EFFICIENT MONTE CARLO PROCEDURES FOR
GENERATING POINTS UNIFORMLY DISTRIBUTED
OVER BOUNDED REGIONS

by

Robert L. Smith

Technical Report #81-1

Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor, Michigan 48109

January 1981

Revised: January 1982

EFFICIENT MONTE CARLO PROCEDURES FOR THE

GENERATION OF POINTS UNIFORMLY DISTRIBUTED OVER

BOUNDED REGIONS

By

Robert L. Smith

We consider the Monte Carlo problem of generating points uniformly distributed within an arbitrary bounded (measurable) region. A class of Markovian methods are considered that generate points asymptotically uniformly distributed within the region. Computational experience suggests the methods are potentially superior to conventional rejection techniques for large dimensional regions.

Key Words: Monte Carlo, multivariate random deviates, uniform distribution, simulation.

Efficient Monte Carlo Procedures for Generating Points Uniformly Distributed Over Bounded Regions

1. INTRODUCTION

We consider the following problem. Given a bounded k-dimensional surface $S \subset R^n$ where $k \leq n$, how can we efficiently generate a pseudo-random point $(X_1, X_2, \ldots, X_n) \in S$ uniformly distributed over S. That is, we require that the probability $X \in A \subseteq S$ be given by $V(A)/V(S)$ where V is the k-dimensional content of (Jordan) measurable sets in S. The problem is actually quite general in that the seemingly broader problem of generating points X distributed over a region S according to a probability density function f over S is in fact equivalent. To see this, merely generate points Y uniformly distributed within the region under the graph of f over S and project Y onto S yielding X. X may be shown to have the desired properties.

Such a procedure is of interest for several reasons. As noted above, the procedure could be used in generating multivariate random deviates with arbitrary density functions. Efficient transformational procedures are currently limited to the multivariate normal and a few other specialized distributions (Rubinstein [16], Schmeiser [17]). Another potential application is in generating large numbers of feasible solutions to mathematical programs uniformly distributed within or on the boundary of the feasible region. These may serve as random starting points for heuristic optimum

seeking procedures (Dixon and Szego [4]) or may serve by them-
selves as stochastic probes yielding statistical information
on the value of the optimal objective function (Patel and
Smith [12], de Haan [3]).

2. BACKGROUND

Conventional approaches to generating uniformly distributed
points over bounded regions S are limited to transformational,
rejection, and composition techniques. We shall consider each in
turn together with their relative merits.

Transformation techniques directly capitalize on the fact
that extremely efficient pseudo-random number generators are widely
available for producing sequences of real numbers drawn from the
interval [0, 1] that have the statistical properties of being in-
dependent and identically distributed over their range (Hammersley
and Handscomb [8]). A set of k such numbers $X = (X_1, X_2, \ldots X_k)$
regarded as a point in R^k yields a pseudo-random point uniformly
distributed within the standard k-dimensional hypercube, H. A
transformation technique then maps X via a smooth deterministic
function T onto S. A necessary and sufficient condition that T(x)
be uniformly distributed over S is that the Jacobian of T be
constant over all x ε H and in fact equal to V(S)*. This is an
extremely efficient technique when T is easily computed. Unfortu-
nately, T is simple and known only for a limited class of regions
S. For example, for

---

*Generating a uniform distribution on the unit hypersphere by scaling
a multivariate normal vector of independent components can be viewed
as a composition map of a transformation of points in a unit cube
to normals followed by a cross-section transformation. However the
resulting transformation T being nonsquare has no determinant in
this case.

paralleletopes T is linear, for hyperspheres T is a simple closed form trigonometric function, and for simplices T is a weighted linear combination of the extreme points of S (Patel and Smith [13], Rubinstein [15]). Unfortunately, the latter idea does not generalize to polytopes. We could of course partition any bounded polytope into a finite union of simplices and use transformational techniques on the simplices (an example of a composition technique [Schmeiser [17]]). Although conceptually sound, the number of simplices and the difficulty of identifying them makes the procedure computationally intractable for even moderately large dimensional problems. Hsuan [9] suggests this method for polygons where the dimensionality is low enough for the procedure to be efficient.

Rejection techniques (Hammersley and Handscomb [8] , Rubinstein[15] rely on the simple principle that if a point is uniformly distributed within a region D enclosing a region S, then it is conditionally uniformly distributed within S given it lies in S. The procedure is then to enclose S by a region D for which a transformation technique is known. Using the transformation T to generate a point T(X) in D, T(X) is accepted if it also lies in S and is otherwise rejected. Each point so accepted will then be uniformly distributed within S. Typical choices for D include a rectangular paralleletope, a hyperphere, and a simplex. For example, if the region S were a bounded polytope, finding D as a paralletope would involve ascertaining maximal feasible ranges on the coordinate variables. Selecting D as a simplex or sphere would require solving a linear or quadratic program respectively. Depending on the complexity of S, one or more of these options may be computationally infeasible. A more serious problem of rejection techniques is that the number of trial points in D that must be generated to get a point in S grows explosively in the dimension of the region S. For example, when S is a hypercube and the enclosing

region D is a circumscribed hypersphere, the expected number of points that must be generated within D to get one within S grows from 1.5 for dimension k = 2 to 10^{30} for k = 100. The n-dimensional content of an n-dimensional sphere of radius r is $2r^n \pi^{n/2}(n\Gamma(n/2))^{-1}$ (Kendell [10]). Hence the content of the sphere enclosing an n-dimensional un cube is given by $2(n\pi/4)^{n/2} (n\Gamma(n/2))^{-1}$.

We will consider in the next section a class of procedures that recursively generates a sequence of points all within the region S. These points have the property that when the initial point is uniformly distributed within S then all are. Moreover, the last point generated is asymptotically independent of the first point as the sequence grows in length. In short, if we generate a long sequence of points and randomly mix their order, the resulting sequence of points has the statistical properties of being independent and identically distributed points uniformly distributed over S.

3. SYMMETRIC MIXING ALGORITHMS AND THEIR ASYMPTOTIC PROPERTIES

We define here a class of randomized algorithms that have the property of (strong) <u>reversibility</u>. That is, the probability density of moving from point x to point y in a region S is equal to the probability density of going from y to x. That is, were the algorithms run backwards in time, their probabilistic properties would remain unaltered and it would be impossible to distinguish which endpoint of the generated path was the origin. This property will be shown to imply a uniform stationary distribution over the points so generated. With a stronger assumption requiring that all subregions be eventually visited,

it will further be concluded that a uniform distribution over the region is asymptotically approached regardless of the starting point.

The class of algorithms to be considered is subsumed by the following general mixing algorithm. The character of a particular algorithm is imparted through a choice of the direction set D as described below.

Mixing Algorithm

1) Choose a starting point X_0 ε S and set i = 0.

2) Generate a random direction d uniformly distributed over a direction set D $\subseteq R^n$. Find the line set L = S ∩ {x | x = x_0 +λd, λ a real scalar} and generate a random point X_{i+1} uniformly distributed over L.

3) If i = N, stop. Otherwise, set i = 1 + 1 and return to 2.

Boneh and Golan [13] first proposed such an algorithm with D a hypersphere for the purpose of identifying non-redundant constraints in nonlinear programs. They also conjectured but did not prove that the points generated were uniformly distributed. Smith [13] later but independently proposed the same algorithm and demonstrated its asymptotic uniformity.

A symmetric mixing algorithm is a mixing algorithm that is reversible in the sense described above. For S an open set in R^n, every mixing algorithm is symmetric since the probability of traveling from x to a neighborhood of y is by construction the same as traveling from y to a neighborhood of x. Before cataloging the assumptions we will later need to impose on the class of symmetric mixing algorithms considered, we will first note that the points generated by a general mixing algorithm X_0, X_1, X_2,...constitute a continuous state Markov Chain for k > 0 and a discrete state Markov Chain for k = 0. Moreover, the chain is homogeneous. Let $P(A|x) = P(X_{i+1}$ ε $A|X_i = x)$

for measurable $A \subseteq S$ be the one step transition probability.
We will later impose one or both of the following regularity
conditions on $P(A|x)$ and, by implication, on the class of mix-
ing algorithms and regions considered. Roughly speaking, the conditior
require the algorithm to distribute points in a continuous
fashion over the entire region S.

Assumption: (a) There exists a measurable function
$f(y|x)$ on S x S such that
$$P(A|x) = \int_A f(y|x)\,dy \text{ for all}$$
measurable $A \subseteq S$.

(b) $f(y|x) > 0$ for all $x, y \in S$.

Note that for the class of symmetric mixing algorithms that
satisfy Assumption a), the transition
probability density function f is symmetric, i.e., $f(y|x)=f(x|y)$
for all $x, y \in S$. Roughly speaking, the probability of going
from x to y is the same as that of going from y to x since they
share the same line direction and the successor point is chosen
uniformly along the shared line segment.

Lemma 1: Under Assumption a), the uniform distribution over
S is a stationary initial distribution for the Markov Chain
induced by a symmetric mixing algorithm, i.e., $\lambda(A) =$
$$\int_S P(A|x)\,\lambda(dx) \text{ where } \lambda(A) = V(A)/V(S) \text{ with A measurable}$$
and V representing k-dimensional content over S.

Proof: For a symmetric mixing algorithm, $f(y|x) = f(x|y)$
and therefore

$$\int_S P(A|x)\ \lambda(dx) = \int_S \int_A f(y|x)\ \lambda(dy)\ \lambda(dx) =$$

$$\int_A \int_S f(x|y)\ \lambda(dx)\ \lambda(dy) = \int_A P(S|y)\ \lambda(dy)$$

$$= \int_A \lambda(dy) = \lambda(A).$$ The interchange of order of integration

is justified by Fubinis's Theorem since $f(y|x) \geq 0$ and the

double integral is finite. ■

Lemma 1 states that if the starting point X_0 were chosen uni-

formly from S, then X_i would remain uniformly distributed

over S for all $i \geq 0$.

Lemma 2: Let X_0, X_1, X_2, ... be the Markov Chain induced by

a symmetric mixing algorithm. Under Assumption a), b), the

chain is indecomposable, i.e., there are no two disjoint

closed* sets A_1, $A_2 \subseteq S$.

Proof: Suppose A_1 and A_2 are closed and $A_1 \cap A_2 = \phi$. Set

$\bar{A} = S - (A_1 \cup A_2)$. Since A_1, A_2, $\bar{A}$ form a partition of S,

$\lambda(A_1) + \lambda(A_2) + \lambda(\bar{A}) = 1$, and hence one of them has strictly

positive measure, say A_1. Since A_2 is non-empty, choose $x \in A_2$.

Then $P(A_1|x) = \int_{A_1} f(y|x)\ dy > 0$ since $f > 0$. Hence

$P(A_2|x) < 1$. Contradiction. Suppose now $\lambda(\bar{A}) > 0$. By the

same argument, $P(A_2|x) < 1$ for $x \in A_2$ and we get a contradic-

tion. The result follows. ■

---

*A non-empty set A is closed if it is almost surely not possi-
ble to leave it, i.e., $P(A|x) = 1$ for all $x \in A$.

<u>Theorem 1</u>: Let X_0, X_1, X_2, ... be the Markov Chain induced

by a symmetric mixing algorithm. Under Assumption a), b), the

uniform distribution λ is the unique stationary distribution

over S, and moreover, the Markov Chain X_0, X_1, X_2, ... with

initial distribution λ is ergodic, i.e., $\lim\limits_{m \to \infty} \frac{1}{m} \sum\limits_{i=0}^{m} \chi_A(X_i) = \lambda(A)$

for measurable $A \subseteq S$ where $\chi_A(x) = \begin{cases} 1 & \text{for } x \, \varepsilon \, A \\ 0 & \text{for } x \, \notin \, A \end{cases}$.

<u>Proof</u>:   From Lemma 2, we have that X_0, X_1, X_2, ... is inde-

composable and from Lemma 1 that λ is a stationary initial

distribution. It follows (see Proposition 7.11, p. 134 and

Theorem 7.16, p. 136, in Breiman [2]*) immediately that λ

is the unique stationary distribution and that the process

with λ as its initial distribution is stationary and in fact

ergodic. ■

Theorem 1 states that the sequence of points generated

by a symmetric mixing algorithm satisfying Assumption a), b)

will visit each subregion of S in the limit a fraction of time

equal to that region's fractional content. That is, the points

X_0, X_1, X_2, ... considered as an ensemble will be uniformly

distributed in the long run throughout the region S. Put

another way, a Chi-square frequency test would be unable to

statistically distinguish the points X_0, X_1, X_2, ... generated

by the symmetric mixing algorithm from those generated by

transformational or rejection techniques; i.e., their

---

* We refer throughout to Breiman [2] even though his results are
stated and proven for measurable $S \subseteq R^1$. The extension to R^n
is only notational since the ordering properties of R^1 are not
necessary to any of the results we use here.

observed long run frequencies of occurrence would be in agree-

ment with an independent and identically distributed uniform

model for X_0, X_1, X_2, ...

Theorem 2: Under Assumption a), b), the Markov Chain

X_0, X_1, X_2, ... induced by a symmetric mixing algorithm is

strongly mixing, i.e., $\lim_{m \to \infty} P(X_m \varepsilon A | X_0 = x) = \lambda(A)$ for all

starting points $x \varepsilon S$ and for all measurable sets $A \subseteq S$.

Proof: Doob [5] has shown that if a stationary initial dis-

tribution exists, then the chain is strongly mixing under the

following conditions: a) the state space S is indecomposable

under $P(A|x)$, b) the motion is nonperiodic, i.e., S is inde-

composable under the n-step transitions probability $P^{(n)}(A|x)$

n = 2, 3, ..., and c) for each $x \varepsilon S$, $P(A|x)$ is absolutely

continuous with respect to the stationary initial distribution.

In our case, a stationary initial distribution exists by

Lemma 1 and a) follows from Lemma 2. b) follows from the same

argument used in establishing Lemma 2 since $f^{(n)}(y|x)$

$\int_S f^{(n-1)}(z|x) f(y|z) dz > 0$ by induction on n. As for c),

$P(A.|x)$ is a bounded indefinite integral of a nonnegative

measurable function $f(y|x)$ and all indefinite integrals of

integrable functions are absolutely continuous with respect

to the defining measure of integration (Theorem H, p. 97,

Halmos [7]).∎

Theorem 2 implies that the serial correlation between any two points X_k and X_{k+M} goes to zero as the number of iterations separating them M grows large. This follows since X_k and X_{k+M} are asymptotically independent in the limit as M goes to infinity. It follows that shuffling (randomly permuting) the indices in a large number of generated points X_0, X_1,...,X_M to produce $X_{(0)}$, $X_{(1)}$,....,$X_{(M)}$ will induce the same uncorrelated effect for <u>any</u> pair of distinct points $X_{(k)}$ and $X_{(k')}$. That is, the shuffled sequence $X_{(0)}$, $X_{(1)}$,....,$X_{(M)}$ will not only exhibit frequencies in accordance with a uniform model, but will also exhibit a lack of serial correlation in agreement with an independent model. In short, the statistical properties of $X_{(0)}$, $X_{(1)}$,....,$X_{(M)}$ will for large M approximate those of M i.i.d. uniform random deviates. An implication of this is that all statistics that are invariant under permutations of the data used in their calculation (e.g., a minimum or an average) may be based directly on the original sequence X_0, X_1,...,X_M but will none-theless have the statistical properties suggested by an i.i.d. uniform model for X_0, X_1,...,X_M. A very important issue in all this is of course how large M must be. We will explore this issue in the last section of the paper.

We will now consider specific algorithms corresponding to different choices of the direction set D all of which are symmetric algorithms that moreover satisfy Assumption a), b) for a large class of regions S.

A Random Directions Algorithm (Boneh and Golan [13], Smith [18]):

Set $D = \{d \in R^n \mid ||d|| = 1\}$. Then selecting d uniformly over the unit sphere D is the operational equivalent of selecting a random direction. Such a d is easily computed by any of a number of techniques. Perhaps the simplest is to generate n independent normally distributed random deviates $N = (N_1, N_2, \ldots, N_n)$ and set $d = N/||N||$ (see e.g., Knuth [11], p. 116).

Establishing the most general class of regions S for which Assumption a), b) holds for this algorithm is a difficult task that we will not attempt here. However, Assumption a), b) is established in the next section for all open regions $S \subseteq R^n$. In particular, S is assumed to be of full dimension n. Arguments for lower dimensional surfaces are not difficult if S is specialized further. For example, for S the surface of a n-dimensional polytope, it can be argued directly that $P(A|x)$ can be written as an integral of a positive probability density function over S. However we lose symmetry unless the algorithm is altered to stall at the current point favoring movement in directions that meet facets less orthogonally. Such a random "bouncing" over the facets of S will by Theorems 1 and 2 be asymptotically uniformly distributed over S. The implied search procedure may have merit for ill structured but small dimensional mathematical programs.

A Coordinate Directions Algorithm (Telgen [19]):

Set $D = \{d \in R^n: d = \pm\, e^i$ for some $i = 1, 2,\ldots,n\}$, where $e^i_j = 1$ for $j = i$ and 0 otherwise}. Although this algorithm was independently suggested by several people (including this author) it was Telgen [19] who first noted the algorithms considerable computational advantage over the Random Directions Algorithm on a per iteration basis. The difference is particularly compelling in the important case where $S = \{x: Ax \leq b\}$. As noted by Telgen [19], the full sparsity of A may be exploited in every iteration in determining the end points of the line segment L. On the other hand, the number of iterations necessary to converge to the uniform distribution may offset this advantage, although preliminary computational experience suggests the converse.

Establishing Assumption a), b) for this algorithm is complicated by the fact that $P(A|x)$ is typically not absolutely continuous with respect to the uniform distribution over S. The probability mass is concentrated along the coordinate axes which are of zero Lebesque measure for open regions S. However, if the infimum of all line set lengths over open S is positive, then every open subregion may be reached in a finite number of iterations M_0. It can then be argued that the M_0-step transition probability $P^{(M_0)}(A|x)$ satisfies Assumption a), b). It follows that the sequence of points $X_0, X_{M_0}, X_{2M_0},\ldots$ is asymptotically uniform over such regions S. Actually, a stronger claim may be made if we approximate S by a rectangular mesh oriented along a rectangular coordinate system. Such an approximation has no

practical restriction since computer accuracy is finite. The
coordinate directions algorithm now generates a finite state
Markov Chain of points X_0, X_1, X_2, ... where the number of
states N is the number of rectangular parallelepipeds in
the mesh. The corresponding transition probability matrix
(P_{ij}) is symmetric, irreducible, and aperiodic. It follows
that (P_{ij}) is doubly stochastic, and hence (Ross [14]) that
$\lim_{m \to \infty} P_{ij}^m = \frac{1}{N}$ for all i, j = 1, 2,...,N.

A Vertex Reaching Algorithm:

To keep the discussion simple, suppose P is an n-dimen-
sional simple polytope with vertices V_1, V_2,...,V_r so that
each vertex has exactly n neighboring vertices. Let
$S = \{V_1, V_2,...,V_r\}$ and think of P transformed
so that the direction set $D = \{d_1, d_2,...,d_n\}$ can be con-
structed of directions that point to the n adjacent vertices
regardless of the vertex V_i we are currently at. Then the
dimension k of S is zero and k-dimensional content reduces to
the cardinality of subsets of S. Hence $\lambda(V_i) = \frac{1}{r}$ for all i
and clearly f(x, y) = 1 for all x, y ε S. From Theorems 1 and 2, the
procedure of starting at any vertex of P and randomly moving
along edges to adjacent vertices in an unbiased way will
asymptotically "lose" one among the vertices of P. Since the
vertices visited are in the long run probabilistically equi-
valent to i.i.d. uniform choices from S, the distribution of
the number of vertices visited until a distinguished vertex

is visited (say the optimum vertex in a concave minimization problem) approximately follows a geometric distribution. In particular, the expected number of vertices visited before success is the number of vertices r. Such an analysis is not performed to recommend such a search procedure, but rather to illustrate how Theorems 1 and 2 can aid in the probabilistic analysis of the general class of randomized algorithms we are considering.

4. CONVERGENCE RATES FOR THE RANDOM DIRECTIONS ALGORITHM

The practical merit of the symmetric mixing algorithms will of course be dependent on their rate of convergence to the uniform distribution. We have explicitly calculated upper bounds on the rate of convergence to uniform for the random directions algorithm and the results are reported in this section. The formula for the bound gives considerable insight on what problem characteristics are important in determining convergence rate. However the bound is of computational interest only for low dimensional problems. It can over-estimate the number of iterations required to achieve uniformity by several orders of magnitude as we shall see below.

Theorem 3: Let X_0, X_1, X_2, ... be the Markov Chain generated by the Random Directions Algorithm over an open region $S \subset R^n$.

Then for any measurable set $A \subseteq S$, $|P(X_m \varepsilon A | X_0 = x) -$

$\lambda(A)| < \left(1 - \frac{\gamma}{n2^{n-1}}\right)^{m-1}$ where γ is the ratio of the n-dimensional

content of S to the n-dimensional content of the smallest sphere

containing S.

<u>Proof</u>: From Case (b), p. 197 in Doob [6] $|P(X_m \varepsilon A | X_0 = x)$

$- \lambda(A)| \leq (1 - \delta\phi(C))^{m-1}$ where ϕ is a finite measure on a σ-field

over S and C is a measurable set in S for which $\phi(C) > 0$ and

$f(y|x) \geq \delta > 0$ for all x, y ε C where $f(y|x)$ is the density of

the absolutely continuous component of P with respect to ϕ. In

our case ϕ is n-dimensional content V. To obtain the strongest

bound we set C equal to S. Our task then is to show $f(y|x)$ exists

and hence that P is itself absolutely continuous and to find $\delta > 0$.

$f(y|x) = \lim_{V(A) \to 0} \frac{P(A|x)}{V(A)}$ if this limit exists for y ε A. Take A as

an n-dimensional cube of content ℓ^n oriented along the ray from

x to y. For ℓ small, $P(A|x) \approx \left(\frac{2\ell^{n-1}}{S_n(r(x, y))}\right)\left(\frac{\ell}{d(x,y)}\right)$ where $r(x, y)$ is

the distance from x to y, $d(x, y)$ is the diameter of S along the

ray from x to y, and $S_n(x)$ is the surface area of an n-

dimensional sphere of radius x. Hence $f(y|x) = \lim_{\ell \to 0} P(A|x)/\ell^n =$

$2/S_n(r(x, y))d(x, y)$ exists. Moreover, $r(x, y) \leq d$ and

$d(x, y) \leq d$ where $d = \max_{x, y \varepsilon \bar{S}} d(x,y)$. Hence $f(y| x) > \delta = 2/dS_n(d)$

for all x, y ε S. Since $S_n(d) = n2^n V_n(d/2)/d$ where $V_n(x)$ is

the volume of an n-dimensional sphere of radius x (Kendall [10],

p. 35), we get $\delta = 1/n2^{n-1}V_n(d/2)$. But $V_n(d/2)$ is the volume

of a circumscribed sphere around S and hence the result follows

since $\phi(C) = V(S)$. ∎

As a numerical illustration of Theorem 3, consider the case of S as a disk in R^2. Then $|P(X_m \varepsilon A \mid X_0 = x) - \lambda(A)| \leq (3/4)^{m-1}$. The convergence as in all cases is exponentially fast. For example, we will be within 1% of a uniform distribution over S after 17 iterations of the random directions algorithm regardless of starting point. Clearly, some starting points X_0 are more conducive to rapid convergence than others, and for this as well as other reasons, 17 iterations is only an upper bound on the required number.

It is interesting that the ratio γ of the content of S to an enclosing sphere plays a role here in algorithm effi-ciency just as in the standard rejection technique. However, a large number of iterations to achieve uniformity in the random directions algorithm case also implies a correspondingly large number of uniform points corresponding to each of the iterations. However for the rejection technique it is only the last (feasible) point that is retained.

Of course, the practical merit of the approach rests on its efficiency for large dimensional problems. The Random Directions Algorithm was therefore run on a 10 dimensional hypercube region with initial point randomly chosen according to a uniform distribution. We chose this regular shape in order to easily test for uniformity of the iteration points generated. It should be noted that although a cube is the easiest of all regions to generate uniform points within by conventional approaches, the measure of difficulty for the Random Directions Algorithms is how different the region is from an enclosing

sphere. The ratio of their corresponding contents in this case is roughly 250 to 1. This gives a theoretical upper bound of over five million iterations to get within an ε of 1% of uniformity. We, however, ran the algorithm for only 10,000 iterations. To avoid serial correlation effects we first sampled every 10th point generated. We then shuffled this smaller population of 1000 points so that their order became random. It should be noted that the points were shuffled but not their components. We then performed Chi-square frequency and serial correlation tests ([Knuth [11], p. 35) on this new sequence of 1,000 points. The cells corresponded to 10 equal volume slabs in the X_i direction for i = 1, 2,...,10. These correspond to 10 Chi-square tests in all. It should be noted that the X_i direction is a favored direction only with respect to the shape of the region and not the performance of the algorithm which is blind to coordinate orientations. Hence aspects of multivariate as opposed to only marginal uniformity are being tested. The results are reported in Tables 1 and 2.

i\j	1	2	3	4	5	6	7	8	9	10	Freq. χ^2 Statistic (ν = 9 d.f.)	Passed Uniformity at α = 10% level of significance
1	88	98	86	82	132	104	102	104	116	88	21.3	No
2	97	122	99	105	94	94	94	99	96	100	6.4	Yes
3	81	84	103	99	107	106	91	117	104	108	11.6	Yes
4	111	112	124	122	110	89	85	83	85	79	27.2	No
5	115	107	98	106	97	93	94	93	92	105	5.5	Yes
6	123	99	91	102	106	99	93	91	92	104	8.6	Yes
7	106	84	106	92	82	96	115	94	108	117	13.5	Yes
8	95	94	110	91	107	74	88	121	114	106	17.8	No
9	94	91	98	103	94	91	116	111	105	97	6.6	Yes
10	116	100	92	99	95	99	115	102	90	92	7.4	Yes

Observed Frequency f_{ij} of Points in Slab j in Coordinate Direction X_i Out of a Total of 1,000 Points

Upper and Lower χ^2 values for α = 10%, ν = 9: (3.3, 16.9)*

Table 1 - Chi-Square Frequency Test Results for 1000 Points Generated in a 10 Dimensional Cube.

Coordinate Direction X_i	Serial χ^2 Statistic (ν = 99 d.f.)	Passed Serial Correlation at α = 10% Level of Significance
1	107.6	Yes
2	85.6	Yes
3	105.6	Yes
4	111.6	Yes
5	84.4	Yes
6	96.0	Yes
7	125.6	No
8	103.6	Yes
9	82.4	Yes
10	103.6	Yes

Upper and Lower χ^2 Values for α = 10%, ν = 99: (77.9, 124.3) *

Table 2 - Chi-Square Serial Test Results for 1,000 Points Generated in a 10 Dimensional Cube.

*Two tailed Chi-square tests were performed since true distributional uniformity would have an observable variance component in a departure from deterministic uniformity.