

Random-Edge Discrepancy of Supersampling Patterns

David P. Dobkin+
Don P. Mitchell

Princeton University
AT&T Bell Laboratories

1. Abstract

Patterns used for supersampling in graphics have been analyzed from statistical and signal-processing viewpoints. We present an analysis based on a type of isotropic discrepancy—how good patterns are at estimating the area above an arbitrary edge through a pixel. An algorithm is presented for computing the worst-case discrepancy. Experimental evidence shows that popular supersampling patterns have discrepancies with better asymptotic behavior than random sampling, which is not inconsistent with theoretical bounds on discrepancy.

2. Introduction

Supersampling is one of the most general approaches to the antialiasing problem in graphics. Since symbolic images usually cannot be prefiltered, aliasing is reduced by sampling at a very high rate, and then digitally resampling to the pixel rate. In applications like ray tracing and distribution ray tracing, this is the only general solution to aliasing currently known. In the simplest form of resampling, supersamples are averaged within a square pixel area to compute a pixel value.

Supersampling can be done in a uniform pattern, but it has been shown that there are advantages to using nonuniform or stochastic sampling patterns. Uniform sampling can lead to visually conspicuous aliasing artifacts like Moiré patterns. This is worst-case behavior in the context of adaptive sampling, where a pure high-frequency signal aliased to a pure low-frequency pattern will fool schemes for deciding where to apply extra samples [Whitted80]. Randomizing the sampling pattern leads to random-noise aliasing and is more likely to avoid the worst-case scenario for adaptive sampling.

The quality of sampling patterns has been analyzed from several viewpoints. Estimating the integral of a pixel area by averaging samples can also be viewed as a statistical sampling problem, and variance-reducing techniques of experimental design (e.g., stratification) can be applied [Lee85, Kajiya86, Purgathofer86, Painter89]. The Central Limit Theorem implies that pixel error will decrease with

the number of samples as $O(N^{-1/2})$, but this viewpoint does not describe overall image-noise characteristics.

Some researchers have taken a *signal-processing* viewpoint of the image-sampling problem [Dippé85, Cook86, Mitchell87] and of the distribution ray-tracing problem [Mitchell91]. Here, it has been shown that sampling patterns can be designed to drive aliasing noise into higher frequencies, where it may be removed by the pixel-resampling process and where it is less visually conspicuous. This is achieved with samples having a high-frequency spectrum ("blue-noise").

A third viewpoint which can be applied to the problem of sample-pattern analysis is the theory of discrepancy or irregularities of distribution [Beck87]. This viewpoint was introduced to computer graphics by Shirley [Shirley91]. Niederreiter has also pointed out the possible importance of discrepancy in computer graphics [Niederreiter92]. This subject grew out of the study of certain *low-discrepancy* sampling patterns which have been used in quasi-Monte Carlo integration [Halton70, Niederreiter78]. What is interesting about discrepancy is that it provides a fairly direct measurement of how good a sampling pattern is at estimating certain simple integrals. From the theory of this subject has emerged the fascinating fact that some sampling patterns yield sampling errors that are asymptotically smaller than the $O(N^{-1/2})$ of uniform random sampling.

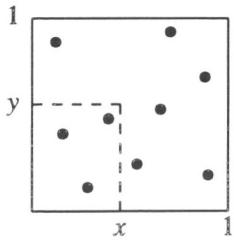
In the next two sections, we study discrepancy function relative to two simple geometric measures, axis-aligned rectangles and arbitrary edges. Theoretical as well as empirical results are presented. In section 4, we give efficient algorithms for computing the discrepancy with respect to arbitrary edges. We conclude with further experimental results and a discussion of open problems.

3. Discrepancy of Axis-Aligned Rectangles

Assume, for example, we are given a pattern of N samples in the unit square. For some (x,y) in the square, we can estimate the area of the rectangle $[0,x] \times [0,y]$ by counting the number of samples $v(x,y)$ within it.

+ The work of this author has been supported in part by NSF grant CCR90-02352 and by The Geometry Center, University of Minnesota, an STC funded by NSF, DOE, and Minnesota Technology, Inc.





The true area is given by the product xy and we will call the error the *local discrepancy* at the point (x,y) :

$$\Delta(x,y) = \frac{v(x,y)}{N} - xy$$

The irregularity of the distribution of samples can be measured by averaging the local discrepancy over all possible values of x and y in the square (with the obvious extension to higher dimensions). The L^∞ -discrepancy is defined to be the maximum absolute value of $\Delta(x,y)$:

$$D_N = \sup_{x,y} |\Delta(x,y)|$$

and the L^2 -discrepancy is given by:

$$T_N = \left[\iint_0^1 \Delta^2(x,y) dx dy \right]^{1/2}$$

This is motivated by the importance of low-discrepancy sampling in numerical integration. For example, in one dimension, there is the significant result:

Theorem (Koksma 1942). If f is a function of bounded variation $V(f)$ on the unit interval I and x_1, \dots, x_N are points in I with L^∞ -discrepancy D_N , then

$$\left| \frac{1}{N} \sum_{i=1}^N f(x_i) - \int_0^1 f(t) dt \right| \leq V(f) D_N$$

This is actually quite easily proven with integration by parts, noting that $\partial v(x)/\partial x$ will be a sequence of Dirac delta functions, $\sum \delta(x-x_i)$. Koksma's theorem was extended to higher dimensions [Niederreiter78], but the definition of bounded variation is problematic. Nevertheless, for a given function obeying the bounded-variation conditions, the error of numerical integration is $O(D_N)$. Roth has proven that, in k dimensions, the best sampling patterns have discrepancy tightly bounded by $T_N = \Theta(N^{-1}(\log N)^{(k-1)/2})$ [Beck87].

Sampling patterns have been constructed with discrepancies of $D_N = O(N^{-1}(\log N)^{k-1})$, such as the *Hammersley points*. Let $\phi_r(n)$ be the radical-inverse function of n base r . Its value is a real number from 0 to 1 constructed by taking the integer n , represented in base r , and reflecting its digits about the decimal point to form a fraction, base r . Given the sequence of prime numbers $2, 3, 5, \dots$, one of N Hammersley points is given by:

$$\mathbf{x}_i = (i/N, \phi_2(i), \phi_3(i), \phi_5(i), \dots)$$

An improvement suggested by Zaremba and generalized to k dimensions by Warnock is based on the folded radical inverse $\psi_r(n)$ [Warnock72]. Here, the i^{th} most significant digit a_i is replaced by $(a_i + i) \bmod r$ before the reflection about the decimal point. In the same paper, Warnock presents an $O(N^2)$ algorithm for computing T_N and experimental results for several proposed low-discrepancy patterns.

Shirley computed the L^∞ - and L^2 -discrepancies of a number of commonly used nonuniform sampling patterns. It is worth doing a similar set of experiments again, but with progressively higher densities to get a flavor of the asymptotic behavior. We will consider four sampling patterns. The first is Zaremba's low-discrepancy pattern generated with the folded radical inverse function. The second is the jittered sampling pattern obtained by randomly perturbing a regular periodic pattern. The third is the random Poisson-distributed pattern which is approximated by generating N uniformly-distributed points on the unit hypercube (in a section of a true Poisson process, N itself would have a Poisson distribution about the mean sample density). The fourth pattern is a "Poisson-disk" pattern generated by a dart-throwing algorithm on the unit torus (promoted as the "best pattern known" by advocates of the signal-processing viewpoint).

Process	16 points	256 points	1600 points
Zaremba	0.0358	0.00255	0.000438
jittered	0.0501	0.00627	0.00161
Dart-Throwing	0.0521	0.00794	0.00258
Poisson	0.0900	0.0211	0.00867

Table 1. 2-Dimensional L^2 -discrepancies

These numbers are averages from 100 trials (except for the deterministic Zaremba pattern). The values for $N = 16$ agree fairly well with Shirley's results, and the values for the Poisson process agree with the theoretical value of $T_N^2 = N^{-1}(2^{-k} - 3^{-k})$ for k -dimensional patterns.

The most important feature to notice is the asymptotic behavior as N increases. For Poisson patterns, the $O(N^{-1/2})$ behavior is evidenced by the fact that increasing N by a factor of 100 only decreased the discrepancy by a factor of 10. At the other extreme, the discrepancy of Zaremba's pattern decreases at an impressive rate. The jitter and dart-throwing patterns are intermediate, but it is very interesting to note that their discrepancy seems to be better than $O(N^{-1/2})$.

4. Discrepancy of Arbitrary Edges

Theorems such as Koksma's and the remarkable $O(N^{-1}(\log N)^{k-1})$ low-discrepancy patterns give an initial impression that vast improvements can be easily made in the efficiency of ray tracing or distribution ray tracing



(by replacing Monte Carlo integration with quasi-Monte Carlo methods). This section may, to some extent, dash those hopes. First, it should be noted that it is easy to create an image which does not have bounded variation everywhere (a checkerboard viewed in perspective, for example). A more fundamental problem occurs if we notice that synthetic images contain edges and curved boundaries at arbitrary orientations. The existence of $\Theta(N^{-1}(\log N)^{(k-1)/2})$ low-discrepancy patterns is limited to the problem of estimating the area of *axis-aligned* rectangles.

Consider the problem of estimating the area of disks or of boxes at arbitrary orientation. Surprisingly, the discrepancy Δ is much larger. Work by Schmidt and others have shown that one can do no better than the lower bound of $\Omega(N^{-1/2k-1/2})$ in k dimensions [Beck87]. Upper bounds on the best discrepancy are known, and are typically larger than the lower bounds by a polylogarithmic factor (for various generalized discrepancy problems). In two dimensions, this still suggests the existence of patterns that are much better than random; however, as the dimension increases, one can probably not do better than random sampling.

A common problem in computer graphics is a pixel in the neighborhood of an edge. Therefore, consider a unit square with an arbitrary line passing through it. As before, we have a pattern of samples inside the square. We can define a *arbitrary-edge discrepancy* between the area of the square above the line and the fraction of sample points above the line. Our goal is to develop an algorithm for measuring the L^∞ norm (worst-case arbitrary line) for a given sampling pattern and then to analyze the behavior of the four sampling patterns.

Theoretical analysis has been done on the similar problem of arbitrary edges through samples within a unit-area disk [Beck87]. Bounds on the worst-case (the best pattern you can get, given the worst-case line) are $\Omega(N^{-3/4}(\log N)^{-7/2})$ and $O(N^{-3/4}(\log N)^{1/2})$. These bounds also apply to our case of points in a unit square [Beck92].

5. Algorithms for Computing Arbitrary-Edge Discrepancy

In this section, we give an $O(N^2)$ algorithm for computing the maximum arbitrary-edge discrepancy of a set of N points in the unit square. Our methods work by appealing to intuition in dual space to aid in deriving the algorithm. We then bring these methods back to primal space to ease their actual implementation. Our techniques first appeared in the papers [Edelsbrunner86, Chazelle85]. Further details are given in [Edelsbrunner87].

The problem presents itself as having as input a collection of points $P_i = (x_i, y_i)$ ($i = 1, \dots, N$) in the unit square. Initially, we might think we have to consider all lines through the unit square. This is not the case. Rather, we

give a collection of $O(N^2)$ lines which are sufficient to determine the maximum arbitrary-edge discrepancy of the set.

We think of lines as being oriented, so that a line gives rise to a region of points above the line within the unit square. Points of this region are in counterclockwise orientation with respect to the line. For each line we can measure an edge discrepancy. This is determined as the difference between the fraction of the points $\{P_i\}$ we expect to see in the region (i.e. the region's area) and the fraction we actually see. We show how to compute the area of the unit square above a clipping line and the number of points above that line. From these quantities, it is a simple matter to compute the edge discrepancy of the line in either orientation.

Our original points are given in primal space (as coordinate pairs (x, y)). Lines are expressed most naturally by equations $y = mx + b$. They can also be represented as points (the coordinate pair (m, b)) in dual space. We will move between these two spaces as needed to simplify our discussion.

To begin, we reverse the problem. Instead of talking about points above a line, we talk about all lines which a point is above. Suppose $P = (x, y)$ is a point in the unit square. Then, P is above the line L given by the point (m, b) in dual space if $y > mx + b$. This gives rise to the line $b = xm - y$ in dual space. Any (m, b) above this line corresponds to a line which P is above. It is this transformation to dual space which defines our algorithm.

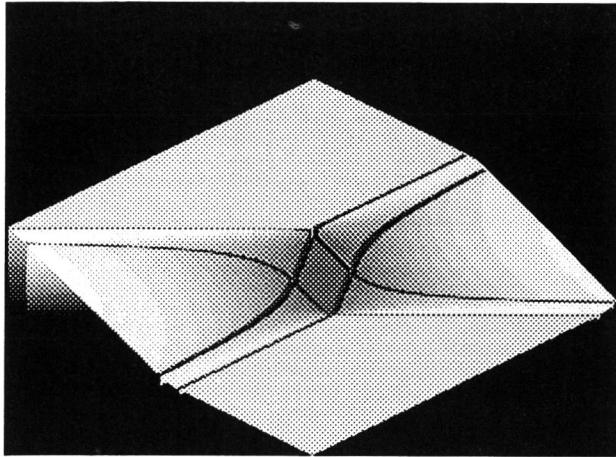
For each point P_i , we now can create a line in dual space. The collection of these lines is called their *arrangement*. Observe that the regions of the arrangement correspond to lines which have the same set of (primal space) points above them. Further, these regions are convex. We next turn to a discussion of how to compute the area of the region of the unit square corresponding to a given (m, b) clipping line. If this area function is well-behaved, in a manner to be made precise below, it will suffice to compute the maximum discrepancy as the maximum discrepancy at any vertex of the line arrangement.

The area function is computed by a case analysis. We consider 9 classes of lines, given three ranges of the y intercept and which of three sides of the square (not on the y axis) the line passes through. For convenience, we define $A(m, b)$ as twice the area of the region in the table below.

The area $A(m, b)/2$ is displayed as the surface in Figure 1. In addition, we can define a counting function $C(m, b)$ which returns the number of points above the line. For a typical set of 16 jittered points, Figure 2 shows the function $C(m, b)/N$ which is the discrete approximation to the area. We define a discrepancy function $D(m, b) = |A(m, b)/2 - C(m, b)/N|$ which is illustrated in Figure 3.



	$b > 1$	$1 > b > 0$	$0 > b$
$m > 1-b$	0	$\frac{(1-b)^2}{m}$	$\frac{1-2b}{m}$
$1-b > m > -b$	$-\frac{(m+b-1)^2}{m}$	$2 - (m + 2b)$	$2 - \frac{(m+b)^2}{m}$
$-b > m$	$\frac{2m+2b-1}{m}$	$2 + \frac{b^2}{m}$	2

Table 2. $A(m,b)$ - twice the area of the region.Figure 1: The Function $A(m,b)$ with boundary lines added

We observe that the area function $A(m,b)$ is continuous between the regions defined in the table. Furthermore, within each region, the partial derivatives $\frac{\partial A(m,b)}{\partial m}$ and $\frac{\partial A(m,b)}{\partial b}$ are non-zero.

This non-zero condition tells us that a point in dual space can always be moved in some direction to increase the area of the clipped region to which it corresponds. Similarly there is a direction of motion which will decrease this area. This leads to the observation.

Lemma: Let R be any region in (m,b) space which is defined by a convex polygon and lies totally in one region of the partition defined above. Then, the extrema of $A(m,b)$ over R occur along the edges of R .

We can extend the power of this lemma by adding the lines which define the boundaries between regions of the

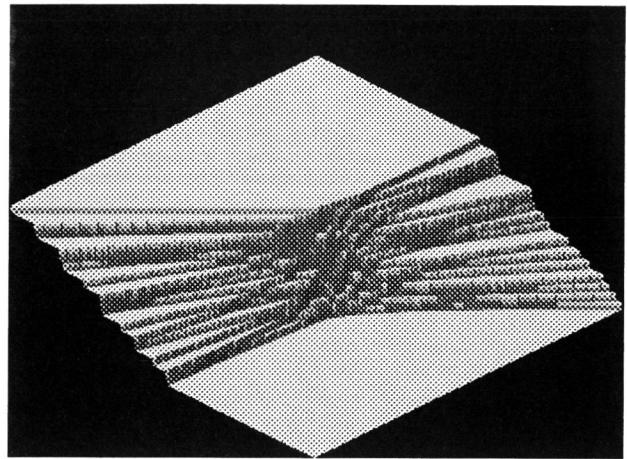


Figure 2: The Count function

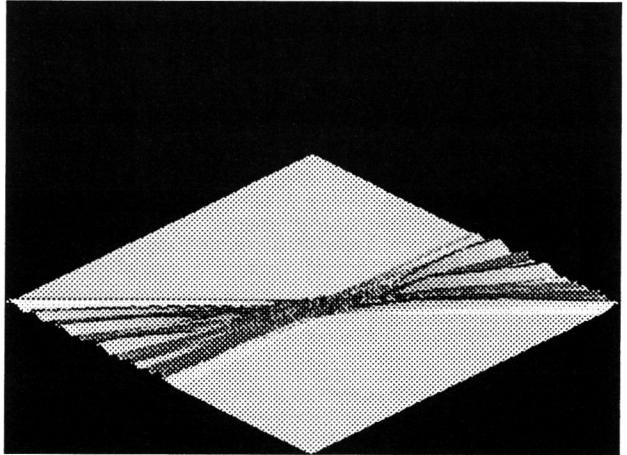


Figure 3: The Discrepancy function

area function to the arrangement. This then leads to the following

Lemma: Consider the set of points $P_i, i=1, \dots, N$. If we form the arrangement of the lines \hat{P}_i (where \hat{P}_i is dual to P_i) with the additional lines $b = 1$, $b = 0$, $b = 1-m$, and $b = -m$, the maximum edge discrepancy occurs along an edge of this arrangement.

Proof: We observe that the added lines create regions in the arrangement such that each region is convex.



Corresponding to each region is a subset of the P_i which lie above all lines (in primal space) represented by points (in dual space) in the region. Further, the partial derivatives of the area function are nonzero and of the same sign within the region (ie the function is also convex). The previous lemma combined with this observation implies that the extrema of the function occur along edges of the region. Since the discrepancy is defined at points of the region by subtracting a constant from the area, the extrema of the discrepancy must also occur along the edges. Note that the added lines yield the subdivision of the area function as shown in Figure 4 illustrating the proof of the theorem.

We have specified above that the regions be convex and that both partial derivatives be non-zero within a region. This assures that if we translate the origin of our coordinate system to any point of the region, we get 4 quadrants. In one of these, the area function increases as we traverse a vector. In another, the function decreases. We can make no assumptions about the other two since we are taking linear combinations of a positive and a negative quantity. Indeed there may be vectors in these quadrants along which the directional derivative passes through zero. It is these vectors that require us to consider the behavior of the area function along edges as well as at vertices.

Consider now a line in dual space, corresponding to a point in primal space. The points on the line in dual space correspond to the lines through the point in primal space. Traversing the dual line is the same as rotating a line about the primal point. Suppose we were to compute the area clipped by each line through the point. This computation is the same as computing this area at each point of the line in dual space. This area function in primal space potentially has 8 extrema. Four of these occur at the corners of the unit square (and correspond exactly to the 4 clipping lines we added to the arrangement). These correspond to situations where the shape of the clipped area changes.

The other 4 potential extrema are more subtle. These occur in situations where our rotating line clips regions through which the area function is non-monotonic even though we pass through no corner point. To see this, observe that the rotating line always divides the unit square into either 2 trapezoids or a pentagon and a triangle. In the former case, extrema are realized at vertices of the unit square and nowhere else. It is the latter case that interests us.

Consider the counterclockwise rotation from the line labelled B to the line labelled A in Figure 5. At B , a triangle is being clipped. The vertices of this triangle lie on the line $y = 1$, at the vertex $(0,1)$ and on the line $x = 0$. The triangle is a right triangle, so its area is the product of the lengths of its two legs along the square's boundary. Consider a line of slope m passing through the point

(p,q) , its area is $(1 - (q - mp))^2/m$. Differentiating, we determine that this results in an extremum when $m = -(q-1)/p$. This extremum is realized by the segments labelled $R_x(A)$ and $R_y(A)$ in Figure 5. This construction could be translated to any of the other corners resulting in possible clipping lines of slope $-q/p$, $-q/(p-1)$, and $-(q-1)/(p-1)$. We realize endpoints of these lines by the reflections shown in Figure 5. Notice that some of the reflections yield points outside the unit square and so can be ignored. Others, such as $R_x(D)$ lie within the square but lead to clipping lines that create trapezoids and so are uninteresting.

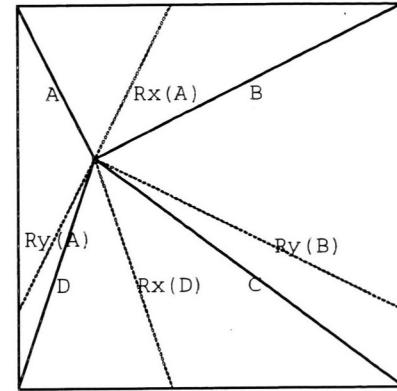


Figure 5: We begin with the four segments A, B, C, D joining the point to the four corners. The other segments represent reflections. $R_x(A)$ (resp. $R_y(B)$) is the reflection of the segment A (resp. B) through a line of constant x (resp. y).

By considering all of the reflections (i.e. 4 lines of special slope for each input point) we assure that we find all extrema. Each of the cutting lines correspond to a point along the dual line. The addition of these $4n$ vertices is sufficient to assure that the maximum discrepancy occurs (ie all extrema of the area function occur) at vertices of the arrangement.

This leads to the theorem:

Theorem (dual) : *The maximum edge discrepancy for the input points $P_i, i=1, \dots, N$ occurs at a vertex of the arrangement constructed as follows:*

(i) We form the arrangement of the lines \hat{P}_i dual to the P_i .

(ii) We add to this arrangement, the boundary lines $b = 1$, $b = 0$, $b = 1-m$, and $b = -m$.

(iii) For each point $P_i = (x_i, y_i)$, we add the points of slope (i.e. m -coordinate in dual space) $-y_i/x_i$, $-(1-y_i)/x_i$, $-y_i/(1-x_i)$, and $-(1-y_i)/(1-x_i)$ as vertices of this



arrangement.

It remains to turn the theorem above into an algorithm. To satisfy condition (iii), we can in quadratic time compute the discrepancy at each of the $4n$ slopes mentioned there. Since this running time will be dominated by anything else we do, we perform this computation and then turn to satisfying conditions (i) and (ii).

We could satisfy the first two conditions by first computing the arrangement of the $N+4$ lines (comprised of the N dual lines and the 4 additional lines). Each region of the arrangement could record the number of points dominated by lines in that region. We would then determine the $O(N^2)$ vertices of the arrangement, compute the discrepancy at each and determine the maximum discrepancy. Finding the arrangement takes $O(N^3)$ by the naive algorithm but can be computed in $O(N^2)$ by the algorithms of [Edelsbrunner86, Chazelle85]. In practice, this approach is prone to difficulties if data are not in general position.

An alternative is to consider how vertices in the arrangement (in dual space) arise. A vertex is at the intersection of two lines, each of which is the dual of a point in primal space. In primal space, the vertex corresponds to the line connecting the 2 points. Thus, we can restate the theorem above as follows:

Theorem (primal): *The vertices of the arrangement created by conditions (i) and (ii) of the dual theorem can be created as follows:*

Given the set of points $P_i, i=1, \dots, N$. Consider the set of lines formed by connecting 2 of the $n+4$ points consisting of the P_i enhanced by the 4 points $(0,0), (0,1), (1,0), (1,1)$. The maximum arbitrary-edge discrepancy is the edge discrepancy of one of these lines.

The proof of this theorem then follows by dualizing the previous theorem statement since the 4 added points are merely the duals of the 4 lines added above.

This theorem gives rise to a trivial $O(N^3)$ algorithm, and to an easily implemented $O(N^2 \log N)$ algorithm for computing the discrepancy. Let p be one of the $N+4$ points and sort the other points radially about p . Now, we choose p and the first point in sort order and determine how many points are in the region they define from which the discrepancy of the line they define can be found. Having done so, we can update to the next point in sorted order by sweeping through the sorted list. Note that if the sorted points are labeled q_1, q_2, \dots, q_k and q_i and q_j lie in the same portion of the unit square when clipped by the line connecting p and q_r , then so do all of the points

between q_i and q_j in either clockwise or counterclockwise order. We use this observation to show that we need sweep through the sorted list no more than 2 times to find the discrepancies of all lines having p as one endpoint. This immediately gives the $O(N^2 \log N)$ algorithm. Note that the four extra points should not be counted by $C(m,b)$, just used to define extra candidate worst-case lines. Finally, we compare this maximum discrepancy with that computed under condition (iii) to get the true answer.

We have implemented both of these algorithms. Timings (in seconds) are given in Table 3 below. These are running times on a SparcStation 2. These times concur with our theoretical analysis and display clearly the benefit of a faster algorithm.

Points	Fast algorithm	Slow algorithm
100	1.7	3.2
200	6.8	23.9
400	28.7	178.7
800	120.2	1399.5
1600	513.2	11272.3
3200	2258.9	89351.3
6400	9554.8	738041.4

Table 3. Times on $O(N^2 \log N)$ and $O(N^3)$ algorithms

It should be noted that it is an easy matter to extend the algorithm to higher dimensions. See [Dobkin92] for further details.

6. Results and Conclusions

L^∞ arbitrary-edge discrepancy was computed for four important types of sampling patterns. For the stochastic point processes (dart-throwing, jittered, Poisson), 100 trials were made to get an average for the process.

Two Monte-Carlo algorithms have been used to search for patterns of low discrepancy. An on-line algorithm builds of patterns one point at a time by trying to find the best next point. Given a pattern of N points, this method generates MN random points and then selects the one which causes the smallest increase in discrepancy (see [Mitchell91] for discussion of a similar on-line blue-noise algorithm). An off-line algorithm starts with N random points and attempts to replace points at random if the replacement will reduce the discrepancy.



Process	16 points	256 points	1600 points
Zaremba	0.184	0.0345	0.0158
jittered	0.183	0.0296	0.00854
Dart-Throwing	0.180	0.0339	0.0118
Poisson	0.299	0.0791	0.0337
On-Line MC	0.169	0.0281	-
Off-Line MC	0.106	0.0215	-

Table 4. Discrepancy of arbitrary edges

It is interesting to see how well Zaremba's pattern works, although it is not converging as dramatically as it could with axis-aligned rectangles. Once again, random sampling is consistent with $O(N^{-1/2})$ accuracy. The jittered and dart-throwing patterns perform best, which is consistent with their good "blue-noise" properties. Of these two, jittered sampling shows a slightly better discrepancy, which is not consistent with its spectral and visual properties, which are inferior to dart-throwing. The Monte Carlo experiments provide a low-water mark, patterns with the lowest discrepancy that we have found. A very interesting open problem is to find the minimum-discrepancy pattern of N points.

In conclusion, we have defined a arbitrary-edge discrepancy measure, motivated by the edge antialiasing problem in graphics. We describe algorithms for computing its L^∞ norm. A very simple $O(N^3)$ algorithm and an only slightly more complicated $O(N^2 \log N)$ algorithm exist in point space, and both of these have been implemented. An $O(N^2)$ algorithm exists in line space, which we have not attempted to implement.

Applying this algorithm to some common supersampling patterns, we find evidence for convergence faster than $O(N^{-1/2})$ which is theoretically possible. This is interesting, but the relative discrepancy of jittered and dart-throwing sampling are somewhat inconsistent with spectral and image-quality properties.

7. References

- [Beck87] Beck, J., Chen, W. W. L. *Irregularities of distribution*, Cambridge University Press (1987).
- [Beck92] Beck, J. *personal communication*.
- [Chazelle85] Chazelle, B., L. J. Guibas and D. T. Lee. The power of geometric duality. *BIT* 25, (1985), 76-90.
- [Cook86] Cook, R. L. Stochastic sampling in computer graphics. *ACM Trans. Graphics*, 5, 1 (January 1986), 51-72.
- [Dippé85] Dippé, M. A. Z. and Wold, E. H. Antialiasing through stochastic sampling. *Computer Graphics*, 19, 3 (July 1985), 69-78.
- [Dobkin92] Dobkin, D., and D. Eppstein. Computing the discrepancy. *Proceedings of Symposium on Computational Geometry*, (1993) , to appear.
- [Edelsbrunner86] Edelsbrunner, H., J. O'Rourke, and R. Seidel. Constructing arrangements of lines and hyperplanes with applications. *SIAM J. Comput.* 15, (1986), 341-363.
- [Edelsbrunner87] Edelsbrunner, H. *Algorithms in Combinatorial Geometry*. Springer-Verlag, Heidelberg, Germany, 1987.
- [Halton70] Halton, J. H. A retrospective and prospective survey of the Monte Carlo method. *SIAM Review*, 12, 1 (January 1970), 1-63.
- [Kajiya86] Kajiya, J. T. The rendering equation. *Computer Graphics* 20, 4, (August 1986), 143-150.
- [Lee85] Lee, M., Redner, R. A., Uselton, S. P. Statistically optimized sampling for distributed ray tracing. *Computer Graphics*, 19, 3 (July 1985), 61-67.
- [Mitchell87] Mitchell, D. P. Generating antialiased images at low sampling densities. *Computer Graphics*, 21, 4 (July 1987), 65-72.
- [Mitchell91] Mitchell, D. P. Spectrally optimal sampling for distribution ray tracing. *Computer Graphics*, 25, 4 (July 1991), 157-164.
- [Niederreiter78] Niederreiter, H. Quasi-Monte Carlo methods and pseudo-random numbers. *Bulletin of the American Mathematical Society*, 84, 6, (November 1978), 957-1041.
- [Niederreiter92] Niederreiter, H. Quasirandom sampling computer graphics. *Proc. 3rd Internat. Seminar on Digital Image Processing in Medicine*, Riga, Latvia (1992), 29-33.
- [Painter89] Painter, J., and Sloan, K. Antialiased ray tracing by adaptive progressive refinement. *Computer Graphics*, 23, 3 (July 1989), 281-288.
- [Purgathofer86] Purgathofer, W. A statistical model for adaptive stochastic sampling. *Eurographics '86* (1986), 145-152.
- [Shirley91] Shirley, P. Discrepancy as a quality measure for sample distributions.



- Proceedings of Eurographics '91*,
(1991) 183-193.
- [Warnock72] Warnock, T. T. Computational investigations of low-discrepancy point sets. *Applications of Number Theory to Numerical Analysis*, Academic Press (1972).
- [Whitted80] Whitted, T. An improved illumination model for shaded display. *Comm. ACM*, 23, 6 (June 1980), 343-349.

