values (in black).

- Decimal numbers already assigned at a subdivision level n will remain at the same positions during all further subdivisions. The principle of this distribution is similar to that of Bayer's dispersed-dot dither [Bayer 1973], [Foley et al. 1990] or that of rotated dispersed-dot dither [Ostromoukhov et al. 1994].

This allows us to build an adaptive importance sampling system based on the Penrose subdivision system with the production rules (1). Our adaptive importance sampling system is simple. First, we cover the area of interest, where the importance is defined, with a pair of tiles of type 'e' and 'f', as shown in Figure 6 (top left). Then, we apply the recursive subdivision process according to the production rules (1). We stop subdividing when the required local subdivision level κ is reached. In this case, we output the center of the 'a' and 'b' type tiles, if the local importance is greater than the decimal value of the F-code of the current tile. Pseudo-code for this algorithm is shown in Appendix A.

Importance density may be scaled by a factor mag , constant for the entire importance density image, in order to obtain the desired number of points. This effect is illustrated in the companion video¹. The required local level of subdivision κ can be determined as

$$\kappa = \lceil \log_{\phi^2} \max_{tile} (mag \cdot I(x,y)) \rceil, \quad (2)$$

where $\lceil \cdot \rceil$ is the usual notation for ceiling, $I(x,y)$ is the importance value at position (x,y) , and $\phi = \frac{1+\sqrt{5}}{2}$ is the *Golden Ratio*. The factor \log_{ϕ^2} can be explained as the factor of self-similarity of Penrose tiling. In fact, from one level of subdivision to the next, the area of Penrose tiles diminishes by factor ϕ^2 . The value $\max_{tile}(\cdot)$ can be achieved with standard scan-conversion on the triangle, for tiles of type 'c', 'd', 'e', and 'f' (no scan-conversion is needed for tiles of type 'a' and 'b' that are supposed to be infinitesimal). This scan-conversion is opened to possible optimization. If less precision is required but speed is capital, the importance can be tested only at a few points within the tile.

4 Lookup Table-based Relaxation

To improve the spatial distribution of the sampling points, we create a table of corrective vectors, which is used at run-time to relocate the sampling points. These corrective vectors, expressed in terms of

¹Siggraph 2004 Full Conference DVD-ROM; also available on the web site of the first author: www.iro.umontreal.ca/~ostrom/ImportanceSampling

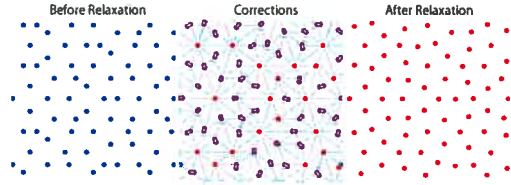


Figure 7: Lloyd's relaxation applied on sampling point set of constant importance, produced with our Penrose tiling-based system. Small corrections are shown as yellow lines connecting the centers of uncorrected (blue) and corrected (red) sampling points.

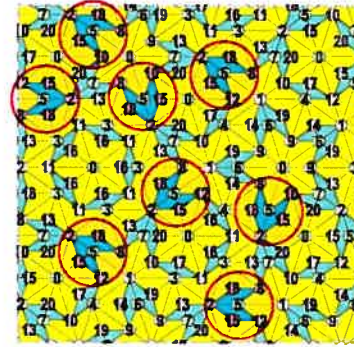


Figure 8: Structural indices i_s obtained by converting the 6 most significant bits of the F-code assigned to each tile to the conventional decimal representation. Only sampling (pentagonal) tiles are considered. In this figure, we use identical colors for tiles of type 'b' and 'c', 'd' and 'e', to better visually identify the structure. Notice how the local neighborhoods around tiles with the same label are similar, after rotation. Highlighted are the neighborhoods around pentagons labeled with $i_s = 5$.

orthogonal basis proper to each tile, will have the effect of "relaxing" the point distribution. See Figure 7 and the companion video¹. Unfortunately, even though there are only two different sampling tiles, the fact that Penrose tiles fill the plane aperiodically makes it impossible to account for every possible correction vector. Nevertheless, the self-similar nature of the tiling can be harnessed to obtain a limited number of corrective vectors. To accomplish this, we relabel the sampling tiles with what we call a "structural index", i_s , which is calculated from the first six bits of their F-code (see Figure 8). This gives a total of 21 different labels (the maximum value encoded with the F-system over 6 bits). Each of these 21 labels has a corresponding corrective vector. This six-bit structural indexing has been found experimentally.

Because these corrective vectors have to be representative of any importance density function, we must optimize them with regards to different importance values. We chose to optimize the vectors over n importance values that are represented by what we call the "importance index", i_v , which is calculated as follows:

$$i_v = \lfloor n \cdot \psi(mag \cdot I(x,y)) \rfloor, \quad (3)$$

where $\lfloor \cdot \rfloor$ is the usual notation for the floor, $I(x,y)$ is the importance value at position (x,y) and $\psi(\cdot)$ maps a real number onto the interval $[0..1]$. We found that with $n = 8$ we get rather smooth gradations across importance values.

The resulting corrective vectors optimized over each importance

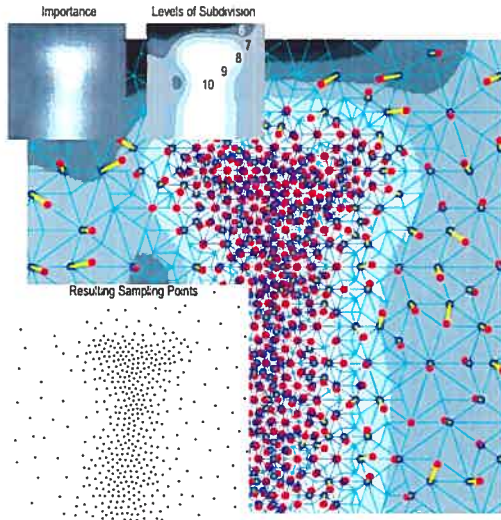


Figure 9: A typical importance map sampled with our system. Blue dots indicate the centers of the sampling tiles. Yellow lines show lookup table-based corrections applied on sampling points. Red dots show the corrected sampling points. Please note seamless transitions between zones of different levels of subdivision.

value are stored in a 8×21 lookup table where the first index is addressed by the importance index i_v , and the second dimension is addressed by the structural index i_s .

The procedure used to generate the lookup table is the following.

First, we initialize the lookup table with zero vectors. Then, for each importance index, i_v , we apply the following process.

1. With our sampling system, create a large patch of sampling points that correspond to the current importance, using the latest version of the lookup table for corrections as will be explained below.
2. Apply Lloyd's relaxation on this set. Be careful to make peripheral points immovable. This is needed because the patch is finite. See Figure 7.
3. For each sampling point, calculate the difference between the uncorrected position and the relaxed position.
4. Calculate the mean value of correction vectors, for all sampling tiles with the same structural index i_s . Store the results in the table at position (i_v, i_s) .
5. Repeat steps (1)-(4) until convergence is attained (typically 5 to 10 iterations).

Considering the lookup table as a vector field, a low-pass filter is then applied to the vectors across importance indices. Then, the whole process is repeated until convergence is attained (typically 5 to 10 iterations). Thus, relaxation and low-pass filtering are applied alternatively.

At the end of the optimization, the corrected points closely match those obtained by true Lloyd relaxation, for all importance levels. The low-pass filtering between each iteration ensures that the points will be distributed adequately over gradients in non-constant importance density functions.

Figures 9, 13 (top), 10, and 11 (top right) illustrate the results achieved with our system. Please note how our technique captures nuances of importance in all subranges of the dynamic range.

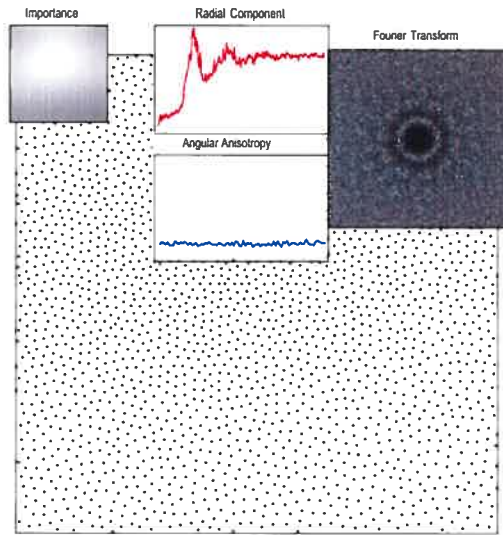


Figure 10: Fourier amplitude spectrum (DFT) of a smooth importance density image sampled with 2600 points, together with its radial component and angular anisotropy of the spectrum. Please notice the typical blue noise profile of the radial component.

Several examples of lookup tables of various sizes can be found on¹. Please notice that the lookup table of size 8×21 described here is compact, yet still gives satisfactory results. Larger size lookup tables are applicable as well.

5 Case Study: Environment Map Sampling

One of many applications of our technique in computer graphics is the sampling of HDR environment maps. The idea is to reduce the environment map to a relatively small number of point light sources, thus speeding up the integration of the incoming illumination. We compare our sampling system with those used in recent incarnations of the above idea, notably *Structured Importance Sampling* [Agarwal et al. 2003] and *LightGen* [Cohen and Debevec 2001]. See the companion video¹.

This problem essentially reduces to the k-centers problem, to which there is no known polynomial-time solution. To achieve a fast solution, an approximation must be used. [Agarwal et al. 2003] use the Hochbaum-Shmoys algorithm, and [Cohen and Debevec 2001] use k-means clustering; both are iterative searches. In [Kollig and Keller 2003], a modified Lloyd's relaxation scheme is used to distribute the sampling points, which is also an iterative process. In our system, the sampling points are deterministic and the lookup table is pre-calculated; only a thresholding operation must be performed during the sampling. Thus, we can obtain an empirical linear time approximation (time on a 2.6 GHz P4 processor):

| | | | | | | | |
|---------------|-----|-----|-----|-----|-----|------|------|
| No. of points | 236 | 343 | 455 | 690 | 930 | 1847 | 3006 |
| Time in ms | 6 | 9 | 12 | 17 | 22 | 42 | 64 |

The resulting sample distribution compares well with the other techniques. The running times, though, are several orders of magnitude lower. In order to obtain running times similar to our system,

¹Siggraph 2004 Full Conference DVD-ROM; also available on the web site of the first author: www.iro.umontreal.ca/~ostrom/ImportanceSampling



Figure 11: Galileo's Tomb environment map, sampled with left: *LightGen*, center: *Structured Importance Sampling*, and right: our system. The map size is 1024×512 , the number of samples is 300. On a 2.6 GHz P4, sampling times were: left: 45 minutes, center: 25 seconds, right: 8 milliseconds.

one could use the cumulative importance density function sampled with a stratified Monte-Carlo sampling pattern or low-discrepancy sequences, as shown in Figure 13. As mentioned before, the resulting sample points do not exhibit a blue-noise distribution. Whether a blue-noise distribution of the lights is better in this context can be debated, but that is beyond the scope of this paper.

With the relative speed of our system, the bottleneck quickly becomes the rendering process. With a rendering system that could handle a few hundred lights in real time, it would be possible to simulate distant illumination with dynamic maps at interactive frame-rates, using our sampling system.

See Figures 1 and 11 and the companion video.

6 Discussion and Future Work

Several important notions are given in this paper without formal proof. For example, the property of uniformity of point distribution introduced in Section 3 must be thoroughly studied and adequately presented. This is a considerable work that goes far beyond the scope of this paper. Adequate mathematical tools for this study should be developed.

As we mentioned in Section 2, three-dimensional extension of Penrose tiling was proposed by Mackay more than twenty years ago; it played an important role in the discovery of quasicrystals. Consequently, one may expect to build a 3D construction, similar to our construction in Section 3, which would result in isotropic 3D point distribution, modulated by 3D importance density functions.

Such hierarchical constructions would be useful in various computer graphics applications. Moreover, a multi-dimensional variant of our construction may exist as well. It would be helpful in various light transport calculations where multi-dimensional importance density functions must be sampled (see also [Veach 1997], [Kollig and Keller 2001], [Kollig and Keller 2002]).

Sampling point sets produced with our method are not perfectly isotropic. One possible way to improve our results would be studying different available aperiodic tilings with the methodology introduced in this paper. Among the known aperiodic tilings that share various properties with Penrose tiling, we are considering Ammann's octagonal tiling [Grünbaum and Shephard 1986], Socolar's dodecagonal tiling [Socolar 1989], and Wang tiling [Grünbaum and Shephard 1986], [Hiller et al. 2001], [Cohen et al. 2003].

Another way to improve isotropy would be to use our method as a starting point for some other methods, such as weighted Lloyd's relaxation, which will then converge in much fewer iterations.

7 Conclusions

Let us summarize the contributions of this paper.

First, we have proposed an original method for sequentially numbering all vertices of Penrose tiling, based on the Fibonacci number system. The Penrose tiling is used as an underlying structure for a recursive subdivision. The numbers associated with the tiles are used as thresholds in the sampling process.

Second, we improve the above system with corrective vectors to en-

sure blue noise properties of the sampling point distribution. This is achieved through an off-line Lloyd relaxation scheme. The corrective vectors obtained in the optimization are stored in a two-dimensional lookup table.

Our technique is very fast because the required processing per sampling point is simple. The processing time grows linearly with the number of sampling points. Typical processing time for sampling a scene with thousands of sampling points can be measured in milliseconds. Because each tile is processed independently, the proposed system is parallelizable and therefore can be efficiently implemented with hardware.

Because of its speed, simplicity, and multi-resolution properties combined with good quality of point distribution, our importance sampling technique may be applied in a large variety of graphical applications.

8 Acknowledgements

For the first author, this paper represents a culmination of a very long-term research project that goes back to his Ph.D. thesis some ten years ago. We would like to express our gratitude to many people who were involved at different degrees in discussions related to the present paper. Thanks to Roger Hersch, Isaac Amidror, David Salesin, Branko Grünbaum, Przemyslaw Prusinkiewicz, Craig Kaplan, Douglas Zongker, Eric Stollnitz, Daniel Wood, Julie Dorsey, Pierre Alliez, Mathieu Desbrun, Alexander Keller, Jiri Patera, Mark Grundland, Pierre McKenzie, Jean Vaucher, and Neil Stewart. We would like to thank the anonymous SIGGRAPH reviewers for their constructive and pertinent comments. Very special thanks to Frédo Durand and Pierre Poulin for their unconditional support and help. Finally, we are indebted to our families for their support during the hard days of the paper submission.

References

- AGARWAL, S., RAMAMOORTHY, R., BELONGIE, S., AND JENSEN, H. 2003. Structured importance sampling of environment maps. *ACM Trans. on Graphics* 22, 3 (July), 605–612.
- ALLIEZ, P., MEYER, M., AND DESBRUN, M. 2002. Interactive geometry remeshing. *ACM Trans. on Graphics* 21, 3, 347–354.
- BAYER, B. 1973. An optimum method for two-level rendition of continuous-tone pictures. In *IEEE Int. Conf. on Communications*, 11–15.
- COHEN, J., AND DEBEVEC, P. 2001. *LightGen, HDRShop plugin*. <http://www.ict.usc.edu/~jcohen/lightgen/lightgen.html>.
- COHEN, M., SHADE, J., HILLER, S., AND DEUSSEN, O. 2003. Wang tiles for image and texture generation. *ACM Trans. on Graphics* 22, 3 (July), 287–294.
- COOK, R. 1986. Stochastic sampling in computer graphics. *ACM Trans. on Graphics* 5, 1 (Jan.), 51–72.
- DEBEVEC, P. 1998. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proc. SIGGRAPH '98*, 189–198.
- DU, Q., FABER, V., AND GUNZBURGER, M. 1999. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review* 41, 4 (Dec.), 637–676.
- FOLEY, J., VAN DAM, A., FEINER, S., AND HUGHES, J. 1990. *Computer Graphics, Principles and Practice*, 2nd ed. Addison-Wesley.
- GARDNER, M. 1977. Extraordinary nonperiodic tiling that enriches the theory of tiles. *Scientific American* 236, 110–121.
- GLASSNER, A. 1995. *Principles of Digital Image Synthesis*. Morgan Kaufmann.
- GLASSNER, A. 1998. Andrew Glassner's notebook: Penrose tiling. *IEEE Computer Graphics & Applications* 18, 4, 78–86.
- GRAHAM, R., KNUTH, D., AND PATASHNIK, O. 1994. *Concrete Mathematics: a Foundation for Computer Science*, 2nd ed. Chapter 6.6. Addison-Wesley.
- GRÜNBAUM, B., AND SHEPHARD, G. 1986. *Tilings and Patterns*. W.H. Freeman.
- HILLER, S., DEUSSEN, O., AND KELLER, A. 2001. Tiled blue noise samples. In *Proc. Vision Modeling and Visualization*, 265–272.
- HOFF, K., CULVER, T., KEYSER, J., LIN, M., AND MANOCHA, D. 1999. Fast computation of generalized voronoi diagrams using graphics hardware. In *Proc. SIGGRAPH '99*, 277–286.
- KNUTH, D. 1997. *The Art of Computer Programming, Volume 1, Fundamental Algorithms*, 3rd ed. page 86. Addison-Wesley.
- KOLLIG, T., AND KELLER, A. 2001. Efficient bidirectional path tracing by randomized quasi-monte carlo integration. *Niederreiter, K. Fang, and F. Hickernell, Eds., Monte Carlo and Quasi-Monte Carlo Methods 2000*, 290–305.
- KOLLIG, T., AND KELLER, A. 2002. Efficient multidimensional sampling. *Computer Graphics Forum* 21, 3, 557–564.
- KOLLIG, T., AND KELLER, A. 2003. Efficient illumination by high dynamic range images. In *Eurographics Symposium on Rendering: 14th Eurographics Workshop on Rendering*, 45–51.
- LLOYD, S. 1983. An optimization approach to relaxation labeling algorithms. *Image and Vision Computing* 1, 2, 85–91.
- MACKAY, A. 1982. Crystallography and the Penrose pattern. *Physica* 114A, 609–613.
- MCCOOL, M., AND FIUME, E. 1992. Hierarchical poisson disk sampling distributions. In *Proc. Graphics Interface '92*, 94–105.
- MITCHELL, D. 1991. Spectrally optimal sampling for distributed ray tracing. In *Proc. SIGGRAPH '91*, vol. 25, 157–164.
- NIEDERREITER, H. 1992. *Random Number Generation and Quasi-Monte-Carlo Methods*. Soc. for Industrial and Applied Mathematics.
- OSTROMOUKHOV, V., HERSCH, R., AND AMIDROR, I. 1994. Rotated dispersion dither: a new technique for digital halftoning. In *Proc. SIGGRAPH '94*, 123–130.
- OSTROMOUKHOV, V. 2001. A simple and efficient error-diffusion algorithm. In *Proc. SIGGRAPH 2001*, 567–572.
- PENROSE, R. 1974. The role of aesthetics in pure and applied mathematical research. *Bull. Inst. Math. & its Appns.* 10, 266–271.
- PENROSE, R. 1979. Pentaplexity, a class of non-periodic tilings of the plane. *The Mathematical Intelligencer* 2, 32–37.
- SECORD, A., HEIDRICH, W., AND STREIT, L. 2002. Fast primitive distribution for illustration. In *13th Eurographics Workshop on Rendering*, 215–226.
- SHIRLEY, P. 1991. Discrepancy as a quality measure for sample distributions. In *Proc. Eurographics '91*, 183–194.
- SOCOLAR, J. 1989. Simple octagonal and dodecagonal quasicrystals. *Physical Review B* 39, 10519–10551.
- STEINHARDT, P., AND OSTLUND, S. 1987. *The Physics of Quasicrystals*. World Scientific.
- SURAZHISKY, V., ALLIEZ, P., AND GOTSMAN, C. 2003. Isotropic remeshing of surfaces: a local parameterization approach. In *Proc. of 12th Int. Meshing Roundtable*.
- ULICHNEY, R. 1987. *Digital Halftoning*. MIT Press.
- ULICHNEY, R. A. 1988. Dithering with blue noise. *Proc. of the IEEE* 76, 56–79.
- VEACH, E. 1997. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis. Stanford University.
- ZHOU, B., AND FANG, X. 2003. Improving mid-tone quality of variable-coefficient error diffusion using threshold modulation. *ACM Trans. on Graphics* 22, 3 (July), 437–444.

APPENDIX A: Pseudo-code of the Adaptive Subdivision and Sampling

```

ADAPTIVESAMP(t of type tile)
1   ▷ Structure tile contains the fields:
2   ▷ type: ['a'..'f']
3   ▷ LOS: Level of Subdivision
4   ▷ vertices: center if 'a' or 'b', triangle otherwise
5   ▷ Fcode
6   ▷ mag and importance are global variables
7   local_LOS ← GETMAXLOSWITHINTILE(t)
8   if t.LOS ≥ local_LOS
9     then ▷ Terminal: don't need more subdivisions
10    local_importance ← mag · GETLOCALIMPORTANCE(t)
11    if t.type = ('c' or 'd' or 'e' or 'f')
12      then return ▷ Not a "sampling tile": do nothing
13    if local_importance ≥ FIBOTODECIMAL(t.Fcode)
14      then OUTPUTSAMPLE(t.center)
15      return
16    else ▷ Need more subdivisions
17      {t1, ..., tn} ← SUBDIVIDEUSINGPRODUCTIONRULES(t)
18      return {ADAPTIVESAMP(t1), ..., ADAPTIVESAMP(tn)}
    
```

It is worth mentioning that the routines GETMAXLOSWITHINTILE(*t*) and GETLOCALIMPORTANCE(*t*) play a very important, even crucial role in the algorithm. If, for any reason (e.g., because of a singularity in the importance density), they fail to evaluate the local importance/max importance, it may result in locally erroneous sampling density.

APPENDIX B: ϕ - and F- Number Systems

Details about ϕ - and F- (Fibonacci) number systems can be found in [Knuth 1997] and [Graham et al. 1994].

The ϕ -system is a positional number system in base ϕ , where $\phi = \frac{1+\sqrt{5}}{2}$ is the *Golden Ratio*. Any rational number x can be expressed in this system exactly as in our conventional binary or decimal systems, except that instead of using powers of two or ten, this system employs powers of ϕ . For example, the number $(101.001)_\phi$ in base ϕ is

$$(101.001)_\phi = \phi^2 + \phi^0 + \phi^{-3} \approx 3.8541_{10}$$

The ϕ -system is closely related to the F-system (the abbreviation for Fibonacci system). The F-system is also a positional system. Any integer n can be presented in the F-system as a sum of Fibonacci numbers F_j multiplied by their positional coefficients, which may be 0's or 1's. Thus, a number n can be expressed by its F-code $(b_m b_{m-1} \dots b_3 b_2)_F$:

$$n = (b_m b_{m-1} \dots b_3 b_2)_F \iff n = \sum_{j=2}^m b_j F_j. \quad (4)$$

The first index in the summation is $j = 2$ because of the convention used for Fibonacci numbers F_j :

$$F_0 = 0, F_1 = 1, F_2 = 1, F_3 = 2, F_4 = 3, F_5 = 5, F_6 = 8, F_7 = 13, \dots$$

The representation of numbers is not unique in the F-system, but it becomes unique if the rule of *normal form* is imposed: two adjacent 1's are not permitted. The procedure of conversion from an arbitrary sequence of 0's and 1's to the normal form, along with many other technical details, can be found in [Graham et al. 1994]. Here are the first twelve integers expressed in the F-system in normal form:

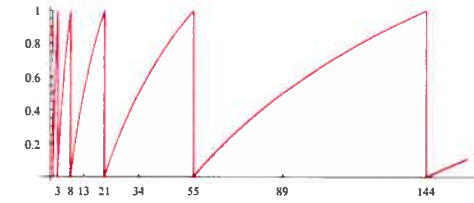


Figure 12: Function $\Psi(x)$. Note that $\Psi(x)$ "jumps" every F_{2j}

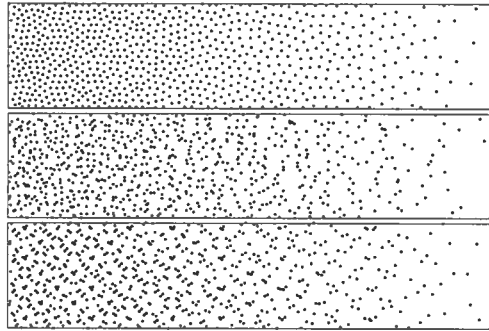


Figure 13: A ramp importance density image sampled with (top) our system and the cumulative importance density function sampled with (middle) a stratified Monte-Carlo pattern, or (bottom) Sobol low-discrepancy sequence.

| | | |
|-----------------------------|-----------------------------|-----------------------------|
| 1 = (00001) _F , | 2 = (00010) _F , | 3 = (00100) _F , |
| 4 = (00101) _F , | 5 = (01000) _F , | 6 = (01001) _F , |
| 7 = (01010) _F , | 8 = (10000) _F , | 9 = (10001) _F , |
| 10 = (10010) _F , | 11 = (10100) _F , | 12 = (10101) _F . |

More examples of interpretation of such sequences of 0's and 1's, which we shall call F-codes, as integer numbers, are shown in Figure 6.

The routine FIBOTODECIMAL converts F-codes to the conventional decimal representation.

```

FIBOTODECIMAL(Fcode)
1  accumulator ← 0
2  for i ← 0 to LENGTH(Fcode) - 1
3    do
4      accumulator ← accumulator + Fcode[i] · Fi-2
5  return accumulator
    
```

Function $\Psi(x)$ that maps a real positive x onto interval $[0..1]$, as shown in Figure 12, is defined as follows:

$$\Psi(x) = (\log_\phi(\sqrt{5} \cdot x)) \bmod 1.$$

It can be easily derived from the well-known Binet's formula

$$F_n = \lfloor \phi^n / \sqrt{5} \rfloor,$$

where $\lfloor \cdot \rfloor$ is the usual notation for the *nint* (Nearest Integer) function.

2.1.3 Discussion

Dans cet article, nous avons présenté une nouvelle technique d'échantillonnage basée sur les tuiles de Penrose. Pour arriver à cette fin, nous avons trouvé une méthode nouvelle de numérotation séquentielle de tous les sommets du jeu de tuiles de Penrose, qui se base sur le système de numérotation de Fibonacci. Les tuiles servent de structure sous-jacente à une subdivision récursive, et les numéros aux sommets servent de seuils dans le processus d'échantillonnage. De plus, nous avons introduit une technique pour améliorer la distribution des points d'échantillonnage, qui utilise un jeu de vecteurs de correction pré-calculés, et stockés dans une table.

Comme le procédé de traitement pour chaque échantillon est très simple, notre technique est très rapide. Les temps de calcul augmentent linéairement avec le nombre de points. Les temps d'exécution sur une fonction typique pour des milliers de points se compte en millisecondes. De plus, comme le traitement de chaque point est indépendant de ses voisins, notre système se paralléliserait bien, et se porterait bien à une implémentation matérielle.

Grâce à sa vitesse, sa simplicité, ses propriétés multi-résolution, ainsi qu'à la bonne qualité de la distribution résultante, notre méthode d'échantillonnage pourrait ouvrir la porte à une grande variété d'applications en infographie.


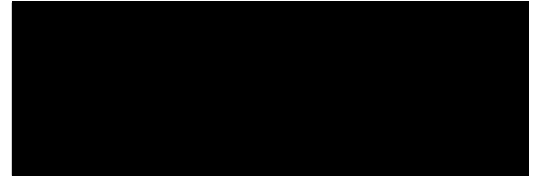
2.1.4 Contribution Personnelle

Les idées génératrices qui ont menées aux découvertes exposées dans cet article proviennent de mon directeur, Victor Ostromoukhov. Elles sont le fruit d'une réflexion approfondie qui s'étale sur plus de dix ans. Mon rôle principal dans ce travail, qui a occupé une proportion considérable des deux années de ma maîtrise, a été d'assister mon directeur dans ses recherches. Mon assistance a consisté en recensement de littérature, en exploration d'applications potentielles, dans l'implémentation d'outils de recherche, et peut-être même au niveau du support moral.

Au niveau de l'article lui-même, j'ai implémenté l'algorithme de manière efficace, de façon à en faire des analyses de performance. Il me revient l'élaboration de l'étude de cas qui a servi à illustrer le potentiel du système d'échantillonnage, ainsi que la rédaction de la section de l'article qui y est dédiée. J'ai aussi monté la vidéo qui y est mentionnée, et qui a fait partie intégrante de la soumission. Finalement, j'ai eu l'honneur de présenter ce travail à SIGGRAPH 2004.

2.1.5 Droits de publication

Par leur signature ci-dessous, chaque coauteur de la publication ci-jointe, *Fast Hierarchical Importance Sampling with Blue Noise Properties* [ODJ04], accorde sa permission explicite pour que l'article soit inclu tel quel dans le présent mémoire. Aussi, il atteste que ma contribution à l'article est effectivement telle que décrite dans le présent mémoire.



Pierre-Marc Jodoin

Selon l'article 2.5 de la politique de droits d'auteurs de l'ACM, l'oeuvre en question peut être incluse dans le présent mémoire, à condition que la notice suivante soit affichée :

“SIGGRAPH ACM, 2004. This is the author’s version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in Proc. SIGGRAPH, Vol. 23, no. 3, 2004”

(http://www.acm.org/pubs/copyright_policy)

2.2 *Fast Triangulated Importance Sampled Point Sets*

Ce deuxième article n'a pas encore été soumis pour publication. Les auteurs sont Charles Donohue et Victor Ostromoukhov. L'article représente une continuation logique du premier article, dans laquelle la problématique est étendue pour inclure la connectivité des points d'échantillonnage, au sens de la triangulation de Delaunay.

2.2.1 Résumé

Cet article présente une extension au système d'échantillonnage proposé dans [ODJ04], qui génère une bonne distribution d'échantillons à partir d'une fonction de densité d'importance. Notre extension consiste à générer efficacement une bonne triangulation des points d'échantillonnage résultants, au sens de Delaunay. Contrairement aux méthodes générales de triangulation de Delaunay, notre méthode exploite le jeu de subdivisions de tuiles triangulaires utilisé par ce système, afin d'en extraire une accélération considérable. Nous proposons un algorithme rapide, qui s'exécute en temps linéaire par rapport au nombre de points générés, en pire cas, ce qui ne peut être accompli par aucun autre algorithme de triangulation de Delaunay connu. Plusieurs domaines en infographie pourraient profiter d'un tel système d'échantillonnage associé à notre algorithme de triangulation, comme le rendu de terrain, le traitement de géométrie 3D et la compression d'images.

Mots Clés : Échantillonnage d'importance, Triangulation Delaunay, Bruit bleu, Tuiles de Penrose.

2.2.2 L'Article

Fast Triangulated Importance Sampled Point Sets

Charles Donohue
University of Montreal

Victor Ostromoukhov
University of Montreal

Abstract

This paper presents an extension to the importance sampling system proposed in [ODJ04], which generates a good sampling pattern given an importance density function in 2D. Our extension consists in efficiently generating a Delaunay triangulation of the resulting sampling points. As opposed to general triangulation algorithms, our method harnesses the triangle subdivision scheme used in the above mentioned sampling system, in order to obtain a considerable speedup. We propose a fast algorithm that runs in worst-case linear time with regards to the desired number of points, something that cannot be achieved with any known Delaunay triangulation algorithm. There are many areas in computer graphics that can benefit from such a sampling system in association with this triangulation algorithm, such as terrain rendering, 3D geometry processing, and image compression.

Keywords: Importance Sampling, Delaunay Triangulation, Blue Noise, Penrose Tiling.

1 Introduction

In the paper [ODJ04], a novel method is proposed to generate a set of discrete sampling points, given a 2D importance density function. The points generated exhibit a local blue-noise distribution, which roughly means that the points do not have alignments or principal directions, and that they are at a minimal distance with respect to each other. Since this system is founded on the Penrose tiles, and because of its ties with quasicrystallography, this system shall henceforth be named *Quasisampler*, for the sake of brevity. The name also stems from the system's potential use in quasi-Monte-Carlo integration. The point sets generated by the *Quasisampler* can be useful in many computer graphics contexts, namely in image compression, where the image could be resampled in an approach similar to [DACB96]; and in remeshing, where the geometry could be resampled within a planar embedding, such as proposed in [AdVDI03].

In certain typical use-cases of the *Quasisampler*, a triangulation of the resulting sampling points must be obtained. Consider for example the case of digital elevation map rendering: an importance function based on the viewing position, the terrain curvature, and other such factors is passed to the *Quasisampler*, which returns a set of points. In order to render the terrain to the display, a triangular mesh must be created, using these points as vertices. It is often desirable that the resulting triangulation avoid narrow triangles ('slivers'). In this context, the Delaunay triangulation, which maximizes the minimum angles of the triangles, can be considered desirable.

There are several algorithms to build a Delaunay triangulation of a set of n points. The more naive approaches, like incremental algorithms such as in [GS83], typically run in $O(n^2)$. There are more efficient ones that can run in $O(n \log n)$ worst case, such as the Shamos and Hoey's divide-and-conquer approach [SH75], Fortune's sweep-line algorithm [For86], or a randomized incremental algorithm augmented with a search structure, such as in [Dev98]. These are all general triangulation algorithms, in the sense that their

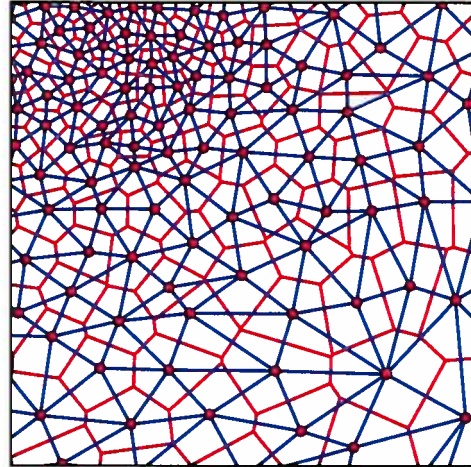


Figure 1: An example of our triangulation shown in blue, with its dual in red. These are respectively equivalent to the Delaunay triangulation and the Voronoi diagram.

input point sets are considered to be provided by an arbitrary black-box.

In this paper, we present an algorithm that can efficiently build Delaunay triangulations of the points generated by the *Quasisampler*. Using a white-box approach, our technique exploits certain properties that arise from the hierarchical subdivision process within the *Quasisampler*, in order to obtain the desired triangulation in $O(n)$. We also show how the algorithm gives fast results in practice, in comparison with other methods.

The paper is organized as follows. The *Quasisampler* system which our method extends is briefly explained in Section 2. Our triangulation algorithm is presented in Section 3, accompanied by results in Section 4. Conclusions and future work follow in Section 5.

2 The Quasisampler

In order to explain how our triangulation algorithm works, we must first make a brief review of the *Quasisampler*, as introduced in [ODJ04]. The three basic steps that the system takes are illustrated in Figure 2.

First, an adaptive tile subdivision scheme is used to build an initial structure. This results in a subdivision tree in which the leaf depths are modulated by the underlying function. The subdivision rules are based on the Penrose tiles [Pen79], but the tiles are all triangular, save for a pair of infinitesimal pentagonal tiles. This makes for a hierarchic structure that can be built only out of triangular subdivisions. Also, the subdivision rules are such that all angles are

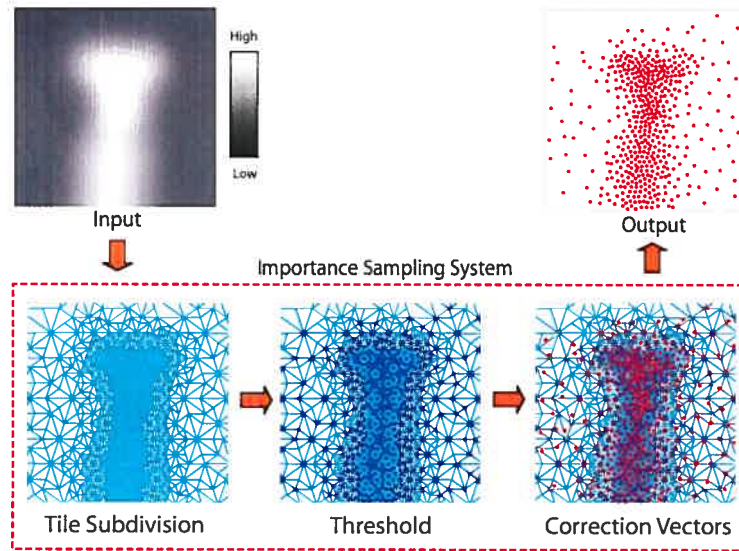


Figure 2: *Quasisampler* outline.

multiples of $\frac{\pi}{10}$, so the trigonometric operations can be tabulated for speed.

Then, the vertices of this structure are numbered, using the Fibonacci number system [Knu97]. The numbers are used as a threshold against the importance function, in order to obtain the desired local density of points. The numbering of the vertices is based on their position in the hierarchy, and the ordinal numbering of the vertices ensures a linear response of point density with regards to the importance values.

Finally, the system applies precalculated correction vectors to the points. This tends to ‘relax’ the points with respect to their neighbors and breaks the inherent structures in the point set. But no proximity queries are needed, as the vectors are applied to each point independently of its neighbors. The correction vectors are stored in a table. It was obtained using an iterative optimization process which involves Lloyd’s relaxation scheme [Llo83].

We are left with a discrete sample distribution, in which the local point density is proportional to the importance density function. Also, the local distributions of points have a blue-noise spectral profile [Uli87; HDK01], which equates to a low anisotropy and no principal directions or alignments. This kind of distribution can be very effective in computer graphics, especially considering the fact that the human visual perception system is very sensitive to such alignments.

Several existing methods can be used to generate point sets with blue noise properties. The techniques that give good quality results, such as Lloyd’s relaxation based techniques, tend to be slow, whereas the faster techniques generally fail to meet the blue-noise requirements. The *Quasisampler* is a fast approximation, yet it is amongst the best in terms of quality. The possibility of generating these good distributions at such a high speed opens the door to many applications which were previously considered unfeasible. But, as is, the system generates a cloud of points, without the connectiv-

ity information that is useful in many applications. We address the connectivity issue in this paper.

3 Triangulation Algorithm

In order to extract connectivity and proximity information from a point set, it is often useful to build a Delaunay triangulation of the set. For a set S of points in the Euclidean plane, the Delaunay triangulation can be defined as the unique triangulation $DT(S)$ of S such that no point in S is inside the circumcircle of any triangle in $DT(S)$. It can also be defined as the dual of the Voronoi diagram of S , as illustrated in Figure 1. The Delaunay triangulation is the target of our algorithm, and it can be built very quickly by harnessing certain intrinsic properties of the *Quasisampler*. The main ideas behind our triangulation algorithm follow.

First, the *Quasisampler* uses a triangular subdivision scheme in order to provide the required density of output points. This structure can be transformed into a proper triangulation by making sure no T-edges remain. Since a strict set of rules is used to build this structure, it is possible to create such a triangulation in linear time, with regards to the number of triangles. Second, not all vertices in the structure will be considered as *active* sampling points because of the thresholding process, so these must be eliminated from the triangulation. The supposition we make at this point is that the connectivity of the triangulation of the final point distribution will be very similar to the connectivity of the structure mentioned above. The *Quasisampler* displaces the points with correction vectors, which can leave us with an invalid topology, but we suppose that this can be corrected in constant time at the local (edge) level. Finally, we can observe that the connectivity information that stems from the original structure, is very close to the Delaunay connectivity after the points are displaced. After a finite number of conditional edge flips, every edge is in the Delaunay set.

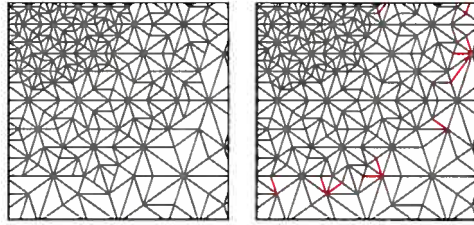


Figure 3: T-edge elimination. Before on the left, after on the right. The edges that have been added are shown in red.

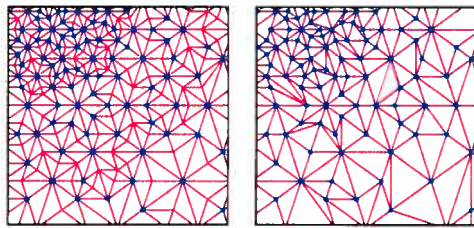


Figure 4: Inactive vertex extraction. Before on the left, after on the right. Blue vertices have passed the thresholding process.

3.1 Our Algorithm

Here is how the algorithm works. The preliminary step is the initialization of the *Quasisampler*, over the given importance density function, as shown in Figure 2. Instead of simply using the output points, we will use the tile subdivision tree structure which the system employs internally.

3.1.1 Base Triangulation

The first step is to create a *valid* triangulation from the underlying sampling structure, meaning there should not be any T-edges. An efficient way of obtaining such a triangulation is to iterate through the tile subdivision tree of the *Quasisampler* in a width-first manner; this has the effect of enforcing the following rule: no two adjacent triangles, which are slated to be subdivided at a subsequent level, will ever be at more than one level of subdivision apart, at any time during the traversal of the tree. This way, whenever a new vertex needs to be added to the current triangulation, it is assured that we only need to split two triangles along their common edge, which is a trivial operation. Also, on the borders of areas at different levels of subdivision, the triangulation remains valid because the triangles on both sides are split. This holds true no matter how many levels of subdivision this border jumps. An example of this operation is shown in Figure 3. So, in order to build our base triangulation, we iterate through the *Quasisampler*'s subdivision tree, successively splitting the edges of the triangles according to the tile's geometry, while making sure that each new vertex holds a reference to the 'sampling' tile from which it has spawned.

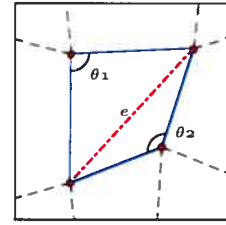


Figure 6: Edge-flip test. If $\theta_1 + \theta_2 > \pi$ then the edge e is flipped.

3.1.2 Inactive Vertex Removal

The triangulation obtained at this point includes every vertex in the tiling. The second step consists in the extraction of the *inactive* vertices from the triangulation. These are the potential sampling points that fail the thresholding step. An example of this step is shown in Figure 4. This process is fairly straightforward; We iterate through all the 'sampling' tiles, and those that have failed the thresholding test are marked for extraction. Finding these vertices in the triangulation is simple, because we have stored reference to the latter. The removal of a vertex from the triangulation involves the re-triangulation of the hole it generates, which can be done with or without enforcing a Delaunay constraint. We have opted for a simple greedy re-triangulation, because the end result is the same, while it is less computationally intensive because we avoid the circumcircle tests. When this greedy approach is chosen, special attention must be brought to collinear points in the re-triangulated area. The original Penrose tiling has alignments in the 10 principal directions, but, depending on the numerical precision chosen for the point representation, some collinear points might appear slightly non-collinear, which can result in triangle slivers. These triangles have an unstable orientation, and can be problematic for the predicates used in further operations on the triangulation. Fortunately, a simple collinearity test avoids these situations, using a numerical precision based on the level of subdivision at the offending point.

This decimated triangulation will serve as the foundation for our final triangulation.

3.1.3 Minimum Angle Edge-Flips

The vertices of the resulting triangulation will need to be displaced by the vectors provided by the sampling system; this can cause an invalid topology at certain vertices, as shown in Figure 5. This leads us to the next step in the algorithm, which is a finite number of conditional edge-flips of the current triangulation.

The process consists in iterating through all the edges of the triangulation, and for each one, calculating the angles at the vertices of each opposing side of both adjacent triangles, as shown in Figure 6. If these angles sum to more than π , then the edge is flipped across the two triangles. This has the effect of maximizing the minimum angles in the triangles, which brings us closer to a Delaunay triangulation, which happens to be minimum angle maximizing. The iteration across all edges represents a single pass, which can leave some edges that are still not in the Delaunay set. In order to obtain a proper Delaunay triangulation, a certain number of these edge flip passes must be made successively. Note that each iteration can be constrained to the local neighborhood of the edges that were flipped in the last iteration, which can greatly reduce the number of tests from one iteration to the next.

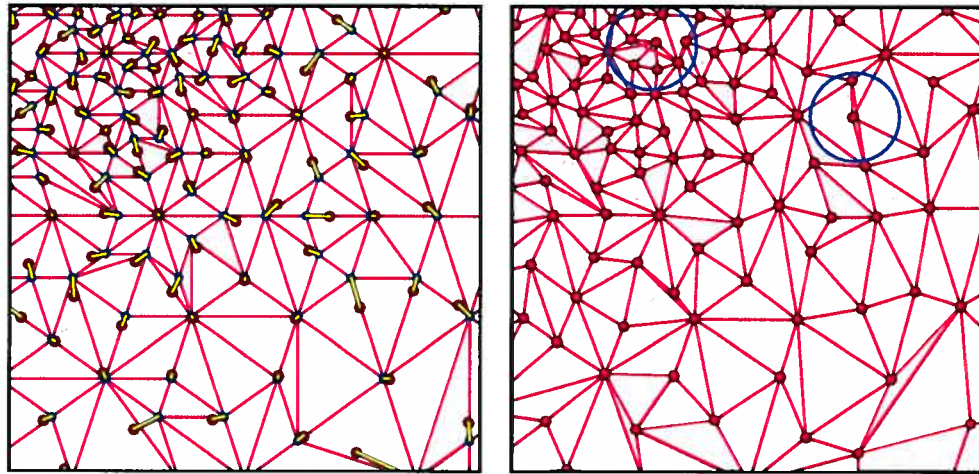


Figure 5: *Quasisampler* correction vectors. Before corrections on the left, after on the right. Notice the invalid topology of the displaced triangulation in certain areas (circled in blue).

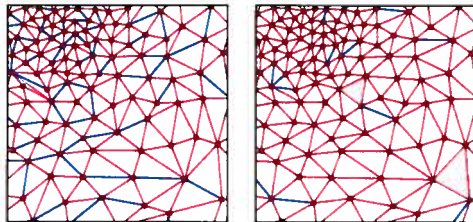


Figure 7: Conditional edge flips. First pass on left, second pass on right. Edges that have flipped are marked in blue.

An example of the results of two passes of this process is shown in Figure 7. The results of the second pass of edge flips are compared with a proper Delaunay triangulation in Figure 8. In this example, only two edges differ from the Delaunay triangulation, and after a third pass (not shown), there is no difference. In many applications, the triangulation resulting from two edge-flip passes is more than adequate, even if it is not a true Delaunay triangulation. In our tests, we have found that the number of offending edges decreases exponentially with each pass, and that with 6 passes, we can reach the bounds of the *Quasisampler*'s precision (using 32 bits for the F-Code). Of course, edges that are ambivalent in the Delaunay sense are not counted.

3.2 Algorithmic Complexity

In order to compare the time complexity of our algorithm with other Delaunay triangulation algorithms, we will make a worst-case analysis, which will give us an order that we can compare with the others.

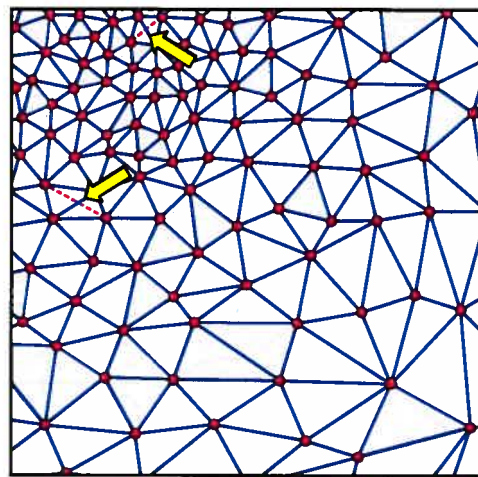


Figure 8: Comparison of our triangulation after two edge-flip passes with a Delaunay triangulation (in blue). Red edges are not in the Delaunay set. After 3 passes, our triangulation is identical to Delaunay's.

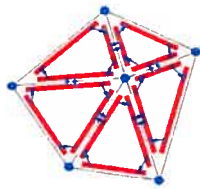


Figure 9: Half-edge data structure (source: www.flipcode.com)

3.2.1 Triangulation Data Structure

In order to help explain the time complexity of our algorithm, here is a quick overview of the triangulation data structures that are common to the compared techniques and ours. The Winged-edge (Baumgart, 1975) or Half-edge (Eastman, 1982) data structures are efficient ways of representing triangle meshes, in that they keep some pointers to local neighbors of the vertices, edges, and faces. This permits efficient adjacency queries into the triangulation, and also accelerates many common operations, such as vertex removal. In the implementation of our algorithm, we use the triangulation data structure provided in the Computational Geometry Algorithms Library (CGAL) [BDTY00], which is based on the Half-edge, as shown in Figure 9. Given that the algorithms used for comparisons are provided by CGAL, they also employ the same data structure. Using this data structure, the following statements about certain operations can be made:

- The insertion of a vertex into the triangulation requires two steps. First, the triangle containing the new point must be found. This operation can be done in $O(n)$, in the worst-case, and can be greatly improved with efficient search structures. Then, generally, the face is split into three triangles, which can be done in $O(1)$.
- The removal of a vertex out of the triangulation involves removing all incident faces and then re-triangulating the resulting hole. This can be done in $O(d^2)$ where d is the degree of the vertex, and this generally reduces to $O(1)$.
- In the case of a Delaunay constraint on the triangulation, the above operations must be followed by a certain number of conditional edge flips, which is proportional to the degree of the vertex. This once again reduces to $O(1)$, thus has no impact on the algorithmic complexity of the previous operations.
- The process of flipping an edge between two adjacent triangles is a trivial operation in $O(1)$ time.

So, to build a Delaunay triangulation of a set of n points using this data structure would take $O(n^2)$ time, assuming a naive face location scheme is used. The search structure proposed by [Dev98] achieves the latter step in $O(\log n)$ worst-case, which results in $O(n \log n)$ for the whole algorithm.

3.2.2 Triangulation Process

Before we start a complexity analysis of our algorithm, we must introduce the following notion. In the *Quasisampler*, the vertices of the base tiling serve as potential sampling points, but a portion of these points fail the thresholding process. Let n be the number of vertices that are ‘active’, and n' be the total number of vertices, then we can state that $n \propto n'$. This is due to the autosimilar behavior of

the Penrose tiling; as the function is scaled to produce more points, the Penrose structure subdivides to deeper levels. These deeper levels are similar to the previous levels, just at a smaller scale. The global behavior remains the same, and the ratio of ‘active’ to ‘inactive’ vertices remains roughly constant.

As previously mentioned, the first step in the algorithm is to build a valid triangulation from the Penrose tiling. This triangulation is built using successive triangle splits, and for each new potential sampling point, a triangle is split along its edge. If a neighboring triangle shares this edge, it is also split. Using the triangulation data structure described before, each split is done in $O(1)$, which implies that the whole triangulation can be built in $O(n')$, or $O(n)$ since $n \propto n'$.

The second step is the extraction of all non-active vertices from the base triangulation. As explained before, the removal of a vertex can be done in $O(d^2)$ time, where d is the degree of the vertex. If we can prove that the vertices have a bound degree, then removal can be considered to be in constant time, meaning that the whole operation can be done in $O(n' - n)$. Again, since $n \propto n'$, this equates to $O(n)$. For vertices that are surrounded by triangles from the same level of subdivision, the worst case that can arise is a 9-connected vertex, as shown in Figure 3, so this is a tight bound. But for vertices on the boundary between different levels of subdivision, this bound is not obvious. Let us examine the worst possible case, which would be starting with two triangles, and then splitting one of them to a high level of subdivision. Since the degrees of all interior vertices are bound, let us only consider the boundary vertices along with the single highly connected vertex of the non-subdivided triangle. All the boundary vertices are 4-connected, and the lone vertex is m -connected, where m is the number of vertices on the edge. So the average degree of these vertices is $\frac{4m+m}{m+1}$, which is bound by 5.

The third step is a finite number of conditional edge-flip passes. First of all, this is an operation on the edges of the triangulation, but this algorithmic analysis is based on the vertex count. So, we use Euler’s theorem, $T - E + V = C$, where T is the number of triangles, E the number of edges, V the number of vertices, and C a constant based on the surface’s genus, with $C = 1$ in the case of a closed polygon. In the creation of the initial triangulation, the addition of a new vertex involves adding either one or two new triangles, so we can say that $T \propto V$, and using Euler’s theorem, $E \propto (V - 1)$. Considering an edge flip as an $O(1)$ operation, a single pass of edge flips is in $O(E)$ time, which then equates to $O(n)$. This step is repeated a finite number of times, and we have found that doing it three times gives us sufficiently good results, and that in some cases a single pass is good enough. As explained before, applying six passes assures a proper triangulation for any case that the *Quasisampler* can generate, because any error would be smaller than what the numerical representation of the points can perceive, assuming 32 bit floating points.

All in all, the algorithm is thus composed of a series of operations that run in $O(n)$, so we can conclude that our algorithm runs in linear time with regards to the number of requested points. This is exciting because the worst-case optimality of the Delaunay triangulation is in $O(n \log n)$. However, we cannot claim to have a general Delaunay triangulation that runs in $O(n)$, because this algorithm runs only on a subset of the general problem, which is limited to the points generated by the *Quasisampler*.

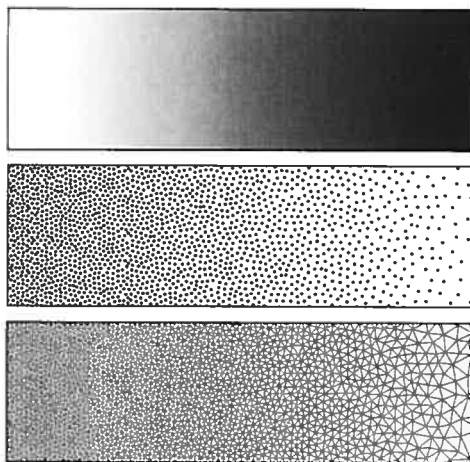


Figure 10: (Top) Gradient ramp importance density function. (Middle) *Quasisampler* output points. (Bottom) Triangulation obtained with our algorithm.

4 Results

4.1 Qualitative Results

An example of using our triangulation algorithm on a gradient ramp is shown in Figure 10. In Figure 12, a high dynamic range image is used as the importance density function. The quality of the results of our triangulation are intrinsically tied to the quality of the Delaunay triangulation. Whether this is a ‘good’ triangulation or not depends of course on the application, but the fact that the Delaunay triangulation maximizes the minimum angles of the triangles gives it many useful properties. Obviously, the quality of the triangulation is also tied to the quality of the distribution of the points generated by the *Quasisampler*. Given that the points follow a local blue-noise (or Poisson-disk) distribution, the dual of the triangulation, called the Voronoi diagram (see Figure 1), is very close to what is called a centroidal Voronoi tessellation, which confers to it some interesting properties, as further explained in [DFG99].

4.2 Quantitative Results

In order to compare our algorithm’s performance with others in real-world applications, we have timed the triangulation algorithms on an increasing number of points. The results are shown in Figure 11. For fairness’ sake, all three algorithms use the same triangulation data structure. Also, the implementations of the two compared algorithms are provided by the CGAL library [BDTY00], known for its good performance. As the graph shows, a naive algorithm such as the incremental insertion method, is no match for our algorithm. The brute force approach, in $O(n^4)$, is not even shown because it is substantially slower than all other methods. Devillers’ algorithm [Dev98] uses an efficient search structure, which gives a nearly linear performance on the point sets generated by the *Quasisampler*, given their blue-noise distribution. This makes for an algorithm that performs in the same order as ours. Nevertheless,

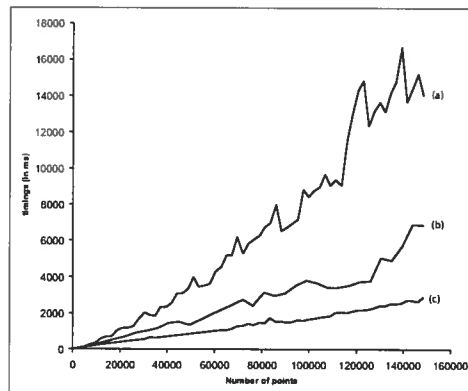


Figure 11: Performance comparison: (a) Incremental insertion. (b) Devillers’ algorithm. (c) Our algorithm. The source function is a non-trivial HDR image.

our algorithm manages to run at least twice as fast, and this is while adhering to the highest quality standards.

In addition, if the application can tolerate a few edges that are not in the Delaunay set, fewer edge flip passes can be afforded, resulting in a faster triangulation. In our implementation, we have observed up to a two-fold speed increase by cutting this corner, but the results may vary according to what is considered an acceptable deviation from the Delaunay set.

5 Conclusions and Future Work

We have addressed an important problem in computer graphics, that is to generate well distributed point sets along with their Delaunay triangulation, given an importance density function in 2D. To this end, we have developed a fast algorithm to generate a Delaunay triangulation of point sets obtained through the importance sampling system described in [ODJ04]. Although we have demonstrated that our algorithm runs in linear time, which cannot be obtained by any known Delaunay triangulation algorithm, we must acknowledge that other triangulation methods run almost linear in the expected (average) case. Regardless, we have demonstrated in our tests that we can run at least twice as fast as the best known algorithms, and possibly even faster if certain concessions to quality can be accepted, in terms of how many edges that are not in the Delaunay set.

As future work, we plan to extend the algorithm in order to generate 3D triangulations, and possibly n -D triangulations. Of course, this depends on whether the *Quasisampler* will be applicable or not in such dimensions. This is a question which we are looking into. Some operations on triangulations are trivial in the 2D case, but become complex in higher dimensions. Also, the L^2 norm on which the Delaunay constraint relies can become problematic in high dimensions, so it is questionable whether the Delaunay triangulation is ‘good’ in such situations.

Another extension that we plan to explore is the case of a dynamic function, where temporal coherency of the function could be exploited to save computation time, as opposed to simply rebuilding the triangulation at each frame. Given that the *Quasisampler* is

expected to return coherent point sets across frames (which is not the case in most other similar systems), the time savings could be considerable.

Finally, we plan to develop applications of the algorithm for promising uses in computer graphics. One such application is image compression, where an image would be partitioned into a triangulation which has a local density proportional to the image complexity. Another application is isotropic remeshing of 3D surfaces, in a manner similar to [AdVDI03], which could be made more interactive with our fast system.

References

- P. A., É. C. de Verdière, O. Devillers, and M. Isenburg. Isotropic surface remeshing. In *Proceedings of Shape Modeling International*, pages 49–58, 2003.
- J.-D. Boissonnat, O. Devillers, M. Teillaud, and M. Yvinec. Triangulations in CGAL (extended abstract). In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 11–18. ACM Press, 2000.
- F. Davoine, M. Antonini, J.-M. Chassery, and M. Barlaud. Fractal image compression based on delaunay triangulation and vector quantization. *IEEE Transactions on Image Processing*, 5(2), Feb. 1996.
- O. Devillers. Improved incremental randomized delaunay triangulation. In *Proceedings of the fourteenth annual symposium on Computational geometry*, pages 106–115. ACM Press, 1998.
- Q. Du, V. Faber, and M. Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. In *SIAM Review*, volume 41, pages 637–676. Society for Industrial and Applied Mathematics, 1999.
- S. Fortune. A sweepline algorithm for voronoi diagrams. In *Proceedings of the second annual symposium on Computational geometry*, pages 313–322. ACM Press, 1986.
- L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of voronoi diagrams. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 221–234. ACM Press, 1983.
- S. Hiller, O. Deussen, and A. Keller. Tiled blue noise samples. In *Proceedings of Vision, Modeling, and Visualization*, volume 3, pages 265–271. IOS Press, 2001.
- D.E. Knuth. *The Art of Computer Programming, Volume 1, Fundamental Algorithms*. page 86. Addison-Wesley, 3rd edition, 1997.
- S. Lloyd. An optimization approach to relaxation labeling algorithms. *Image and Vision Computing*, 1(2):85–91, 1983.
- V. Ostromoukhov, C. Donohue, and P.-M. Jodoin. Fast hierarchical importance sampling with blue noise properties. *ACM Transactions on Graphics*, 23(3), August 2004. Proc. SIGGRAPH 2004.
- R. Penrose. Pentaplexity, a class of non-periodic tilings of the plane. *The Mathematical Intelligencer*, 2:32–37, 1979.
- M. I. Shamos and D. Hoey. Closest-point problems. In *Proceedings of the sixteenth Annual Symposium on Foundations of Computer Science*, pages 151–162. IEEE, 1975.
- R. Ulichney. *Digital Halftoning*. MIT Press, 1987.

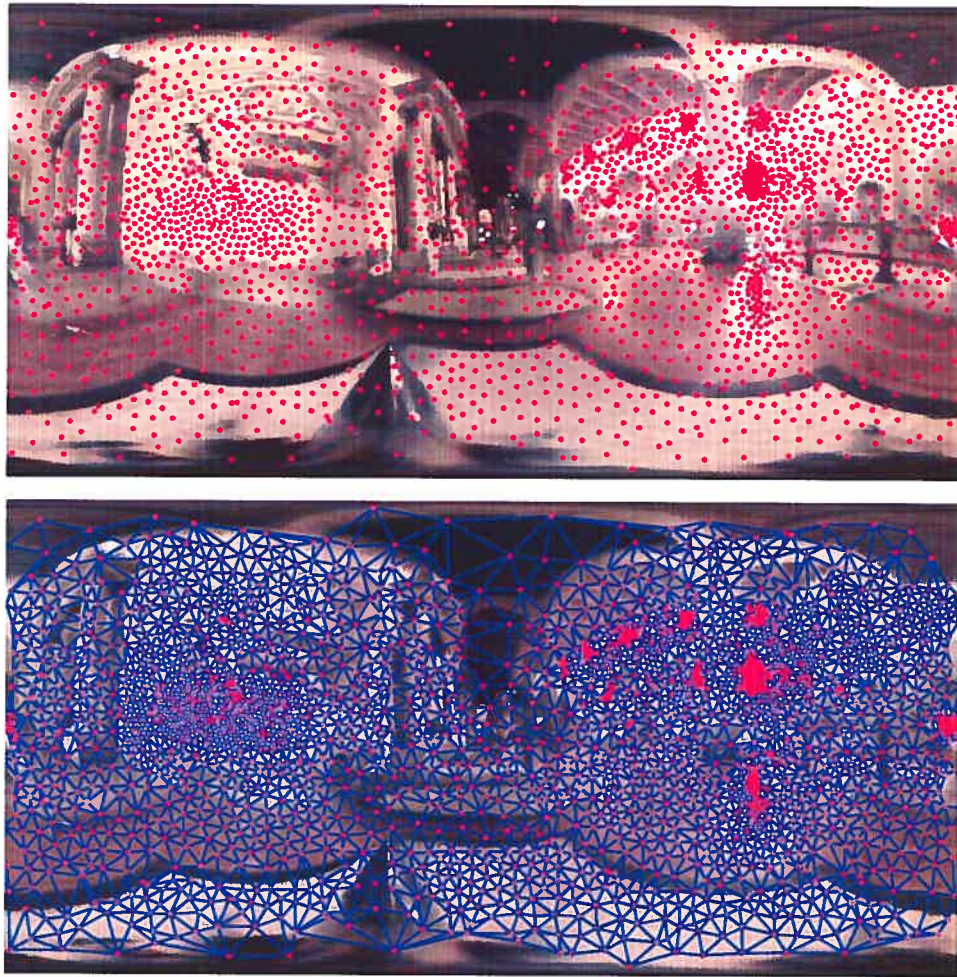


Figure 12: HDR map sampled with the *Quasisampler* (top), then triangulated using our algorithm (bottom). (HDR image source: Paul Debevec.)

2.2.3 Discussion

Dans cet article, nous avons adressé un problème important en infographie en abordant la notion de connectivité des points d'échantillonnage. Nous avons présenté un algorithme rapide pour la triangulation de Delaunay des ensembles de points générés par le système d'échantillonnage décrit dans l'article précédent. Nous avons démontré que notre algorithme se déroule en temps linéaire par rapport au nombre de points, en pire cas, ce qui est un exploit qu'aucun autre algorithme de triangulation de Delaunay ne peut atteindre. Il faut quand même tempérer cette déclaration par le fait que certains algorithmes peuvent donner des performances quasi-linéaires dans les cas typiques. Toutefois, nous avons démontré dans nos tests que nous pouvons être au moins deux fois plus rapide que les meilleurs algorithmes existants. De plus, les vitesses d'exécution peuvent être encore améliorées si on peut accepter que certaines arêtes ne fassent pas partie de l'ensemble Delaunay.

Dans nos travaux futurs, nous voudrions étendre cet algorithme aux triangulations 3D, et possiblement N-D, mais tout dépend de la possibilité d'étendre le système de base à ces dimensions, ce qui n'est pas encore fait. De plus, certaines opérations sur les triangulations qui sont triviales en 2D peuvent devenir compliquées en plus haute dimension. D'ailleurs, les choses peuvent se complexifier d'avantage à très haute dimension, alors que la métrique de distance L^2 sur laquelle dépend la contrainte de Delaunay devient moins significative.

Une autre extension que nous planifions explorer est celle des cas de fonctions dynamiques, dans lesquels on pourrait exploiter la cohérence temporelle du signal d'entrée afin d'éviter de rebâtir la structure de base à zéro. Le fait que le système d'échantillonnage soit déterministe et qu'il retourne des ensembles cohérents de points d'image en image, nous permettrait de faire des grandes économies en temps de calcul.

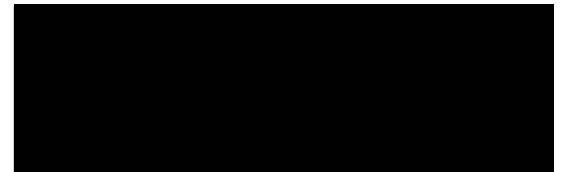
Finalement, nous voulons explorer l'application de notre algorithme de triangulation à certains problèmes prometteurs en infographie. Une première application serait la compression d'images, où l'image pourrait être partitionnée en une triangulation dans laquelle la densité serait proportionnelle à la complexité de celle-ci. Une autre application intéressante serait celle du remaillage isotropique des surfaces 3D, dont l'interactivité pourrait être améliorée par notre système.

2.2.4 Contribution Personnelle

En tant que premier auteur de cet article, il me revient l'élaboration de l'algorithme de triangulation qui y est présenté. J'ai aussi implémenté l'algorithme pour les fins de tests, monté les preuves algorithmiques, et rédigé l'article. Étant donné que la méthode est intimement liée au système d'échantillonnage proposé dans le premier article, la collaboration avec Victor Ostromoukhov a été très étroite à tous les égards.

2.2.5 Droits de publication

Par sa signature ci-dessous, le coauteur de la publication ci-jointe, *Fast Triangulated Importance Sampled Point Sets*, accorde sa permission explicite pour que l'article soit inclu tel quel dans le présent mémoire. Aussi, il atteste que ma contribution à l'article est effectivement telle que décrite dans le présent mémoire.



Victor Ostromoukhov

Chapitre 3

Conclusion

En exploitant certaines propriétés des tuiles de Penrose, nous avons présenté deux méthodes qui se penchent sur des problèmes reliés à l'échantillonnage en infographie.

Dans le premier article, nous proposons une technique d'échantillonnage 2D, qui non seulement donne des résultats de bonne qualité, mais peut le faire à une vitesse beaucoup plus rapide que les techniques existantes qui donnent de telles distributions. Nous avons illustré notre méthode avec un cas typique d'application, dans le contexte du rendu d'images de synthèse, mais le système peut s'appliquer à une grande variété de problèmes en infographie.

Dans le second article, nous proposons une technique d'échantillonnage qui inclut également la notion de la connectivité des échantillons résultants, au sens de Delaunay. La technique permet d'obtenir une triangulation de Delaunay dans temps un d'au moins deux fois plus rapide qu'avec les meilleures techniques existantes pour accomplir cette tâche. Combinée avec la méthode d'échantillonnage précédente, nous avons un système qui donne des résultats de bonne qualité et à haute vitesse, ce qui ouvre la porte à plusieurs applications en infographie, dans lesquelles la vitesse joue un rôle critique.

En conclusion, nous croyons sincèrement que l'approche proposée dans ces articles face aux problèmes d'échantillonnage est très prometteuse, et que nous n'avons qu'effleuré le potentiel réel des tuiles de Penrose dans le domaine de l'infographie.

Bibliographie

- [AdVDI03] P. Alliez, É. C. de Verdière, O. Devillers et M. Isenburg. “Isotropic Surface Remeshing”. In *Proceedings of Shape Modeling International*, pages 49–58, 2003.
- [AMD02] P. Alliez, M. Meyer et M. Desbrun. “Interactive Geometry Remeshing”. *ACM Trans. on Graphics*, volume 21, numéro 3, pages 347–354, 2002.
- [ARBJ03] S. Agarwal, R. Ramamoorthi, S. Belongie et H.W. Jensen. “Structured Importance Sampling of Environment Maps”. *ACM Trans. on Graphics*, volume 22, numéro 3, pages 605–612, juillet 2003.
- [BDTY00] J.-D. Boissonnat, O. Devillers, M. Teillaud et M. Yvinec. “Triangulations in CGAL (extended abstract)”. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pages 11–18. ACM Press, 2000.
- [CD01] J. Cohen et P. Debevec. *LightGen, HDRShop plugin*. <http://www.ict.usc.edu/~jcohen/lightgen/lightgen.html>, 2001.
- [Coo86] R.L. Cook. “Stochastic Sampling in Computer Graphics”. *ACM Trans. on Graphics*, volume 5, numéro 1, pages 51–72, janvier 1986.
- [CSHD03] M.F. Cohen, J. Shade, S. Hiller et O. Deussen. “Wang Tiles for Image and Texture Generation”. *ACM Transactions on Graphics*, volume 22, numéro 3, pages 287–294, juillet 2003.
- [DACB96] F. Davoine, M. Antonini, J.-M. Chassery et M. Barlaud. “Fractal image compression based on Delaunay triangulation and vector quantization”. *IEEE Transactions on Image Processing*, volume 5, numéro 2, 1996.

- [Deb98] P. Debevec. “Rendering Synthetic Objects Into Real Scenes : Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography”. In *Proc. SIGGRAPH '98*, pages 189–198, juillet 1998.
- [Dev98] O. Devillers. “Improved incremental randomized Delaunay triangulation”. In *Proceedings of the fourteenth annual symposium on Computational geometry*, pages 106–115. ACM Press, 1998.
- [DFG99] Q. Du, V. Faber et M. Gunzburger. “Centroidal Voronoi Tessellations : Applications and Algorithms”. *SIAM Review*, volume 41, numéro 4, pages 637–676, décembre 1999.
- [For86] S. Fortune. “A sweepline algorithm for Voronoi diagrams”. In *Proceedings of the second annual symposium on Computational geometry*, pages 313–322. ACM Press, 1986.
- [FvDFH90] J.D. Foley, A. van Dam, S.K. Feiner et J.F. Hughes. *Computer Graphics, Principles and Practice*. Addison-Wesley, 2nd édition, 1990.
- [Gar77] M. Gardner. “Extraordinary Nonperiodic Tiling that Enriches the Theory of Tiles”. *Scientific American*, volume 236, pages 110–121, 1977.
- [GKS90] L. J. Guibas, D. E. Knuth et M. Sharir. “Randomized incremental construction of Delaunay and Voronoi diagrams”. In *Proceedings of the seventeenth international colloquium on Automata, languages and programming*, pages 414–431. Springer-Verlag New York, Inc., 1990.
- [Gla98] A. Glassner. “Andrew Glassner’s Notebook : Penrose Tiling”. *IEEE Computer Graphics & Applications*, volume 18, numéro 4, pages 78–86, 1998.
- [GS83] L. J. Guibas et J. Stolfi. “Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams”. In *Proceedings of the fifteenth annual ACM symposium on Theory of computing*, pages 221–234. ACM Press, 1983.
- [GS86] B. Grünbaum et G.C. Shephard. *Tilings and Patterns*. W.H. Freeman, 1986.
- [HCK+99] K. Hoff, T. Culver, J. Keyser, M. Lin et D. Manocha. “Fast Computation of Generalized Voronoi Diagrams using Graphics Hardware”. In *Proc. SIGGRAPH '99*, pages 277–286, août 1999.

- [HDK01] S. Hiller, O. Deussen et A. Keller. “Tiled Blue Noise Samples”. In *Proceedings of Vision, Modeling, and Visualization*, volume 3, pages 265–271. IOS Press, 2001.
- [KK01] T. Kollig et A. Keller. “Efficient bidirectional path tracing by randomized quasi-Monte Carlo integration”. *Niederreiter, K. Fang, and F. Hickernell, Eds., Monte Carlo and Quasi-Monte Carlo Methods 2000*, pages 290–305, 2001.
- [KK02] T. Kollig et A. Keller. “Efficient Multidimensional Sampling”. *Computer Graphics Forum*, volume 21, numéro 3, pages 557–564, 2002.
- [KK03] T. Kollig et A. Keller. “Efficient Illumination by High Dynamic Range Images”. In *Eurographics Symposium on Rendering : 14th Eurographics Workshop on Rendering*, pages 45–51, 2003.
- [Knu97] D.E. Knuth. *The Art of Computer Programming, Volume 1, Fundamental Algorithms*. page 86. Addison-Wesley, 3^e édition, 1997.
- [Llo83] S. Lloyd. “An optimization Approach to Relaxation Labeling Algorithms”. *Image and Vision Computing*, volume 1, numéro 2, pages 85–91, 1983.
- [Mac82] A.L. Mackay. “Crystallography and the Penrose Pattern”. *Physica*, volume 114A, pages 609–613, 1982.
- [MF92] M. McCool et E. Fiume. “Hierarchical Poisson Disk Sampling Distributions”. In *Proc. Graphics Interface '92*, pages 94–105, mai 1992.
- [Mit91] D.P. Mitchell. “Spectrally Optimal Sampling for Distributed Ray Tracing”. In *Proc. SIGGRAPH '91*, volume 25, pages 157–164, juillet 1991.
- [Nie92] H. Niederreiter. *Random Number Generation and Quasi-Monte-Carlo Methods*. Soc. for Industrial and Applied Mathematics, 1992.
- [ODJ04] V. Ostromoukhov, C. Donohue et P.-M. Jodoin. “Fast Hierarchical Importance Sampling with Blue Noise Properties”. *ACM Transactions on Graphics*, volume 23, numéro 3, August 2004. Proc. SIGGRAPH 2004.
- [OHA94] V. Ostromoukhov, R.D. Hersch et I. Amidror. “Rotated Dispersion Dither : a New Technique for Digital Halftoning”. In *Proc. SIGGRAPH '94*, pages 123–130, juillet 1994.

- [Ost01] V. Ostromoukhov. "A Simple and Efficient Error-Diffusion Algorithm". In *Proc. SIGGRAPH 2001*, pages 567–572, août 2001.
- [Pen74] R. Penrose. "The role of Aesthetics in Pure and Applied Mathematical Research". *Bull. Inst. Math. & its Applns.*, volume 10, pages 266–271, 1974.
- [Pen79] R. Penrose. "Pentaplexity, a Class of Non-Periodic Tilings of the Plane". *The Mathematical Intelligencer*, volume 2, pages 32–37, 1979.
- [QD99] M. Gunzburger Q. Du, V. Faber. "Centroidal Voronoi Tessellations : Applications and Algorithms". In *SIAM Review*, volume 41, pages 637–676. Society for Industrial and Applied Mathematics, 1999.
- [SAG03] V. Surazhsky, P. Alliez et C. Gotsman. "Isotropic Remeshing of Surfaces : a Local Parameterization Approach". In *Proc. of 12th Int. Meshing Roundtable*, 2003.
- [SH75] M. I. Shamos et D. Hoey. "Closest-point problems". In *Proceedings of the sixteenth Annual Symposium on Foundations of Computer Science*, pages 151–162. IEEE, 1975.
- [Shi91] P.S. Shirley. "Discrepancy as a Quality Measure for Sample Distributions". In *Proc. Eurographics '91*, pages 183–194, septembre 1991.
- [SHS02] A. Secord, W. Heidrich et L. Streit. "Fast Primitive Distribution for Illustration". In *13th Eurographics Workshop on Rendering*, pages 215–226, juin 2002.
- [SO87] P. Steinhardt et S. Ostlund. *The Physics of Quasicrystals*. World Scientific, 1987.
- [Soc89] J.E.S. Socolar. "Simple Octagonal and Dodecagonal Quasicrystals". *Physical Review*, volume B39, pages 10519–10551, 1989.
- [Uli87] R. Ulichney. *Digital Halftoning*. MIT Press, 1987.
- [Uli88] R. A. Ulichney. "Dithering with Blue Noise". *Proc. of the IEEE*, volume 76, pages 56–79, 1988.
- [Vea97] E. Veach. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis. Stanford University, 1997.