

CLOSE-POINT SPATIAL TESTS AND THEIR APPLICATION TO RANDOM NUMBER GENERATORS

PIERRE L'ÉCUYER, JEAN-FRANÇOIS CORDEAU, and RICHARD SIMARD

Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, C.P. 6128, Succ. Centre-Ville, Montréal, H3C 3J7, Canada
lecuyer@iro.umontreal.ca

(Received March 1997; revisions received December 1997, July 1998; accepted August 1998)

We study statistical tests of uniformity based on the L_p -distances between the m nearest pairs of points, for n points generated uniformly over the k -dimensional unit hypercube or unit torus. The number of distinct pairs at distance no more than t , for $t \geq 0$, is a stochastic process whose initial part, after an appropriate transformation and as $n \rightarrow \infty$, is asymptotically a Poisson process with unit rate. Convergence to this asymptotic is slow in the hypercube as soon as k exceeds 2 or 3, due to edge effects, but is reasonably fast in the torus. We look at the quality of approximation of the exact distributions of the tests statistics by their asymptotic distributions, discuss computational issues, and apply the tests to random number generators. Linear congruential generators fail decisively certain variants of the tests as soon as n approaches the square root of the period length.

Multidimensional goodness-of-fit tests based on nearest-neighbor distances have been proposed and studied in the statistical literature; see Bickel and Breiman (1983), Cressie (1993), Ripley (1977, 1988), and other references therein. Given a set of points in space, these tests seek evidence against the hypothesis that the points are an i.i.d. sample from a specified multivariate distribution. They are based, for example, on the distance separating the nearest pair of points, or on the number of pairs that are less than a given distance apart, or on some more general function of the distances from each point to its nearest neighbor. In most cases, the distribution of the relevant statistic under the null hypothesis is known asymptotically as the number of sample points increases to infinity. Otherwise, this distribution is often estimated by Monte Carlo simulation (Ripley 1988, Schilling 1983a).

In this paper, we study different aspects of such goodness-of-fit tests from an empirical perspective. First, we introduce new test variants. Second, we assess how well the exact distribution of the test statistic can be approximated by the asymptotic one, as a function of the test parameters. Third, we look at the issue of computational costs for large sample sizes and in large dimensions. Fourth, we study how certain types of random number generators (RNGs) behave with respect to these tests. No statistical test can ever prove that a given generator is flawless, but the tests may improve somewhat our confidence in a generator, or in some cases destroy this confidence completely. For more on random number generators and their testing, see, for example, Knuth (1981), L'Écuyer (1992, 1994), Marsaglia (1985), and Niederreiter (1992).

The null hypothesis here is that the points produced by the uniform RNGs are independently and uniformly distributed in the unit hypercube. The tests are designed to detect clustering or repulsion between these points. Several types of simulations (e.g., in spatial statistics, computational geometry,

operations research, etc.) involve random points in multidimensional space and their results, in many cases, can be sensitive to excessive clustering or repulsion. For example, the average value of the objective function in a stochastic vehicle routing problem can be affected by too much clustering between the (random) customer's locations. It is therefore highly desirable that the batteries of empirical tests, to which the generators are submitted before being placed into general-purpose software packages, contain tests that are sensitive to such clustering, repulsion, or other spatial regularities that may adversely affect the results of the users.

Other tests commonly applied to RNGs, such as the serial test and the collision test (Knuth 1981), can also detect spatial clustering in the unit hypercube. However, they have less power than the specially designed tests studied here against alternatives that correspond to regular structures of commonly used generators. Consider for example the following standard *serial test*. Partition the two-dimensional unit hypercube into 2^{14} cubic cells, generate 2^{18} random points, count how many fall in each cell, and apply a standard chi-square test. The commonly used linear congruential generators (LCGs) of period length $2^{31} - 1$ will pass this test easily most of the time. With some close-pair tests described in this paper, all these LCGs fail repeatedly with 2^{18} points at significance level 10^{-15} .

We now give an overview of the paper. In §1, we introduce tests of uniformity based on close pairs of points in the k -dimensional unit hypercube or unit torus. These tests throw n random points, compute the L_p -distances between the pairs of points that are close to each other, and sort these distances by increasing order. For a specific function φ_n , the number of distinct pairs at a distance less than $\varphi_n(t)$, as a function of t , is approximately a Poisson process with unit rate. Saunders and Funk (1977) suggested certain tests on this Poisson process, but here we consider different ones. We also consider *two-level tests*, where the entire

Subject classifications: Simulation, random number generation, statistical tests, goodness-of-fit, spatial statistics.

Area of review: SIMULATION.

procedure is repeated N times and the distribution of the N p -values is compared with the uniform. As a special case, one can look only at the distance D_n^* between the two nearest points. This D_n^* generalizes d_1 in Ripley and Silverman (1978) and D^k in Ripley (1987, Theorem 2.6). For RNGs with a regular structure, the N replicates of D_n^* tend to cluster. We propose certain transformations that enhance the ability of the test to detect such clustering. Bickel and Breiman (1983) introduced a goodness-of-fit test specifically designed to test the hypothesis of a given multivariate density in the unit hypercube. Because this test is also based on distances between close points in the hypercube, it is natural to compare it with our tests and we do so.

In practice all these tests use the asymptotic theoretical distributions for $n \rightarrow \infty$. In §2, we examine how well these asymptotics approximate the exact finite-sample-size distributions. To reduce the approximation error, we recommend computing the distances in the torus instead of the hypercube. The effect of the approximation error on the tests is estimated by the Anderson-Darling distance between the exact and asymptotic distributions. These error assessments are interesting in general, not only for RNG testing.

In §3, we give CPU timings for computing the m shortest interpoint distances for different values of n, k , and p . In large dimensions (say, $k \geq 8$), the computations are much faster for large p (up to $p = \infty$) than for small p .

In §4, we first apply a set of tests to a small selection of well-known RNGs. We find that all the LCGs fail and all the other selected generators pass. The fact that all the LCGs fail is no big surprise, because their regular lattice structure implies that the small interpoint distances have a limited number of possible values. Moreover, if the lattice structure of the LCG is good, there is a large lower bound on D_n^* , independent of n (Ripley 1987, p. 26). Therefore, for large n , the nearest-pair distance will be larger than expected. To see the effect of this on the test results, we selected LCGs of different prime period lengths, ranging (approximately) from 2^{14} to 2^{40} , and observed at which sample size they started to fail the tests decisively. This gives an idea of how soon the structure of these generators begins to affect the results of simulations involving similar random variables. We also give examples of generators whose bad structure is detected by the tests only by large-dimensional tests.

1. CLOSE-PAIR TESTS OF UNIFORMITY

1.1. L_p -Distance in the Unit Hypercube and Torus

We consider the k -dimensional unit hypercube $[0, 1]^k$, with the L_p -norm:

$$\|X\|_p = \begin{cases} (|x_1|^p + \dots + |x_k|^p)^{1/p} & \text{if } 1 \leq p < \infty, \\ \max(|x_1|, \dots, |x_k|) & \text{if } p = \infty, \end{cases}$$

for $X = (x_1, \dots, x_k) \in \mathbb{R}^k$. The distance between two points X_1 and X_2 in \mathbb{R}^k is thus $\|X_2 - X_1\|_p$. One obtains a torus by identifying (pairwise) the opposite sides of the

unit hypercube. Points that are face to face on opposite sides are thus considered close to each other. This is equivalent to replacing the L_p norm by

$$\|X\|_p^o = \begin{cases} [\min(|x_1|, 1 - |x_1|)^p + \dots + \min(|x_k|, 1 - |x_k|)^p]^{1/p} & \text{if } 1 \leq p < \infty, \\ \max(\min(|x_1|, 1 - |x_1|), \dots, \min(|x_k|, 1 - |x_k|)) & \text{if } p = \infty, \end{cases}$$

defined for vectors X whose coordinates belong to the interval $[-1, 1]$. The following applies with either of these two norms.

1.2. A Poisson Process Approximation for the Close-Pair Countings

Let X_1, \dots, X_n be n points in $[0, 1]^k$. Let $D_{n,i,j} = \|X_j - X_i\|_p$ be the distance between X_i and X_j , $D_{n,i} = \min_{j \neq i} D_{n,i,j}$ the distance from X_i to its nearest neighbor, and $D_n^* = \min_{1 \leq i \leq n} D_{n,i}$ the distance between the two nearest points. Define the null hypothesis \mathcal{H}_0 : “ X_1, \dots, X_n are i.i.d. random variables uniformly distributed over $[0, 1]^k$.”

Put $\lambda(n) = n(n - 1)V_k(1)/2$, where $V_k(r) = [2r\Gamma(1 + 1/p)]^k / \Gamma(1 + k/p)$ is the volume of the ball $\{x \in \mathbb{R}^k \mid \|x\|_p \leq r\}$, and Γ is the usual gamma function. For each $t \geq 0$, let $Y_n(t)$ be the number of distinct pairs of points (X_i, X_j) , with $i < j$, such that $D_{n,i,j} \leq (t/\lambda(n))^{1/k}$.

PROPOSITION 1. *In the unit hypercube or the unit torus, take any fixed $t_1 > 0$ and assume that \mathcal{H}_0 holds. Then, as $n \rightarrow \infty$, $\{Y_n(t), 0 \leq t \leq t_1\}$ converges weakly to a Poisson process with unit rate truncated to the interval $[0, t_1]$. Moreover, in the unit torus, for $t \leq \lambda(n)/2^k$, the exact mean and variance of $Y_n(t)$ are: $E[Y_n(t)] = t$ and $\text{Var}[Y_n(t)] = t - 2t^2/(n(n - 1))$.*

PROOF. The first part follows from the results of Saunders and Funk (1977) or Silverman and Brown (1978), after making the appropriate change of variables. For the second part, let $I_{i,j} = I[\lambda(n)D_{n,i,j}^k \leq t]$ for $i < j$, where I is the indicator function. In the torus, these $I_{i,j}$ are pairwise (but not mutually) independent Bernoulli random variables with parameter $q = P[I_{i,j} = 1] = P[D_{n,i,j} \leq (t/\lambda(n))^{1/k}] = V_k((t/\lambda(n))^{1/k}) = tV_k(1)/\lambda(n) = 2t/(n(n - 1))$. Then,

$$E[Y_n(t)] = \sum_{i=1}^{n-1} \sum_{j=i+1}^n E[I_{i,j}] = \frac{n(n - 1)}{2} q = t.$$

Because the $I_{i,j}$ are pairwise independent,

$$\begin{aligned} \text{Var}[Y_n(t)] &= \sum_{i=1}^{n-1} \sum_{j=i+1}^n \text{Var}[I_{i,j}] \\ &= \frac{n(n - 1)}{2} q(1 - q) = t - \frac{2t^2}{n(n - 1)}. \quad \square \end{aligned}$$

Let $T_{n,i} = \inf\{t \geq 0 \mid Y_n(t) \geq i\}$, $i = 1, 2, 3, \dots$, be the jump times of Y_n , with $T_{n,0} = 0$. Proposition 1 implies that for any fixed integer $m > 0$, for large enough n , the random variables

$$W_{n,i}^* = 1 - \exp[-(T_{n,i} - T_{n,i-1})], \quad 1 \leq i \leq m$$

are approximately i.i.d. $U(0, 1)$. In the unit torus, the first moment of $Y_n(t)$ matches exactly that of the Poisson process with unit rate, and the second moment matches up to $O(t^2n^{-2})$. But this is not true in the hypercube (the $I_{i,j}$ are not pairwise independent in that case). Exact formulas for the mean and variance of the $W_{n,i}^*$ in the torus or the hypercube are not available.

1.3. Goodness-of-Fit Tests Based on the Poisson Process

The test statistic $W_{n,1}^*$ was proposed by Ripley (1977) for $p = 2$ and further studied in Ripley and Silverman (1978), Silverman and Brown (1978), and other references given therein. Saunders and Funk (1977) suggest using $G_n^+(t_0, t_1) = \sup_{t_0 \leq t \leq t_1} Y_n(t)/t$, for $0 < t_0 < t_1$ fixed. This statistic tends to inflate when the points are more clustered than they should be. To detect the opposite situation of too much repulsion between the points, one may consider the companion statistic $G_n^-(t_0, t_1) = \inf_{t_0 \leq t \leq t_1} Y_n(t)/t$. These two statistics have complicated discontinuous distributions that can be derived from the results of Pyke (1959). In our experiments, they were less sensitive than those that we now introduce.

As a first alternative, we simply compare the empirical distribution of $W_{n,1}^*, \dots, W_{n,m}^*$ to the uniform, using the Anderson-Darling (AD) goodness-of-fit statistic. If $U_{(1)} \leq \dots \leq U_{(m)}$ denote the values of $W_{n,1}^*, \dots, W_{n,m}^*$ sorted by increasing order and \hat{F} their empirical distribution, then the AD statistic is defined by

$$A_m^2 \stackrel{\text{def}}{=} \int_0^1 \frac{(\hat{F}(u) - u)^2}{u(1-u)} du = -m - \frac{1}{m} \sum_{j=1}^m \{(2j-1)\ln(U_{(j)}) + (2m+1-2j)\ln(1-U_{(j)})\}. \tag{1}$$

The null hypothesis $\check{\mathcal{H}}_0$: “The $W_{n,i}^*$ are i.i.d. $U(0, 1)$ random variables” is rejected when the value of A_m^2 is too large. This hypothesis differs from \mathcal{H}_0 , but holds approximately when \mathcal{H}_0 holds and n is large enough. We call this test the *m-nearest-pairs* (*m-NP*) test. For the distribution of A_m^2 under $\check{\mathcal{H}}_0$, we refer to Durbin (1973).

1.4. Two-Level Tests

Increasing the sample size n normally increases the power of the test and decreases the approximation error (i.e., the difference between \mathcal{H}_0 and $\check{\mathcal{H}}_0$). However, the computing time for the m nearest pairs increases faster than linearly with n (see §3). The size of the memory to store the points also becomes a problem when n gets too large. Eventually, instead of increasing n further, one would rather replicate independently the entire procedure, say N times, then compute the N values of A_m^2 and test the fit of their distribution to the AD distribution. If F_m denotes the theoretical distribution of A_m^2 under $\check{\mathcal{H}}_0$, then the random variable

$\delta = 1 - F_m(A_m^2)$ is the p value of the first-level m -NP test. The N p values $\delta_1, \dots, \delta_N$ are i.i.d. $U(0, 1)$ under $\check{\mathcal{H}}_0$. At the second level, the AD statistic is

$$A_N^2 = -N - \frac{1}{N} \cdot \sum_{i=1}^N \{(2i-1)\ln(\delta_{(i)}) + (2N+1-2i)\ln(1-\delta_{(i)})\}, \tag{2}$$

where $\delta_{(1)}, \dots, \delta_{(N)}$ are the δ_i sorted by increasing order. One rejects $\check{\mathcal{H}}_0$ if the p value $\delta' = 1 - F_N(A_N^2)$ is deemed too small. This setup fits the paradigm of the two-level tests that are commonly used in RNG testing (Fishman 1996, Knuth 1981).

1.5. The Nearest-Pair Test

As an interesting special case, take $m = 1$; that is, compute only the distance between the nearest pair of points, and repeat this N times, independently. This yields N replicates of $W_{n,1}^*$, which we denote by $U_{(1)} \leq \dots \leq U_{(N)}$ after they are sorted by increasing order. One can then test the null hypothesis $\check{\mathcal{H}}_0$: “The N replicates of $W_{n,1}^*$ are i.i.d. $U(0, 1)$ random variables,” via the AD statistic A_N^2 defined by replacing m with N in (1). The hypothesis is rejected if $\delta_0 = 1 - F_N(A_N^2)$ is too small. We call this the *nearest-pair* (NP) test.

The NP test is not exactly equivalent to computing the p value δ' of the statistic (1) for the 1-NP test replicated N times. In the 1-NP test, δ is small when $U = W_{n,1}^*$ is either close to 0 or close to 1, and δ is close to 1 when U is close to 1/2 (one has $\delta = \min(2U, 2(1-U))$ when $m = 1$). It is thus more appropriate to use δ_0 rather than δ' when $m = 1$ to avoid spoiling the power of the test for detecting values of $W_{n,1}^*$ close to 1/2.

1.6. Spacings, Power Ratio, and Other Transformations

When the points produced by an RNG have a structure that is too regular, the small values of $D_{n,i,j}$ often tend to cluster. This happens in particular with LCGs (see §4). This means that the jumps of Y_n tend to cluster, so many $W_{n,i}^*$ are very close (or equal) to zero. This produces a large value of A_m^2 (see the definition of A_m^2). For the NP test, the N replicates of $W_{n,i}^*$ typically tend to cluster, but not necessarily near zero or one. This does not affect A_N^2 as much as A_m^2 in the m -NP test. To make the effect of clustering more easily detectable by A_N^2 , we consider two types of transformations on the observations: The *spacings* and the *power ratio*. Both are well known in statistics (e.g., Stephens 1986), but their use in conjunction with the NP test seems new.

Consider again $U_{(1)}, \dots, U_{(N)}$, the N replicates of $W_{n,1}^*$ sorted by increasing order. Define the *spacings* $S_i = U_{(i+1)} - U_{(i)}$, $i = 0, \dots, N$, where $U_{(0)} = 0$ and $U_{(N+1)} = 1$. If the $U_{(i)}$ tend to cluster, several of these S_i will take very small values. Now, one can transform these spacings

into a new set of spacings as follows. Sort S_0, \dots, S_N to obtain $S_{(0)} \leq \dots \leq S_{(N)}$, and then compute the weighted differences

$$S'_0 = (N + 1)S_{(0)},$$

$$S'_i = (N - i + 1)(S_{(i)} - S_{(i-1)}), \quad 1 \leq i \leq N.$$

Under $\tilde{\mathcal{H}}_0$, the vector (S'_0, \dots, S'_N) has the same distribution as (S_0, \dots, S_N) (see, e.g., Pyke 1965). Consequently, $\{U'_i = S'_0 + \dots + S'_{i-1}, 1 \leq i \leq N\}$ forms a new sample of N i.i.d. uniforms. Now, if several S_i are close to zero, the first several S'_i and U'_i will also be close to zero. As a result, if the AD test is applied to the U'_i , A_N^2 will be large.

The spacings transformation can be iterated to obtain a new set of spacings S''_0, \dots, S''_N , new uniforms U''_1, \dots, U''_N , and so on. We tried this, but the second and subsequent iterations did not increase the power in our experiments. In a similar vein, we tried applying the spacings transformation to the (sorted) values of $W_{n,1}^*, \dots, W_{n,m}^*$ in the m -NP test, and this tended to *reduce* the power of the test. When several observations are already concentrated near zero or one, further iterations of the spacings do not help.

The other transformation that we consider is the *power ratio* (see Stephens 1986):

$$U'_i = (U_{(i)}/U_{(i+1)})^i, \quad i = 1, \dots, N.$$

Under $\tilde{\mathcal{H}}_0$, these U'_i form a new sample of i.i.d. $U(0, 1)$. If the $U_{(i)}$ are clustered, this will produce several U'_i close to one. This transformation can also be iterated, but we found no advantage in applying it more than once in our context, just as for the spacings transformation.

1.7. The Bickel-Breiman Statistic

As a goodness-of-fit test for an arbitrary density in the Euclidean space, Bickel and Breiman (1983) proposed a nearest-neighbor statistic, which in the case of the uniform density becomes

$$B_n = \sum_{i=1}^n (W_{n(i)} - i/n)^2, \quad (3)$$

where $W_{n(1)} \leq \dots \leq W_{n(n)}$ are the ordered values of the $W_{n,i} = 1 - \exp[-nV_k(D_{n,i})]$, $1 \leq i \leq n$. Their development works with any norm, and for a nonuniform density the $W_{n,i}$ are simply defined in a more general way. These $W_{n,i}$ are approximately $U(0, 1)$ under \mathcal{H}_0 , and B_n measures their deviation from uniformity.

The theoretical distribution of B_n under \mathcal{H}_0 is hard to obtain. As $n \rightarrow \infty$, $B_n \Rightarrow \int_0^1 Z^2(t) dt$, where Z is a Gaussian process with mean zero and complicated covariance function (Bickel and Breiman 1983, equation (5.13)) that depends on k . Schilling (1983b) was able to compute that covariance for $k = 1$ and for $k \rightarrow \infty$, and obtained the limit distribution of B_n under \mathcal{H}_0 as both k and n go to infinity. For finite $k > 1$, the distribution of B_n can be estimated by simulation, as did Schilling (1983b).

Table 1. Error estimates in the hypercube, for $p = 2$.

n	k	N	$\hat{\Delta}_{0,n}$	n	k	N	$\hat{\Delta}_{0,n}$
10	2	10^5	40				
25	2	10^5	15				
10^2	2	10^5	1				
10^3	2	10^5	1	10^4	2	10^4	0
10^3	4	10^5	40	10^4	4	10^4	2
10^3	6	10^5	$\approx 1,000$	10^4	6	10^4	16
10^3	9	10^5	$\approx 10,000$	10^4	9	10^4	350

2. QUALITY OF THE APPROXIMATION FOR FINITE N

If n is not large enough, the difference between the distribution of our test statistic under \mathcal{H}_0 and that under $\tilde{\mathcal{H}}_0$ or $\check{\mathcal{H}}_0$ could be substantial, leading to too many false rejections. We did a simulation study to assess (crudely) what the minimal n should be in terms of p , k , m , and N for the different tests. For this and all the other simulation studies in this paper, we used the generator G15 of §4. We also checked the results with G14 and G16, and they agreed.

To explain what we did, consider the NP test. The test statistic (2) satisfies

$$A_N^2 = \int_0^1 \frac{(\hat{F}(u) - u)^2}{u(1-u)} du \leq \int_0^1 \frac{(\hat{F}(u) - F_{0,n}(u))^2}{u(1-u)} du + \int_0^1 \frac{(F_{0,n}(u) - u)^2}{u(1-u)} du, \quad (4)$$

where $F_{0,n}(u) \stackrel{\text{def}}{=} P[W_{n,1}^* \leq u | \mathcal{H}_0]$. Denote the last two integrals in (4) by $\tilde{A}_{N,n}^2$ and $\Delta_{0,n}$, respectively. ($F_{0,n}$ and $\Delta_{0,n}$ depend on n and k , but are independent of N .) This $\tilde{A}_{N,n}^2$ is the AD statistic that we *should* calculate if we knew $F_{0,n}$, whereas $\Delta_{0,n}$ is the Anderson-Darling distance between the uniform distribution and the true distribution of $W_{n,1}^*$ under \mathcal{H}_0 . This $\Delta_{0,n}$ is an upper bound on the error $A_N^2 - \tilde{A}_{N,n}^2$.

Because $F_{0,n}$ is unknown, we cannot compute $\Delta_{0,n}$ exactly, but we can estimate it by simulation. The obvious candidate for an estimator is

$$\hat{\Delta}_{0,n} = \int_0^1 \frac{(\hat{F}_{0,n}(u) - u)^2}{u(1-u)} du, \quad (5)$$

where $\hat{F}_{0,n}$ is the empirical distribution of N i.i.d. replicates of $W_{n,1}^*$, for a very large N . This estimator turns out to be A_N^2 itself. As $N \rightarrow \infty$, $\tilde{A}_{N,n}^2$ converges in distribution to a random variable A having the asymptotic AD distribution, so A_N^2 converges to $\Delta_{0,n} + \zeta$, where ζ is a random variable satisfying $0 \leq \zeta \leq A$, because $A_N^2 \leq \Delta_{0,n} + \tilde{A}_{N,n}^2$.

The theoretical distribution of A_N^2 under $\tilde{\mathcal{H}}_0$ (or, equivalently, of $\tilde{A}_{N,n}^2$ under \mathcal{H}_0) converges quickly with N . For $N \geq 10$, one has $P[A_N^2 > 2.5] \approx 0.05$ and $P[A_N^2 > 3.9] \approx 0.01$. Therefore, a value of $\hat{\Delta}_{0,n}$ larger than 3 or 4 is a statistically significant indication of approximation error.

Table 1 reports a few values of $\hat{\Delta}_{0,n}$ computed for the unit hypercube with the Euclidean norm ($p = 2$). These

simulations used $N = 10^5$ replications, except for larger values of n where we took $N = 10^4$. The approximation error is clearly *nonnegligible* for $n \leq 10^3$ in four dimensions or more, and even for $n = 10^4$ and $k \geq 6$. The same problem occurs with $p = 1$ and $p = \infty$, and for the other tests such as m -NP, and so on. The effect of the error is in fact *worse* for the m -NP tests than for the NP tests, and increases with m .

These results contrast with those of Ripley (1987, p. 26), who said “The approximation . . . is remarkably accurate for n as small as 25.” Even in two dimensions, we find a significant error for $n = 25$. However, Ripley was using a single value of $W_{n,1}^*$ for his tests, not a goodness-of-fit test for the distribution of N replicates as we do here, and he reported experiments only for two dimensions.

Most of the approximation error that we just measured is due to the boundary effect. The proof of Proposition 1 assumes grossly that the ball of radius $r = D_{n,i}$ centered at X_i is contained in the unit hypercube, at least for the smallest $D_{n,i}$ s. But this holds only for the points X_i that are at a distance at least r from each boundary. The fraction of the hypercube volume where this property holds is $(1 - 2r)^k$. For fixed r , it decreases to zero exponentially fast in k . For large k , most of the space is near the boundary. To keep $(1 - 2D_{n,i})^k$ close to one, n must increase exponentially fast as a function of k . For example, suppose that for a given k we compute the value of r such that $(1 - 2r)^k \approx 0.95$ (i.e., 95% of the points are at distance at least r from the boundary) and then the minimal n such that $P[D_n^* \leq r | \mathcal{H}_0] \approx 0.95$ (i.e., there is a 95% chance that the shortest distance is less than r). For $k = 2, 4, 8, 12, 16$ (for instance), one obtains $n \geq 110, 27,150, 1.16 \times 10^{10}, 2.25 \times 10^{16}, 1.17 \times 10^{23}$, respectively. The required n quickly becomes excessive.

Removing the boundary effect motivates taking the unit torus instead of the hypercube. We re-computed $\hat{\Delta}_{0,n}$ as defined previously, with $N = 10^5$, but using the distances in the torus. For $k \leq 12$ and $n \geq 1,000$, $\hat{\Delta}_{0,n}$ never exceeded the 0.01 significance level. We performed a similar empirical analysis for all the other tests that we tried, such as the m -NP test, the NP test with the spacings or power ratio transformations, and so on. Our findings can be succinctly summarized by the following (conservative) rule of thumb: For $k \leq 8$ and $n \geq 4m^2\sqrt{N}$, all these tests appear safe; i.e., they give p values that are reasonably close to the correct ones. The approximation error increases with k and m and decreases with n . Increasing N generally increases the effect of that error on the test outcome. The smaller the p value in a given test, the larger is the value of A_N^2 and the smaller are the chances that this is due only to the approximation error. It is therefore advisable to seek extremely small p values before rejecting a generator.

We also performed simulation experiments with the Bickel-Breiman statistic in the torus. We estimated the distribution of B_n under \mathcal{H}_0 for values of k ranging from 1 to 20 and n from 100 to 10,000. Our results with the L_2 -norm agree with those of Schilling (1983b), and our esti-

Table 2. CPU time (sec) to find the nearest pair in the hypercube, for $p = 2$.

k	$n = 10^3$	$n = 10^4$	$n = 10^5$	$n = 10^6$
2	0.01	0.05	1.3	17
4	0.01	0.11	2.3	31
8	0.03	0.47	6.3	
12	0.17	1.7	23	
16	0.62	13	120	

ated distribution for any $k \geq 15$ and $n \geq 10^3$ is close to his infinite-dimensional approximation (with no statistically significant difference for $N = 10^4$). However, computing B_n is very time consuming for such a large k . Practical parameter ranges could be $n \leq 10^5$ for $k = 2$, $n \leq 10^4$ for $k = 4$, and $n \leq 10^3$ for $k \leq 15$ and $p = \infty$. To apply the tests, we selected $(p, k) = (2, 2), (\infty, 2), (\infty, 15)$. For these parameter values, we estimated the distribution of B_{1000} by simulation with $N = 10^6$ for $k = 2$, and $N = 10^5$ for $k = 15$, fitted a least-squares cubic spline approximation to the data, and used these estimated distributions for our tests. We actually repeated this for some larger values of n , and the distribution did not change significantly, indicating that the distribution for $n = 1,000$ is already quite close to the asymptotic one.

Another important practical issue is the *discretization error* when computing the AD statistic. For floating-point numbers with a 53-bit precision, the difference between two numbers close to 1 and less than 2^{-53} apart (for example) will be considered as zero. And if one of the $U_{(j)}$ is zero in (2), then $A_N^2 = \infty$. In our implementation we thus (heuristically) replaced any $U_{(j)}$ or $1 - U_{(j)}$ that was 0 by 2^{-54} in the computation of A_N^2 . Under the null hypothesis this has a negligible effect, at least for the parameter values we consider.

3. COMPUTING TIMES

The naive way of computing $W_{n,1}^*, \dots, W_{n,m}^*$ for a given set of points is to compute the distances between all $n(n-1)/2$ pairs and keep the m shortest ones. We implemented a better algorithm, which is a modification of that outlined by Preparata and Shamos (1985). It is explained in a longer version of this paper, which is available from the authors. It works for any $p \geq 1$, for the hypercube and the torus, and is programmed in the Modula-2 language.

Tables 2 to 4 give some CPU timings (in seconds) to find the nearest pair of points (i.e., $m = 1$) with our implementation for different values of n and k . Each table entry is the average over five trials, and nonsignificant digits have been removed or replaced by zeroes. The blank entries are cases that we did not try because they seemed to take a lot of time. These and all other computations reported in this paper were performed on a SUN Ultra-1. The timings depend of course on the choice of machine and compiler. They are meant to give a crude indication of the speed of the algorithm as a function of the parameters.

Table 2 is for the Euclidean distance in the unit hyper-

Table 3. CPU time (sec) to find the nearest pair in the torus, for $p = 2$.

k	$n = 10^3$	$n = 10^4$	$n = 10^5$	$n = 10^6$
2	0.01	0.05	1.3	18
4	0.01	0.09	2.4	31
8	0.05	0.6	7.7	
12	0.45	6.0	54	
16	5.0	80	876	

cube, and Tables 3 and 4 are for the unit torus for $p = 2$ and $p = \infty$, respectively. We made similar experiments with $p = 1$ and also with $m = 10$ and $m = 32$. The results can be summarized as follows. The computing times increase faster than linearly in k and increase much faster in k for small p than for large p . In small dimensions (e.g., $k \leq 4$), the computing time is almost the same for the hypercube and torus and almost independent of p . But in larger dimensions, computing the nearest pair is more expensive in the torus than in the hypercube and much more expensive for small p than for large p . When increasing m from 1 to 32, the increase in CPU time goes from negligible in two and four dimensions to approximately 100% for $(p, k, n) = (\infty, 24, 10^4)$, 300% for $(p, k, n) = (2, 16, 10^4)$, and more for $p = 1$.

To compute B_n , one needs the nearest-neighbor distance from each point. Table 5 gives some timings for our implementations with $p = \infty$ in the unit torus. As can be seen by comparing these results to those of Table 4, computing B_n is much more time consuming than finding only the closest pairs.

4. EXPERIMENTAL RESULTS FOR RNG TESTING

In this section, we apply close-pair tests to some RNGs to investigate the relative effectiveness of different test variants and parameter choices for detecting deficiencies in RNGs, particularly in LCGs. We know in advance that LCGs should fail these tests because of their regular lattice structure. We want to see for which sample size an LCG with good lattice structure fails decisively, as a function of its period length, for the different test variants. This will give some indication about how safe are the linear-type generators with large periods and good lattice structure for simulating similar random variables.

Table 4. CPU time (sec) to find the nearest pair in the torus, for $p = \infty$.

k	$n = 10^3$	$n = 10^4$	$n = 10^5$	$n = 10^6$
2	0.01	0.05	1.3	18
4	0.01	0.10	2.4	31
8	0.03	0.32	5.5	
12	0.06	0.94	15	
16	0.11	2.2	31	
20	0.19	4.1	68	
24	0.32	8.0		

Table 5. CPU time (sec) to compute B_n in the torus, for $p = \infty$.

k	$n = 10^3$	$n = 10^4$	$n = 10^5$	$n = 10^6$
2	0.1	1	16	210
4	0.3	6	305	
8	0.8	29	1300	
12	1.5	72	2800	
16	2.3	130		

4.1. A Sample of Random Number Generators

Table 6 gives a short list of RNGs selected to illustrate our tests. Each generator outputs a real number between 0 and 1 at each step. The points X_i are nonoverlapping k -dimensional vectors of successive output values. G1 to G7 are well-known LCGs, based on a recurrence of the form $x_i = (ax_{i-1} + c) \bmod M$, with output $u_i = x_i/M$ at step i . They are discussed in many books and used in various software packages (see, e.g., Bratley et al. 1987, Fishman 1996, Law and Kelton 1991). G8 is a multiple recursive generator (MRG) of order 5, with modulus $M = 2^{31} - 1$ and multipliers $a_1 = 107374182$, $a_5 = 104480$, $a_2 = a_3 = a_4 = 0$, proposed by L'Écuyer et al. (1993). G9 is an *explicit* inversive generator of the form $x_i = (ai + b) \bmod M$, $z_i = x_i^{-1} \bmod M = x_i^{M-2} \bmod M$, $u_i = z_i/M$ (Eichenauer-Herrmann 1992, Hellekalek 1995). G10 is the GFSR generator given in Ripley (1990, appendix). G11, G12, and G13 are the combined LCG of L'Écuyer (1988), the combined MRG given in L'Écuyer (1996a, Figure 1), and the combined Tausworthe generator in L'Écuyer (1996b, Figure 1), respectively. G14 to G16 are double precision versions of G11 to G13. Each call to G14 makes two calls to G11 to get two uniforms w_1 and w_2 and outputs $u = (w_1 + 2^{-20}w_2) \bmod 1$. The motivation for this is to obtain more bits of resolution in the output. G15 and G16 work the same way.

Table 6. List of selected generators.

G1.	LCG with $M = 2^{31} - 1$ and $a = 742938285$.
G2.	LCG with $M = 2^{31} - 1$ and $a = 630360016$.
G3.	LCG with $M = 2^{31} - 1$ and $a = 16807$.
G4.	LCG with $M = 2^{32}$, $a = 69069$, and $c = 1$.
G5.	MCG with $M = 2^{31}$ and $a = 65539$.
G6.	LCG with $M = 2^{31}$ and $a = 452807053$.
G7.	LCG with $M = 2^{31}$, $a = 1103515245$, $c = 12345$.
G8.	MRG of order 5, from L'Écuyer, Blouin, and Couture (1993).
G9.	Explicit inversive generator with $M = 2^{31}$ and $a = b = 1$.
G10.	GFSR-521 in the Appendix of Ripley (1990).
G11.	Combined LCG in Fig. 3 of L'Écuyer (1988).
G12.	Combined MRG in Fig. 1 of L'Écuyer (1996a).
G13.	Combined Tausworthe generator in Fig. 1 of L'Écuyer (1996b).
G14.	A double precision version of G11.
G15.	A double precision version of G12.
G16.	A double precision version of G13.

Table 7. Parameters for some NP and m -NP test.

Test	p	k	m	n	N	Nn	Nnk
T1	∞	2	32	2^{14}	1	2^{14}	2^{15}
T2	∞	2	16	2^{12}	4	2^{14}	2^{15}
T3	∞	2	8	2^{10}	16	2^{14}	2^{15}
T4	∞	2	32	2^{17}	1	2^{17}	2^{18}
T5	∞	2	8	2^{13}	16	2^{17}	2^{18}
T6	∞	4	32	2^{17}	1	2^{17}	2^{19}
T7	∞	4	8	2^{13}	16	2^{17}	2^{19}
T8	∞	8	32	2^{17}	1	2^{17}	2^{20}
T9	∞	2	32	2^{20}	32	2^{25}	2^{26}
T10	∞	4	32	2^{20}	32	2^{25}	2^{27}
T11	∞	8	32	2^{20}	32	2^{25}	2^{28}

4.2. Test Results for the Selected Generators

Table 7 gives a list of test parameters for the NP and m -NP tests in the unit torus. Table 8 gives another set for the Bickel-Breiman test. In both tables, Nn and Nnk are the total number of points generated and the total number of calls to the generator for the test, respectively. These parameter values have been chosen to illustrate things that typically happen when these tests are applied to LCGs. For each selection of parameters, we applied (among others) the following four tests: the NP test, the NP test combined with the spacings transformation (NP-S), the NP test combined with the power ratio transformation (NP-PR), and the m -NP test. We now summarize the results.

None of the generators G8 to G16 had difficulty with these tests. We observed a few p values just below 0.01 now and then, as normally expected. On the other hand, all the LCGs G1 to G7 failed several tests. In particular, with T9 to T11, which are two-level and throw approximately one million points in the hypercube 32 times, in different dimensions, G1 to G7 failed all the tests at significance level 10^{-15} . For the other selected parameter sets, for the LCGs, the m -NP test generally appears more efficient than the other three types of tests, and gains power when m is increased (at least for $m \leq 32$). Also, for a fixed value of Nn , it is usually better to take $N = 1$. For example, we obtained several p values less than 10^{-15} with T1 (for the m -NP test), a few p values below 0.01 for T2, and no p value less than 0.01 for T3. With T4, T6, and T8, the 32-NP test had several p values less than 10^{-15} , but the

Table 8. Parameter values for the Bickel-Breiman test.

Test	p	k	n	N	Nn	Nnk
B1	2	2	2^{18}	1	2^{18}	2^{19}
B2	2	2	2^{18}	16	2^{22}	2^{23}
B3	2	2	2^{20}	16	2^{24}	2^{25}
B4	∞	2	2^{16}	1	2^{16}	2^{17}
B5	∞	2	2^{14}	16	2^{18}	2^{19}
B6	∞	2	2^{18}	1	2^{18}	2^{19}
B7	∞	2	2^{18}	16	2^{22}	2^{23}
B8	∞	2	2^{20}	1	2^{20}	2^{21}
B9	∞	2	2^{20}	16	2^{24}	2^{25}
B10	∞	15	2^{16}	1	2^{16}	$2^{19.9}$
B11	∞	15	2^{18}	1	2^{18}	$2^{21.9}$

Table 9. The p -values for G3.

Test	NP	NP-S	NP-PR	m -NP
T1				ϵ
T2				3.1E-3
T3				
T4				ϵ
T5		9.3E-7	4.1E-7	ϵ
T6	1.7E-3	3.4E-3	1.7E-3	ϵ
T7				1.2E-3
T8				ϵ
T9	ϵ	ϵ	ϵ	ϵ
T10	ϵ	ϵ	ϵ	ϵ
T11	ϵ	ϵ	ϵ	ϵ

other three types of tests had very few below 10^{-3} . Of course, we should not expect the NP tests with $N = 1$ to perform well unless n is quite large because they look only at $W_{n,1}^*$. With T5 and T7, NP-S and NP-PR generally did better than m -NP with several very small p values, some below 10^{-15} . But there were a few exceptions. For example, for G3 and T5, the p value of the 8-NP test was less than 10^{-15} , whereas those of the NP-S and NP-PR tests were around 10^{-6} . For a specific illustration, Table 9 reports the “suspect” p values (those smaller than 0.01) for the generator G3. The entries with nonsuspect p values are left blank. Values smaller than 10^{-15} are denoted ϵ .

We ran similar tests with $p = 1$ and $p = 2$, and for other parameter sets not shown here. The tests with $p = \infty$ were at least as sensitive in our experiments as those with smaller values of p . For large k , the tests with $p = \infty$ also run faster. For the LCGs in our list, for a fixed n , the tests with $k = 2$ were as sensitive and less costly to run than the higher dimensional ones. For a given number of calls to the generator, they were the most efficient. However, one should not conclude that $k = 2$ is sufficient in general. Taking $k > 2$ is appropriate and indicated if there is reason to believe that a generator may behave badly for some large k or if defective behavior for a specific $k > 2$ would badly affect the application at hand.

Figures 1 and 2 illustrate the typical behavior of an LCG when it starts to fail the m -NP test. They are for G3 with

Figure 1. Part of the trajectory of $Y_n(t)$ for G3 with $n = 2^{14}$, $k = 2$, $p = \infty$.

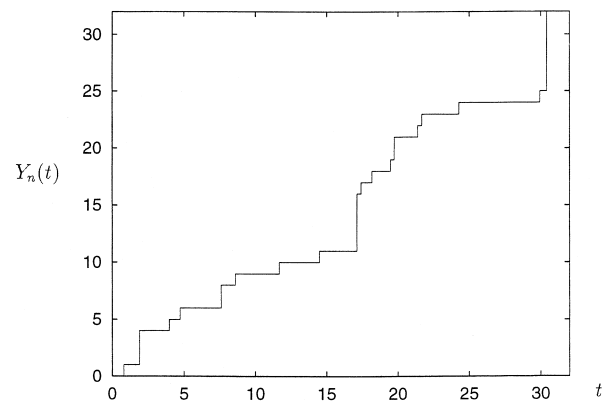
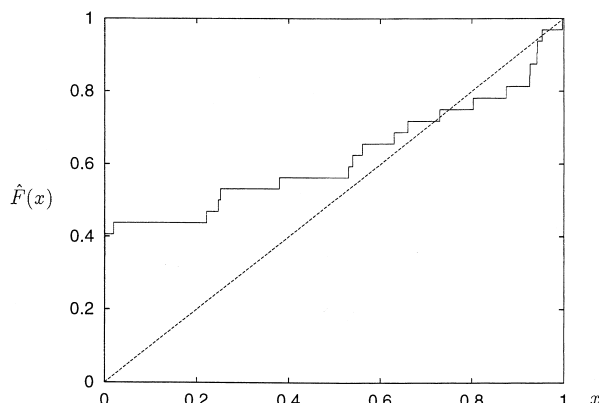


Figure 2. Empirical distribution of $W_{n,1}^*, \dots, W_{n,32}^*$ for G3 with $n = 2^{14}$, $k = 2$, $p = \infty$.



T1. Figure 1 shows the first 32 jumps of Y_n , Figure 2 shows the distribution of the corresponding $W_{n,i}^*$. We count a jump each time $Y_n(t)$ increases by one unit. Some of the jumps occur at the same t (for instance, the second, third, and fourth jumps, as well as the seventh and eighth, and so on). For larger n , this behavior is accentuated. For $n = 2^{17}$ (T4), for example, the first 32 jumps occur at only five different values of t . Because of these simultaneous jumps, several $W_{n,i}^*$ are zero, so their empirical distribution has a large jump at the origin (Figure 2). The AD statistic then becomes huge and would be infinite if the $W_{n,i}^*$ that are zero were not replaced by 2^{-54} .

What causes this kind of behavior? Because of the regular lattice structure of the LCG, the distances between the close points are distributed over a (small) finite set of values. When n increases, this set remains fixed and the smallest values in the set are picked by a larger number of pairs (the number of pairs is $O(n^2)$, so the number of pairs at a given distance should also increase as $O(n^2)$ in probability). The other generators also have their output values distributed over a finite set (e.g., are all multiples of $1/M$ for G9, etc.). But this discretization is too fine to affect the results of the tests performed here.

Figure 3 shows the empirical distribution of the $W_{n,1}^*$ s

Figure 3. Empirical distribution of the $W_{n,1}^*$ s for G4 with T7.

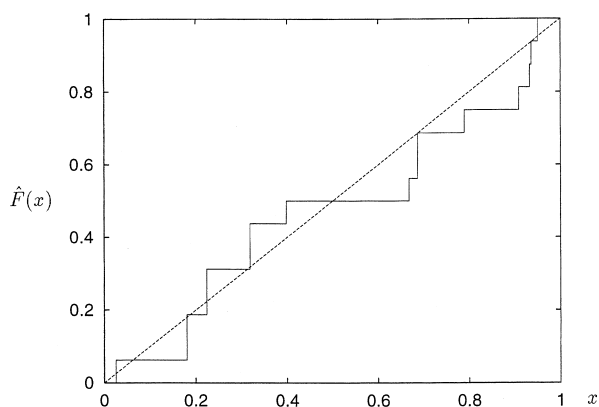
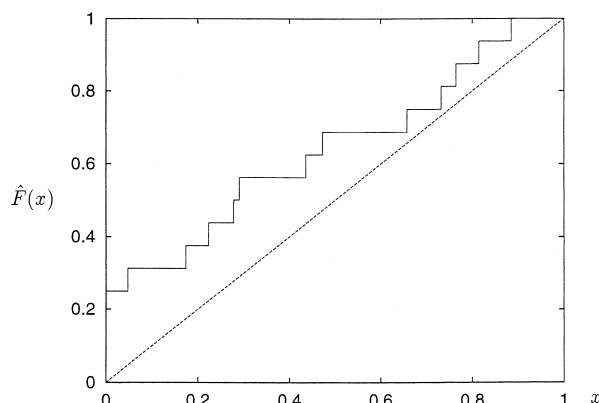


Figure 4. Empirical distribution of the $W_{n,1}^*$ s after IS for G4 with T7.



for G4 with T7. There are five cases where two values of $W_{n,1}^*$ are equal or nearly equal, but the AD statistic does not detect this. Figure 4 gives the empirical distribution of the uniforms after applying the spacings transformation to these $W_{n,1}^*$ s. There is now a large jump at the origin, easily detected by the AD statistic. A similar figure for the distribution after the power ratio transformation would show a large jump at the other end, near 1.

Table 10 gives the suspect p values for the Bickel-Breiman tests of Table 8. The tests with $N = 16$ are two-level tests, where the second level computes an AD statistic from the 16 p values of the first level, as in Equation (2). All the LCGs G1–G7 fail some of the tests, but for total sample sizes significantly larger than for the m -NP tests. For a given Nn , a larger n usually gives more power, but at a larger CPU cost. G8 to G16 passed these tests. The Bickel-Breiman statistic is sensitive to general clustering between the points, but is not very sensitive to clustering or equalities between the values of the first few nearest-neighbor distances $D_{n,i}$, in contrast to the m -NP tests. Certain generators could produce a different type of clustering than the LCGs and perhaps the Bickel-Breiman tests could then be more useful.

Table 10. The p -values for the Bickel-Breiman tests.

Test	G1	G2	G3	G4	G5	G6	G7
B1					ϵ	2.2E-3	
B2			1.0E-4		ϵ	ϵ	
B3	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ	ϵ
B4							
B5							
B6			ϵ		ϵ		
B7			ϵ	ϵ	ϵ	3.0E-4	
B8	5.7E-3		ϵ	ϵ	ϵ	ϵ	
B9	8.4E-12	1.4E-4	ϵ	ϵ	ϵ	ϵ	ϵ
B10					ϵ	ϵ	
B11	4.3E-5	5.5E-6	ϵ		ϵ	ϵ	ϵ

Table 11. The p values for the NP test with $N = 1$, for several LCGs.

e	$\nu = 0$	$\nu = 1$	$\nu = 2$	$\nu = 3$
14			5.5E-07	ϵ
16		7.6E-03	3.3E-09	ϵ
18			7.5E-06	ϵ
20		1.8E-03	1.0E-11	ϵ
22		9.5E-03	8.1E-09	ϵ
24		7.0E-04	2.3E-13	ϵ
26		2.6E-03	4.6E-11	ϵ
28			1.0E-07	ϵ
30			1.8E-07	ϵ
32			6.5E-08	ϵ
34		3.0E-03	7.6E-11	ϵ
36		7.2E-03	2.8E-09	ϵ
38		3.7E-03	1.8E-10	ϵ
40		2.5E-03	4.1E-11	ϵ

4.3. Other Experiments and Examples with LCGs and MRGs

For each even integer e from 14 to 40, we took an LCG with modulus M equal to the largest prime less than 2^e , with period length $M - 1$, and with excellent lattice structure in two dimensions. The selected LCGs are those with the best value of M_8 in L'Écuyer (1999, Table 2). For each e , we applied the tests in $k = 2$ dimensions with $p = \infty$ and $n = 2^{\nu+e/2} \approx 2^\nu \sqrt{M}$, for $\nu = -4, \dots, 4$. Table 11 gives the p values for the NP test with $N = 1$. The columns not shown in the table are blank for $\nu < 0$ and filled with ϵ for $\nu > 3$. The upcoming of the small p values when ν is increased is remarkably systematic for the different values of e . For this and other similar tables, define ν^* as the smallest value of ν for which most of the column entries are ϵ . As a crude rule, the test rejects almost certainly any LCG in a very decisive way at sample size $n \approx 2^{\nu^*} \sqrt{M}$. For the NP test with $N = 1$, we have $\nu^* = 3$.

We made similar tables for other close-pair tests and observed the same systematic behavior, except that ν^* took different values depending on the test. We also observed that the smallest value of ν where most of the column entries are less than 0.01 is almost always $\nu^* - 1$. Table 12 gives the value of ν^* that we obtained for certain tests, with $p = \infty$ and $k = 2$. To reject decisively, the NP test with $N = 1$ (which is the test discussed by Ripley 1987) needs $n \approx 8\sqrt{M}$, whereas the 32-NP test with $N = 1$ needs $n \approx \sqrt{M}$. With $N = 32$, the NP-S and NP-PR tests need a sample size approximately eight times smaller than the NP

Table 12. Value of ν^* for different tests applied to LCGs ($p = \infty, k = 2$).

Test	N	ν^*
NP	1	3
32-NP	1	0
NP	32	1
NP-S	32	-2
NP-PR	32	-2
16-NP	16	-1
32-NP	32	-1

test. So, the spacings and power ratio transformations are quite effective.

We now give two examples of long-period generators that pass the tests in two dimensions but fail badly in some higher dimension. Consider the MRG defined by $x_n = (3x_{n-1} - 7x_{n-2}) \bmod M$ and $u_n = x_n/M$, where $M = 2^{31} - 1$. This generator has period length $M^2 - 1 \approx 2^{62}$. In two dimensions, all the pairs $(i/M, j/M), i, j \in \{0, \dots, M - 1\}$, except $(0, 0)$, appear exactly once as a value of (u_n, u_{n+1}) over the generator's period. But in three dimensions, all the triples (u_n, u_{n+1}, u_{n+2}) lie on 10 planes. We applied the two-dimensional tests with parameters T9 to this generator, and it passed nicely. But for T10, the p values were all ϵ . In three dimensions, for $N = 1$, the p value of the 16-NP test is already 2.8×10^{-12} with $n = 2^{10}$ and is ϵ for $n = 2^{12}$.

As another example, consider $x_n = (2x_{n-1} - 2x_{n-5}) \bmod M$ and $u_n = x_n/M$, where $M = 2^{31} - 22641$. Here, the period length is $M^5 - 1 \approx 2^{155}$, and the structure is good up to five dimensions but awful in six or more dimensions: The points all lie in at most four hyperplanes. This generator easily passed all the close-pair tests that we tried in two to five dimensions, but it failed badly in six dimensions or more.

The two previous examples have been constructed on purpose to have a bad structure in three and six dimensions, respectively. But generators with similar bad properties have been proposed and used in the past. For example, G5 in our list (RANDU) has a much worse structure in three dimensions than in two dimensions (Knuth 1981). Also, the add-with-carry, subtract-with-borrow, and additive lagged-Fibonacci generators behave very similarly to the MRG of order 5 in our last example (L'Écuyer 1994, 1997).

5. CONCLUSION

We have examined and compared different variants of close-pair tests to detect the regularities of linear congruential generators. For $N = 1$, the m -NP test appears the most powerful, and it gets better as n and m are increased. The NP-S and NP-PR also do well for large enough N . We recommend keeping $n \geq 4m^2 \sqrt{N}$ for $k \leq 8$ to avoid damaging errors of approximation by the asymptotics. For a fixed value of $Nn, N = 1$ seems best for the m -NP test with large enough m . But the computing time increases faster than linearly with n . The main bottleneck for using a large n is usually the memory size. For example, for $n = 10^7$ and $k = 4$, approximately 320 megabytes of memory are needed just to store the points. Therefore, to implement a very stringent test, one should first raise n to the highest reasonable value (depending on the available memory on the computer at hand), then increase N .

From the results of Table 12, we can anticipate for example that any good LCG with period 2^{60} will fail decisively a 32-NP test with $N = 1$ and $n \approx 2^{30}$, or with $N = 32$ and $n \approx 2^{28}$. Such sample sizes are not out of reach for

current computers with enough memory. And if the lattice structure is not so good, clear failure may occur for much smaller n . But for a good LCG with period length (say) over 2^{100} , the sample size required for testing is prohibitively large with current technology.

ACKNOWLEDGMENT

This work has been supported by the National Science and Engineering Research Council of Canada grants no. ODGP0110050 and SMF0169893, and FCAR-Québec grant no. 93ER1654. The paper has been greatly improved thanks to the suggestions of the Area Editor Barry Nelson, the Associate Editor, and two anonymous referees. Part of this research was made while the first author was visiting the pLab group, headed by Peter Hellekalek, at the University of Salzburg, Austria. Special thanks to Peter, Hannes Leeb, and Stefan Wegenkittl from the group for their support and their comments on the paper.

REFERENCES

- Bickel, P. J., L. Breiman. 1983. Sums of functions of nearest neighbor distances, moment bounds, limit theorems and a goodness of fit test. *Ann. Probab.* **11**(1), 185–214.
- Bratley, P., B. L. Fox, L. E. Schrage. 1987. *A Guide to Simulation*. Second ed. Springer-Verlag, New York.
- Cressie, N. 1993. *Statistics for Spatial Data*. Wiley, New York.
- Durbin, J. 1973. *Distribution Theory for Tests Based on the Sample Distribution Function*. SIAM CBMS-NSF Regional Conference Series in Applied Mathematics, Philadelphia, PA.
- Eichenauer-Herrmann, J. 1992. Inversive congruential pseudorandom numbers: A tutorial. *Internat. Statist. Reviews* **60** 167–176.
- Fishman, G. S. 1996. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer Series in Operations Research, Springer-Verlag, New York.
- Hellekalek, P. 1995. Inversive pseudorandom number generators: Concepts, results, and links. In *Proceedings of the 1995 Winter Simulation Conference*. C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman (eds.) IEEE Press, 255–262.
- Knuth, D. E. 1981. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Second ed. Addison-Wesley, Reading, MA.
- Law, A. M., W. D. Kelton. 1991. *Simulation Modeling and Analysis*. Second ed. McGraw-Hill, New York.
- L'Écuyer, P. 1988. Efficient and portable combined random number generators. *Comm. ACM* **31**(6) 742–749, 774. (See also the correspondence in the same journal, 32, 8 (1989) 1019–1024.)
- . 1992. Testing random number generators. In *Proc. 1992 Winter Simulation Conference* IEEE Press, 305–313.
- . 1994. Uniform random number generation. *Ann. Oper. Res.* **53** 77–120.
- . 1996a. Combined multiple recursive random number generators. *Oper. Res.* **44**(5) 816–822.
- . 1996b. Maximally equidistributed combined Tausworthe generators. *Math. Computation* **65**(213) 203–213.
- . 1997. Bad lattice structures for vectors of non-successive values produced by some linear recurrences. *INFORMS J. Comput.* **9**(1), 57–60.
- . 1999. A table of linear congruential generators of different sizes and good lattice structure. *Math. Computation*. **68** (225) 249–260.
- , F. Blouin, R. Couture. 1993. A search for good multiple recursive random number generators. *ACM Trans. Modeling and Computer Simulation* **3**(2) 87–98.
- Marsaglia, G. 1985. A current view of random number generators. In *Computer Science and Statistics, Sixteenth Symposium on the Interface*, Elsevier Science Publishers, North-Holland, Amsterdam, 3–10.
- Niederreiter, H. 1992. *Random Number Generation and Quasi-Monte Carlo Methods*. Volume 63 of *SIAM CBMS-NSF Regional Conference Series in Applied Mathematics*. Philadelphia, PA.
- Preparata, F. P., M. I. Shamos. 1985. *Computational Geometry: An Introduction*. Texts and Monographs in Computer Science, Springer-Verlag, New York.
- Pyke, R. 1959. The supremum and infimum of the Poisson process. *Ann. Math. Statist.* **30** 568–576.
- . 1965. Spacings. *J. Royal Statist. Soc. Series B* **27** 395–449.
- Ripley, B. D. 1977. Modelling spatial patterns. *J. Royal Statist. Soc. Series B* **39** 172–212.
- . 1987. *Stochastic Simulation*. Wiley, New York.
- . 1988. *Statistical Inference for Spatial Processes*. Cambridge University Press, New York.
- . 1990. Thoughts on pseudorandom number generators. *J. Comput. Appl. Math.* **31** 153–163.
- , B. W. Silverman. 1978. Quick tests for spatial interaction. *Biometrika* **65**(3) 641–642.
- Saunders, R., G. M. Funk. 1977. Poisson limits for a clustering model of Strauss. *J. Appl. Probab.* **14** 776–784.
- Schilling, M. F. 1983a. Goodness of fit testing in \mathbb{R}^m based on the weighted empirical distribution of certain nearest neighbor statistics. *Ann. Statist.* **11** 1–12.
- . 1983b. An infinite-dimensional approximation for nearest neighbor goodness of fit tests. *Ann. Statist.* **11** 13–24.
- Silverman, B., T. Brown. 1978. Short distances, flat triangles and Poisson limits. *J. Appl. Probab.* **15** 815–825.
- Stephens, M. S. 1986. Tests for the uniform distribution. In *Goodness-of-Fit Techniques*. R. B. D'Agostino and M. S. Stephens (eds.) Marcel Dekker, New York and Basel, 331–366.