SIAM J. SCI. COMPUT.
Vol. 24, No. 2, pp. 652–668
© 2002 Society for Industrial and Applied Mathematics

SPARSE SERIAL TESTS OF UNIFORMITY FOR RANDOM NUMBER GENERATORS*

PIERRE L'ECUYER†, RICHARD SIMARD†, AND STEFAN WEGENKITTL‡

Abstract. Different versions of the serial test for testing the uniformity and independence of vectors of successive values produced by a (pseudo)random number generator are studied. These tests partition the t-dimensional unit hypercube into k cubic cells of equal volume, generate n points (vectors) in this hypercube, count how many points fall in each cell, and compute a test statistic defined as the sum of values of some univariate function f applied to these k individual counters. Both overlapping and nonoverlapping vectors are considered. For different families of generators, such as linear congruential, Tausworthe, nonlinear inversive, etc., different ways of choosing these functions and of choosing k are compared, and formulas are obtained for the (estimated) sample size required to reject the null hypothesis of independent uniform random variables, as a function of the period length of the generator. For the classes of alternatives that correspond to linear generators, the most efficient tests turn out to have $k \gg n$ (in contrast to what is usually done or recommended in simulation books) and to use overlapping vectors.

Key words. random number generation, goodness-of-fit, serial test, collision test, m-tuple test, multinomial distribution, OPSO

AMS subject classifications. 60F05, 60F17, 62L20, 93E23, 93E25, 93E30

PII. S1064827598349033

1. Introduction.

The aim of this paper is to examine certain types of *serial tests* for testing the uniformity and independence of the output sequence of general-purpose uniform random number generators (RNGs) such as those found in software libraries. These RNGs are supposed to produce "imitations" of mutually independent random variables uniformly distributed over the interval $[0, 1)$ (i.i.d. $U(0, 1)$). Testing an RNG whose output sequence is $U_0, U_1, U_2, \ldots$ amounts to testing the null hypothesis $\mathcal{H}_0$, which states that the U_i are independent identically distributed (i.i.d.) $U(0, 1)$.

To approximate this multidimensional uniformity, good RNGs are usually designed (theoretically) so that the multiset Ψ_t of all vectors $(u_0, \ldots, u_{t-1})$ of their first t successive output values, from all possible initial seeds, covers the t-dimensional unit hypercube $[0, 1)^t$ very evenly, at least for t up to some t_0, where t_0 is chosen somewhere between 5 and 50, for example. When the initial seed is chosen randomly, this Ψ_t can be viewed in some sense as the sample space from which points are chosen at random to approximate the uniform distribution over $[0, 1)^t$. For more background on the construction of RNGs, see, for example, [13, 17, 21, 35].

For large t, the structure of Ψ_t typically is hard to analyze theoretically. Moreover, even for a small t, one would often generate several successive t-dimensional vectors

*Received by the editors December 4, 1998; accepted for publication (in revised form) June 21, 2001; published electronically October 16, 2002. This work was supported by National Science and Engineering Research Council of Canada grants ODGP0110050 and SMF0169893, by FCAR-Québec grant 93ER1654, and by Austrian Science Fund FWF project P11143-MAT. Most of the work was performed while the first author was visiting Salzburg University and North Carolina State University, 1997–98 (thanks to Peter Hellekalek and James R. Wilson).

http://www.siam.org/journals/sisc/24-2/34903.html

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

‡Institute of Mathematics, University of Salzburg, Hellbrunnerstrasse 34, A-5020 Salzburg, Austria (stefan.wegenkittl@u24.at).

of the form $(u_{ti}, \ldots, u_{ti+t-1})$, $i \geq 0$. Empirical statistical testing then comes into play because the dependence structure of these vectors is hard to analyze theoretically. An excessive regularity of Ψ_t implies that statistical tests would fail when their sample sizes approach the period length of the generator. But how close to the period length can one get before trouble begins?

Several goodness-of-fit tests for $\mathcal{H}_0$ have been proposed and studied in the past (see, e.g., [13, 9, 26, 41] and references therein). Statistical tests can never *certify for good* an RNG. Different types of tests detect different types of deficiencies, and the more diversified the available battery of tests is, the better.

A simple and widely used test for RNGs is the *serial test* [1, 6, 8, 13], which operates as follows. Partition the interval $[0, 1)$ into d equal segments. This determines a partition of $[0, 1)^t$ into $k = d^t$ cubic cells of equal size. Generate nt random numbers $U_0, \ldots, U_{nt-1}$, construct the points $V_{ti} = (U_{ti}, \ldots, U_{ti+t-1})$, $i = 0, \ldots, n-1$, and let X_j be the number of these points falling into cell j for $j = 0, \ldots, k-1$. Under $\mathcal{H}_0$, the vector $(X_0, \ldots, X_{k-1})$ has the multinomial distribution with parameters $(n, 1/k, \ldots, 1/k)$. The usual version of the test, as described, for example, in [6, 13, 14], is based on Pearson's chi-square statistic

$$X^2 = \sum_{j=0}^{k-1} \frac{(X_j - \lambda)^2}{\lambda} = -n + \frac{1}{\lambda} \sum_{j=0}^{k-1} X_j^2, \tag{1.1}$$

where $\lambda = n/k$ is the average number of points per cell and the distribution of X^2 under $\mathcal{H}_0$ is approximated by the chi-square distribution with $k-1$ degrees of freedom when, say, $\lambda \geq 5$.

In this paper, we consider test statistics of the general form

$$Y = \sum_{j=0}^{k-1} f_{n,k}(X_j), \tag{1.2}$$

where $f_{n,k}$ is a real-valued function which may depend on n and k. We are interested, for instance, in the *power divergence* statistic

$$D_\delta = \sum_{j=0}^{k-1} \frac{2}{\delta(1+\delta)} X_j \left[(X_j/\lambda)^\delta - 1 \right], \tag{1.3}$$

where $\delta > -1$ is a real-valued parameter (by $\delta = 0$, we understand the limit when $\delta \to 0$). One could also consider $\delta \to -1$ and $\delta < -1$, but this seems unnecessary in the context of this paper. Note that $D_1 = X^2$. The power divergence statistic is studied in [39] and references therein. A more general class is the *φ-divergence* family, where $f_{n,k}(X_j) = \lambda \varphi(X_j/\lambda)$ (see, e.g., [4, 34]). Other forms of $f_{n,k}$ that we consider are $f_{n,k}(x) = I[x \geq b]$ (where I denotes the indicator function), $f_{n,k}(x) = I[x = 0]$, and $f_{n,k}(x) = \max(0, x-1)$, for which the corresponding Y is the number of cells with at least b points, the number of empty cells, and the number of collisions, respectively.

We are interested in not only the *dense* case, where $\lambda > 1$, but also the *sparse* case, where λ is small, sometimes much smaller than 1. We also consider (circular) *overlapping* versions of these statistics, where $U_i = U_{i-n}$ for $i \geq n$ and V_{ti} is replaced by V_i.

In a slightly modified setup, the constant n is replaced by a Poisson random variable η with mean n. Then, $(X_0, \ldots, X_{k-1})$ is a vector of i.i.d. Poisson random

variables with mean λ instead of a multinomial vector, and the distribution of Y becomes easier to analyze because of this i.i.d. property. For large k and n, however, the difference between the two setups is practically negligible, so for simplicity and convenience our experiments deal only with $\eta = n$.

A *first-order* test observes the value of Y, say y, and rejects $\mathcal{H}_0$ if the p-value

$$p = P[Y > y \mid \mathcal{H}_0]$$

is much too close to either 0 or 1. The function f usually is chosen so that p too close to 0 means that the points tend to concentrate in certain cells and avoid the others, whereas p close to 1 means that they are distributed in the cells with excessive uniformity. So p can be viewed as a measure of uniformity and is approximately a $U(0,1)$ random variable under $\mathcal{H}_0$ if the distribution of Y is approximately continuous.

A *second-order* (or two-level) test would obtain N "independent" copies of Y, say $Y_1, \ldots, Y_N$, compute $F(Y_1), \ldots, F(Y_N)$, where F is the theoretical distribution of Y under $\mathcal{H}_0$, and compare their empirical distribution to the uniform. Such a two-level procedure is widely applied when testing RNGs (see [6, 13, 16, 29, 30]). Its main supporting arguments are that it tests the RNG sequence at not only the global level but also a local level (i.e., there could be bad behavior over short subsequences which "cancels out" over larger subsequences) and that it permits one to apply certain tests with a larger total sample size (for example, the memory size of the computer limits the values of n and/or k in the serial test, but the total sample size can exceed n by taking $N > 1$). Our extensive empirical investigations indicate that for a fixed total sample size Nn, when testing RNGs, a test with $N = 1$ typically is more efficient than the corresponding test with $N > 1$. This means that for typical RNGs, the type of structure found in one (reasonably long) subsequence usually is found in (practically) all subsequences of the same length. In other words, when an RNG begun from a given seed fails a certain test spectacularly, it usually fails that test for *most* admissible seeds.

The common way of applying serial tests to RNGs is to select a few specific generators and some arbitrarily chosen test parameters, run the tests, and check whether or not $\mathcal{H}_0$ is rejected. Our aim in this paper is to examine in a more systematic way the interaction between the serial tests and certain families of RNGs. From each family we take an RNG with period length near 2^e, chosen on the basis of the usual theoretical criteria, for integers e ranging from 10 to 40 approximately. We then examine, for different ways of choosing k and constructing the points V_i, how the p-value of the test evolves as a function of the sample size n. The typical behavior is that p takes "reasonable" values for a while, say, for n up to some threshold n_0, then converges to 0 or 1 exponentially fast with n. Our main interest is in examining the relationship between n_0 and e. We adjust (crudely) a regression model of the form $\log_2 n_0 = \gamma e + \nu + \epsilon$, where γ and ν are two constants and ϵ is a small noise. The result gives us an idea of the size (or period length) of RNG (within a given family) required to be safe with respect to these serial tests for the sample sizes that are practically feasible on current computers. It turns out that for popular families of RNGs such as linear congruential, multiple recursive, and shift-register, the most sensitive tests choose k proportional to 2^e and yield $\gamma = 1/2$ and $1 \le \nu \le 5$, which means that n_0 is a few times the square root of the RNG's period length.

The results depend, of course, on the choice of f in (1.2) and on how d and t are chosen. For example, for linear congruential generators (LCGs) selected on the basis of the spectral test [6, 13, 24], the serial test is most sensitive when $k \approx 2^e$, in

which case $n_0 = O(\sqrt{k})$. These "most efficient" tests are very sparse ($\lambda \ll 1$). Such large values of k yield tests more sensitive than the usual ones (for which $k \ll 2^e$ and $\lambda \geq 5$ or so) because the excessive regularity of LCGs really shows up at that level of partitioning. For $k \gg 2^e$, the partition eventually becomes so fine that each cell contains either 0 or 1 point, and the test loses all its sensitivity.

For fixed n, the nonoverlapping test typically is slightly more efficient than the overlapping one because it relies on a larger amount of independent information. However, the difference typically is almost negligible (see section 5.3) and the nonoverlapping test requires t times more random numbers. If we fix the total number of U_i's that are used so the nonoverlapping test is based on n points (whereas the overlapping one is based on nt points), for example, then the overlapping test typically is more efficient. It also is more costly to compute and its distribution is generally more complicated. If we compare the two tests for a fixed computing budget, we see that the overlapping one has an advantage when t is large and when the time to generate the random numbers is an important fraction of the total CPU time for applying the test.

In section 2 we collect some results on the asymptotic distribution of Y for the *dense case* where k is fixed and $n \to \infty$, the *sparse case* where both $k \to \infty$ and $n \to \infty$ so that $n/k \to \delta < \infty$, and the *very sparse case* where $n/k \to 0$. In section 3 we do the same for the overlapping setup. In section 4 we briefly discuss the efficiency of these statistics for certain classes of alternatives. Systematic experiments with these tests and certain families of RNGs are reported in section 5. In section 6, we apply the tests to a short list of RNGs proposed in the literature or widely used and available in software libraries. Most of these generators fail miserably. However, several recently proposed RNGs are robust enough to pass all these tests, at least for practically feasible sample sizes.

2. Power divergence test statistics for nonoverlapping vectors. We briefly discuss some choices of $f_{n,k}$ in (1.2) which correspond to previously introduced tests. We then provide formulas for the exact mean and variance as well as limit theorems for the dense and sparse cases.

2.1. Choices of $f_{n,k}$. Some choices of $f_{n,k}$ are given in Table 2.1. In each case, Y is a measure of clustering: it tends to increase when the points are *less* evenly distributed between the cells. The well-known Pearson and loglikelihood statistics, X^2 and G^2, are both special cases of the power divergence, with $\delta = 1$ and $\delta \to 0$, respectively [39]. H is related to G^2 via the relation $H = \log_2(k) - G^2/(2n \ln 2)$. The statistics N_b, W_b, and C count, respectively, the number of cells that contain exactly b points (for $b \geq 0$), the number of cells that contain at least b points (for $b \geq 1$), and the number of collisions (i.e., the number of times a point falls in a cell that already has a point in it). They are related by $N_0 = k - W_1 = k - n + C$, $W_b = N_b + \cdots + N_n$, and $C = W_2 + \cdots + W_n$.

2.2. Mean and variance. Before looking at the distribution of Y, we give expressions for computing its exact mean and variance under $\mathcal{H}_0$.

If the number of points is fixed at n, $(X_0, \ldots, X_{k-1})$ is multinomial. Denoting $\mu = E[f_{n,k}(X_j)]$, one obtains, after some algebraic manipulations,

TABLE 2.1
Some choices of $f_{n,k}$ and the corresponding statistics.

Y	$f_{n,k}(x)$	name
D_δ	$2x[(x/\lambda)^\delta - 1]/(\delta(1+\delta))$	power divergence
X^2	$(x-\lambda)^2/\lambda$	Pearson
G^2	$2x\ln(x/\lambda)$	loglikelihood
$-H$	$(x/n)\log_2(x/n)$	negative entropy
N_b	$I[x=b]$	number of cells with exactly b points
W_b	$I[x\ge b]$	number of cells with at least b points
N_0	$I[x=0]$	number of empty cells
C	$(x-1)\,I[x>1]$	number of collisions

$$E[Y] = k\mu \;=\; \sum_{x=0}^{n} \binom{n}{x} \frac{(k-1)^{n-x}}{k^{n-1}} f(x), \tag{2.1}$$

$$\begin{aligned}
\mathrm{Var}[Y] &= E\left[\left(\sum_{j=0}^{k-1}(f(X_j)-\mu)\right)^2\right] \\
&= k\,E\left[(f(X_0)-\mu)^2\right] + k(k-1)\,E\left[(f(X_0)-\mu)(f(X_1)-\mu)\right] \\
&= \sum_{x=0}^{n} \binom{n}{x} \frac{(k-1)^{n-x}}{k^{n-1}} (f(x)-\mu)^2 \qquad (2.2)\\
&\quad + \sum_{x=0}^{\lfloor n/2\rfloor} \binom{n}{x}\binom{n-x}{x} \frac{(k-1)(k-2)^{n-2x}}{k^{n-1}} (f(x)-\mu)^2 \\
&\quad + 2\sum_{x=0}^{n} \sum_{y=0}^{\min(n-x,\,x-1)} \binom{n}{x}\binom{n-x}{y} \frac{(k-1)(k-2)^{n-x-y}}{k^{n-1}} \\
&\qquad \cdot (f(x)-\mu)(f(y)-\mu).
\end{aligned}$$

Although they contain a lot of summands, these formulas are practical in the sparse case since, for the Y's defined in Table 2.1, when n and k are large and $\lambda = n/k$ is small, only the terms for small x and y in the above sums are nonnegligible. These terms converge to 0 exponentially fast as a function of $x+y$ when $x+y\to\infty$. The first two moments of Y are then easy to compute by truncating the sums after a small number of terms. For example, with $n = k = 1000$, the relative errors on $E[H]$ and $\mathrm{Var}[H]$ are less than 10^{-10} if the sums are stopped at $x, y = 14$ instead of 1000 and less than 10^{-15} if the sums are stopped at $x, y = 18$. A similar behavior is observed for the other statistics.

Expressions (2.1) and (2.2) still are valid in the dense case but, for larger λ, more terms need to be considered. Approximations for the mean and variance of D_δ when $\lambda \gg 1$, with error terms in $o(1/n)$, are provided in [39, Chap. 5, p. 65].

In the Poisson setup, where n is the mean of a Poisson random variable, the X_j are i.i.d. Poisson(λ) and the expressions become

$$E[Y] = k\mu \;=\; k\sum_{x=0}^{\infty} \frac{\lambda^x e^{-\lambda}}{x!} f(x), \tag{2.3}$$

$$\text{Var}[Y] = k \sum_{x=0}^{\infty} \frac{\lambda^x e^{-\lambda}}{x!} (f(x) - \mu)^2. \tag{2.4}$$

2.3. Limit theorems. The limiting distribution of D_δ is chi-square in the dense case and normal in the sparse case. *Two-moment-corrected* versions of these results are stated in the next proposition, where $D_\delta^{(C)}$ and $D_\delta^{(N)}$ have exactly the same mean and variance as their asymptotic distributions (e.g., 0 and 1 in the normal case). Read and Cressie [39] recommend this type of standardization, which tends to be closer to the asymptotic distribution than a standardization by the asymptotic mean and variance. The two-moment corrections become increasingly important when δ gets away from around 1. The mean and variance of D_δ can be computed as explained in the previous subsection. Another possibility would be to correct the distribution itself, e.g., using Edgeworth-type expansions [39, p. 68]. This gives extremely complicated expressions, due in part to the discrete nature of the multinomial distribution, and the gain is small.

PROPOSITION 2.1. *For $\delta > -1$, the following holds under $\mathcal{H}_0$:*

(i) (Dense case.) *If k is fixed and $n \to \infty$, then in the multinomial setup,*

$$D_\delta^{(C)} \stackrel{\text{def}}{=} \frac{D_\delta - k\mu + (k-1)\sigma_C}{\sigma_C} \Rightarrow \chi^2(k-1),$$

where $\sigma_C^2 = \text{Var}[D_\delta]/(2(k-1))$, $\Rightarrow$ denotes convergence in distribution, and $\chi^2(k-1)$ is the chi-square distribution with $k-1$ degrees of freedom. In the Poisson setup, $D_\delta^{(C)} \Rightarrow \chi^2(k)$ instead.

(ii) (Sparse case.) *For both the multinomial and Poisson setups, if $k \to \infty$, $n \to \infty$, and $n/k \to \lambda_0$ where $0 < \lambda_0 < \infty$, then*

$$D_\delta^{(N)} \stackrel{\text{def}}{=} \frac{D_\delta - k\mu}{\sigma_N} \Rightarrow N(0,1),$$

where $\sigma_N^2 = \text{Var}[D_\delta]$, and $N(0,1)$ is the standard normal distribution.

Proof. For the multinomial setup, part (i) can be found in [39, p. 46], whereas part (ii) follows from Theorem 1 of [11] by noting that all the X_j's here have the same distribution. The proofs simplify for the Poisson setup, due to the independence. The $Z_j = (X_j - n/k)/\sqrt{n/k}$ are i.i.d. and asymptotically $N(0,1)$ in the dense case, so their sum of squares, which is X^2, is asymptotically $\chi^2(k)$. □

We now turn to the *counting* random variables N_b, W_b, and C. These are *not* approximately chi-square in the dense case. In fact, if $n \to \infty$ for fixed k, it is clear that $N_b \to 0$ with probability 1 for any fixed b. This implies that $W_b \to k$ and $C \to n - k$, so these random variables are all degenerate.

For the Poisson setup, each X_i is Poisson(λ), so $p_b \stackrel{\text{def}}{=} P[X_i = b] = e^{-\lambda}\lambda^b/b!$ for $b \geq 0$ and N_b is BN(k, p_b), a binomial with parameters k and p_b. If k is large and p_b is small, N_b is thus approximately Poisson with (exact) mean

$$E[N_b] = kp_b = \frac{n^b e^{-\lambda}}{k^{b-1} b!} \quad \text{for } b \geq 0. \tag{2.5}$$

The next result covers other cases as well.

PROPOSITION 2.2. *For the Poisson or the multinomial setup, under $\mathcal{H}_0$, suppose that $k \to \infty$ and $n \to \infty$, and let λ_∞, γ_0, and λ_0 denote positive constants.*

(i) *If* $b \geq 2$ *and* $n^b/(k^{b-1}b!) \to \lambda_\infty$, *then* $W_b \Rightarrow N_b \Rightarrow \text{Poisson}(\lambda_\infty)$. *For* $b = 2$, *one also has* $C \Rightarrow N_2$.
(ii) *For* $b = 0$, *if* $n/k - \ln(k) \to \gamma_0$, *then* $N_0 \Rightarrow \text{Poisson}(e^{-\gamma_0})$.
(iii) *If* $k \to \infty$ *and* $n/k \to \lambda_0 > 0$, *then for* $Y = N_b$, W_b, *or* C,

$$\frac{Y - E[Y]}{(\text{Var}[Y])^{1/2}} \Rightarrow N(0,1).$$

Proof. In (i), since $\lambda = n/k \to 0$, one has for the Poisson case $E[N_{b+1}]/E[N_b] = \lambda/(b+1) \to 0$ and $E[W_{b+1}]/E[N_b] = E[\sum_{i=1}^{\infty} N_{b+i}]/E[N_b] = \sum_{i=1}^{\infty} \lambda^i b!/(b+i)! \leq b! \sum_{i=1}^{\infty} \lambda^i/i! = b!(e^\lambda - 1) \to 0$. The relative contribution of W_{b+1} to the sum $W_b = N_b + W_{b+1}$ (a sum of correlated Poisson random variables) is then negligible compared with that of N_b, so N_b and W_b have the same asymptotic distribution (this follows from Lemma 6.2.2 of [2]). Likewise, under these conditions with $b = 2$, C has the same asymptotic distribution as N_2 because $C = N_2 + \sum_{i=3}^{\infty}(i-1)N_i$, and therefore $E[C - N_2]/E[N_2] = E[\sum_{i=3}^{\infty}(i-1)N_i]/E[N_2] = 2\sum_{i=3}^{\infty}(i-1)\lambda^{i-2}/i! < \sum_{j=1}^{\infty}\lambda^j/j! = e^\lambda - 1$. For the multinomial setup, it has been shown (see [2, section 6.2]) that N_b and W_b, for $b \geq 2$, are asymptotically Poisson(kp_b) when $\lambda \to 0$, the same as for the Poisson setup. The argument for W_2 also applies for C, again using Lemma 6.2.2 of [2], and this proves (i). For $b = 0$, for the Poisson setup, we saw already that N_0 is asymptotically Poisson(λ_∞) if $ke^{-n/k} \to \lambda_\infty$, i.e., if $\ln(k) - (n/k) \to \ln(\lambda_\infty) = -\gamma_0$. For the multinomial case, the same result follows from Theorem 6.D of [2], and this proves (ii). Part (iii) is obtained by applying Theorem 1 of [11]. □

The exact distributions of C and N_0 under $\mathcal{H}_0$, for the multinomial setup, are given by

$$P(C = c) = P(N_0 = k - n + c) = \frac{k(k-1)\cdots(k-n+c+1)}{k^n} \left\{ \begin{array}{c} n \\ n-c \end{array} \right\},$$

where $\left\{ \begin{array}{c} k \\ n \end{array} \right\}$ are the Stirling numbers of the second kind (see [13, p. 71], where an algorithm is given to compute all the nonnegligible probabilities in time $O(n \log n)$).

In our implementation of the test based on C, we used the Poisson approximation for $\lambda \leq 1/32$, the normal approximation for $\lambda > 1/32$ and $n > 2^{15}$, and the exact distribution otherwise.

3. Overlapping vectors. For the overlapping case, let $X_{t,j}^{(o)}$ be the number of overlapping vectors V_i, $i = 0, \ldots, n-1$, that fall into cell j. Now, formulas (2.1) and (2.2) for the mean and variance, and the limit theorems in Propositions 2.1 and 2.2, no longer stand. The analysis is more difficult than for the disjoint case because, in general, $P[X_{t,i}^{(o)} = x]$ depends on i and $P[X_{t,i}^{(o)} = x, X_{t,j}^{(o)} = y]$ depends on the pair (i,j) in a nontrivial way.

Theoretical results are available in the overlapping multinomial setup for the Pearson statistic in the dense case. Let

$$X_{(t)}^2 = \sum_{j=0}^{k-1} \frac{(X_{t,j}^{(o)} - n/k)^2}{n/k}$$

and let $X_{(t-1)}^2$ be the equivalent of $X_{(t)}^2$ for the overlapping vectors of dimension $t-1$,

$$X_{(t-1)}^2 = \sum_{j=0}^{k'-1} \frac{(X_{t-1,j}^{(o)} - n/k')^2}{n/k'},$$

where $k' = d^{t-1}$. Consider the statistic $\tilde{X}^2 = X^2_{(t)} - X^2_{(t-1)}$. Good [8] has shown that $E[X^2_{(t)}] = d^t - 1$ exactly (see his equation (5) and top of page 280) and that when $n \to \infty$ for d and t fixed, $\tilde{X}^2 \Rightarrow \chi^2(k - k')$ (see page 284). This setup, usually with $n/k \geq 5$ or so is called the *overlapping serial test* or the *m-tuple* test in the literature and has been used previously to test RNGs (e.g., [1, 29, 30]). The next proposition generalizes the result of Good to the power divergence statistic in the dense case. Further generalization is given in [43].

PROPOSITION 3.1. *Let*

$$D_{\delta,(t)} = \sum_{j=0}^{k-1} \frac{2}{\delta(1+\delta)} X^{(o)}_{t,j} \left[(X^{(o)}_{t,j}/\lambda)^\delta - 1 \right] \tag{3.1}$$

be the power divergence statistic for the t-dimensional overlapping vectors, and define $\tilde{D}_{\delta,(t)} = D_{\delta,(t)} - D_{\delta,(t-1)}$. *Under* $\mathcal{H}_0$, *in the multinomial setup, if* $\delta > -1$, k *is fixed, and* $n \to \infty$, *then* $\tilde{D}_{\delta,(t)} \Rightarrow \chi^2(d^t - d^{t-1})$.

Proof. The result is well known for $\delta = 1$. Moreover, a Taylor series expansion of $D_{\delta,(t)}$ in powers of $X^{(o)}_{t,j}/\lambda - 1$ easily shows that $D_{\delta,(t)} = D_{1,(t)} + o_p(1)$, where $o_p(1) \to 0$ in probability as $n \to \infty$ (see [39, Thm. A6.1]). Therefore, $\tilde{D}_{\delta,(t)}$ has the same asymptotic distribution as $\tilde{D}_{1,(t)}$ and this completes the proof. □

For the sparse case, where $k, n \to \infty$ and $n/k \to \lambda_0$ with $0 < \lambda_0 < \infty$, our simulation experiments support the conjecture that

$$\tilde{X}^2_N \stackrel{\text{def}}{=} \frac{\tilde{X}^2 - (k - k')}{\sqrt{2(k - k')}} \Rightarrow N(0, 1).$$

The overlapping empty-cells-count test has been discussed in a heuristic way in a number of papers. For $t = 2$, Marsaglia [30] calls it the *overlapping pairs sparse occupancy* (OPSO) and suggests a few specific parameters without providing the underlying theory. Marsaglia and Zaman [32] speculate that N_0 should be approximately normally distributed with mean $ke^{-\lambda}$ and variance $ke^{-\lambda}(1 - 3e^{-\lambda})$. This make sense only if λ is not too large or not too close to zero. We studied empirically this approximation and found it reasonably accurate only for $2 \leq \lambda \leq 5$ (approximately). The approximation could certainly be improved by refining the variance formula.

Proposition 2.2(i) and (ii) should hold in the overlapping case as well. Our simulation experiments indicate that the Poisson approximation for C is very accurate for, say, $\lambda < 1/32$, and already quite good for $\lambda \leq 1$, when n is large.

4. Which test statistic to use and what to expect. The linear feedback shift register (LFSR) generator, LCG, and multiple recursive generator (MRG) in our lists are constructed so that their point sets Ψ_t over the entire period are *superuniformly* distributed. Thus, we may be afraid, if k is large enough, that very few cells (if any) contain more than 1 point and that D_δ, C, N_0, N_b, and W_b for $b \geq 2$ are smaller than expected. In the extreme case where $C = 0$, assuming that the distribution of C under $\mathcal{H}_0$ is approximately Poisson with mean $n^2/(2k)$, the left p-value of the collision test is $p_{\mathrm{L}} = P[C \leq 0 \mid \mathcal{H}_0] \approx e^{-n^2/(2k)}$. For a fixed number of cells, this p-value approaches 0 exponentially fast in the square of the sample size n. For example, $p_{\mathrm{L}} \approx 3.3 \cdot 10^{-4}$, $1.3 \cdot 10^{-14}$, and $3.4 \cdot 10^{-56}$ for $n = 4\sqrt{k}$, $8\sqrt{k}$, and $16\sqrt{k}$, respectively. Assuming that k is near the RNG's period length, i.e., $k \approx 2^e$, this means that the test begins to fail abruptly when the sample size exceeds approximately 4 times the square root of

the period length. As we shall see, this is precisely what happens for certain popular classes of generators. If we use the statistic W_b, instead of C, in the same situation, we have $p_{\mathrm{L}} = P[W_b \le 0 \mid \mathcal{H}_0] \approx e^{-n^b/(k^{b-1}b!)}$, and the sample size required to obtain a p-value less than a fixed (small) constant is $n = O(k^{(b-1)/b})$ for $b \ge 2$. In this setup, C and N_2 are equivalent to W_2 and choosing $b > 2$ gives a less efficient test.

Suppose now that we have the opposite situation: too many collisions. One simple model of this situation is the following alternative $\mathcal{H}_1$: The points are i.i.d. uniformly distributed over k_1 boxes, the other $k - k_1$ boxes being always empty. Under $\mathcal{H}_1$, W_b is approximately Poisson with mean $\lambda_1 = n^b e^{-n/k_1}/(k_1^{b-1}b!)$ (if n is large and λ_1 is small) instead of $\lambda_0 = n^b e^{-n/k}/(k^{b-1}b!)$. Therefore, for a given α_0, and x_0 such that $\alpha_0 = P[W_b \ge x_0 \mid \mathcal{H}_0]$, the power of the test at level α_0 is

$$P[W_b \ge x_0 \mid \mathcal{H}_1] \approx 1 - \sum_{x=0}^{x_0-1} \frac{e^{-\lambda_1}\lambda_1^x}{x!},$$

where x_0 depends on b. When b increases, for a fixed α_0, x_0 decreases and λ_1 decreases as well if $n/k_1 \le b + 1$. So $b = 2$ maximizes the power unless n/k_1 is large. In fact, the test can have significant power only if λ_1 exceeds a few units (otherwise, with large probability, one has $W_b = 0$ and $\mathcal{H}_0$ is not rejected). This means $\lambda_1 = O(1)$, i.e., $n = O(k_1^{(b-1)/b}(b!)^{1/b}e^{n/(bk_1)})$, which can be approximated by $O(k_1^{(b-1)/b})$ if k_1 is reasonably large. Then, $b = 2$ is the best choice. If k_1 is small, λ_1 is maximized (approximately) by taking $b = \max(2, \lceil n/k_1 \rceil - 1)$.

The alternative $\mathcal{H}_1$ just discussed can be generalized as follows: Suppose that the k_1 cells have a probability larger than $1/k$, while the other $k - k_1$ cells have a smaller probability. $\mathcal{H}_1$ is called a *hole* (resp., *peak*, *split*) alternative if k_1/k is near 1 (resp., near 0, near 1/2). We made extensive numerical experiments regarding the power of the tests under these alternatives and found the following: Hole alternatives can be detected only when n/k is reasonably large (dense case) because in the sparse case one expects several empty cells anyway. The best test statistics for detecting them are those based on the number of empty cells N_0 and D_δ with δ as small as possible (e.g., $-1 < \delta \le 0$). For a peak alternative, the power of D_δ increases with δ as a concave function, with a rate of increase that typically becomes very small for δ larger than 3 or 4 (or higher, if the peak is very narrow). The other test statistics in Table 2.1 usually are not competitive with, say, D_4, under this alternative, except for W_b which comes close when $b \approx n/k_1$ (however, it is hard to choose the right b because k_1 is generally unknown). The split alternative with the probability of the $k - k_1$ low-probability cells equal to 0 is easy to detect, and the *collision test* (using C or W_2) is our recommendation. The power of D_δ is essentially the same as that of C and W_2, for most δ, because $E[W_3]$ has a negligible value, which implies that there is almost a one-to-one correspondence between C, W_2, and D_δ. However, with the small n that suffices for detection in this situation, $E[W_2]$ is small and the distribution of D_δ is concentrated on a small number of values, so neither normal nor chi-square is a good approximation of its distribution. Of course, the power of the test would improve if the high-probability cells were aggregated into a smaller number of cells, and similarly for the low-probability cells. But to do this, one needs to know where these cells are a priori.

These observations extend (and agree with) those made previously by several authors (see [39] and references therein), who already noted that for D_δ the power decreases with δ for a hole alternative and increases with δ for a peak alternative. This implies, in particular, that G^2 and H are better (resp., worse) test statistics than X^2

for detecting a hole (resp., a peak). In the case of a split alternative for which the cell probabilities are only slightly perturbed, X^2 is optimal in terms of Pitman's asymptotic efficiency, whereas G^2 is optimal in terms of Bahadur's efficiency (see [39] for details).

5. Empirical evaluation for RNG families.

5.1. Selected families of RNGs. We now report systematic experiments to assess the effectiveness of serial tests for detecting the regularities in specific families of *small* RNGs. The RNG families that we consider are named LFSR3, GoodLCG, BadLCG2, MRG2, CombL2, and InvExpl. Within each family we constructed a list of specific RNG instances, with period lengths near 2^e for (integer) values of e ranging from 10 to 40. These RNGs are too small to be considered for serious general purpose software, but their study gives a good indication of the behavior of larger instances from the same families. At step n, a generator outputs a number $u_n \in [0, 1)$.

The LFSR3s are combined linear feedback shift register (LFSR) (or Tausworthe) generators with three components, of the form

$$\begin{aligned} x_{j,n} &= (a_{r_j} x_{j,n-r_j} + a_{k_j} x_{j,n-k_j}) \bmod 2, \quad 1 \le j \le 3, \\ u_{j,n} &= \sum_{i=1}^{32} x_{j,ns_j+i-1} 2^{-i}, \quad 1 \le j \le 3, \\ u_n &= u_{1,n} \oplus u_{2,n} \oplus u_{3,n}, \end{aligned}$$

where $\oplus$ means bitwise exclusive-or, and (k_j, r_j, s_j), $1 \le j \le 3$, are constant parameters selected so that the k_j's are reasonably close to each other and the sequence $\{u_n\}$ has period length $(2^{k_1} - 1)(2^{k_2} - 1)(2^{k_3} - 1)$ and is maximally equidistributed (see [19] for the definition and for further details on these generators).

The GoodLCGs are linear congruential generators (LCGs) of the form

$$x_n = a x_{n-1} \bmod m; \qquad u_n = x_n/m, \tag{5.1}$$

where m is a prime near 2^e and a is selected so that the period length is $m - 1$ and the LCG exhibits excellent behavior with respect to the spectral test (i.e., an excellent lattice structure) in up to at least 8 dimensions. The BadLCG2s have the same structure except that their a is chosen so that they have a mediocre lattice structure in 2 dimensions. More details and the values of a and m can be found in [24, 26]. The MRG2s are multiple recursive generators of order 2 of the form

$$x_n = (a_1 x_{n-1} + a_2 x_{n-2}) \bmod m; \qquad u_n = x_n/m, \tag{5.2}$$

with period length $m^2 - 1$ and excellent lattice structure, as for the GoodLCGs [17, 21].

The CombL2s combine two LCGs as proposed in [15],

$$\begin{aligned} x_{j,n} &= a_j x_{j,n-1} \bmod m_j, \quad 1 \le j \le 2; \\ u_n &= ((x_{1,n} + x_{2,n}) \bmod m_1)/m_1 \end{aligned}$$

so that the combined generator has period length $(m_1 - 1)(m_2 - 1)/2$ and an excellent lattice structure (see [28] for details on that lattice structure).

InvExpl denotes a family of explicit inversive nonlinear generators of period length m, defined by

$$x_n = (123n)^{-1} \bmod m; \qquad u_n = x_n/m, \tag{5.3}$$

where m is prime and $(an)^{-1} \bmod m = (an)^{m-2} \bmod m$ (see [5]).

TABLE 5.1

The log-p-values ℓ for the GoodLCGs with period length $\rho \approx 2^e$, for the collision test (based on C) in $t = 2$ dimensions, with $k \approx 2^e$ cells and sample size $n = 2^{e/2+\nu}$. The table entries give the values of ℓ. The symbols $\leftarrow$ and $\rightarrow$ mean $\ell \le -14$ and $\ell \ge 14$, respectively. Here, we have $\tilde{\nu} = 2$ and $\nu^* = 4$.

e	$\nu = 1$	$\nu = 2$	$\nu = 3$	$\nu = 4$	$\nu = 5$
12			-3	$\leftarrow$	$\leftarrow$
13				-11	$\leftarrow$
14			-4	-13	$\leftarrow$
15		-2	-7	$\leftarrow$	$\leftarrow$
16		-3	-8	$\leftarrow$	$\leftarrow$
17		-2	-4	$\leftarrow$	$\leftarrow$
18			-3	$\leftarrow$	$\leftarrow$
19		-2	-10	$\leftarrow$	$\leftarrow$
20		-3	-9	$\leftarrow$	$\leftarrow$
21		-2	-4	$\leftarrow$	$\leftarrow$
22			-3	-11	$\leftarrow$
23		-3	-8	$\leftarrow$	$\leftarrow$
24		-3	-12	$\leftarrow$	$\leftarrow$
25			-8	$\leftarrow$	$\leftarrow$
26		-2	-6	$\leftarrow$	$\leftarrow$
27		-2	-6	$\leftarrow$	$\leftarrow$
28			-2	$\leftarrow$	$\leftarrow$
29		-3	-13	$\leftarrow$	$\leftarrow$
30			-5	$\leftarrow$	$\leftarrow$

5.2. The log-*p*-values. For a given test statistic Y taking value y, let $p_{\mathrm{L}} = P[Y \le y \mid \mathcal{H}_0]$ and $p_{\mathrm{R}} = P[Y \ge y \mid \mathcal{H}_0]$. We define the *log-p-value* of the test as

$$\ell = \begin{cases} k & \text{if } 10^{-(k+1)} < p_{\mathrm{R}} \le 10^{-k}, \quad k > 0, \\ -k & \text{if } 10^{-(k+1)} < p_{\mathrm{L}} \le 10^{-k}, \quad k > 0, \\ 0 & \text{otherwise.} \end{cases}$$

For example, $\ell = 2$ means that the right p-value is between 0.01 and 0.001. For a given class of RNGs, given Y, t, and a way of choosing k, we apply the test for different values of e and with sample size $n = 2^{\gamma e+\nu}$ for $\nu = \ldots, -2, -1, 0, 1, 2, \ldots$, where the constant γ is chosen so that the test begins to fail at approximately the same value of ν for all (or most) e. More specifically, we define $\tilde{\nu}$ (resp., ν^*) as the smallest values of ν for which the absolute log-p-value satisfies $|\ell| \ge 2$ (resp., $|\ell| \ge 14$) for a majority of values of e. These thresholds are arbitrary.

5.3. Test results: Examples and summary. Tables 5.1 and 5.2 give the log-p-values for the collision test applied to the GoodLCGs and BadLCG2s, respectively, in $t = 2$ dimensions with $d = \lfloor 2^{e/2} \rfloor$ (so $k \approx 2^e$) and $n = 2^{e/2+\nu}$. Only the log-p-values ℓ outside the set $\{-1, 0, 1\}$, which correspond to p-values less than 0.01, are displayed. The symbols $\leftarrow$ and $\rightarrow$ mean $\ell \le -14$ and $\ell \ge 14$, respectively. The columns not shown are mostly blank on the left of the tables and filled with arrows on the right of the tables. The small p-values appear with striking regularity at about the same ν for all e in each of these tables. This is also true for other values of e not shown in the tables. One has $\tilde{\nu} = 2$ and $\nu^* = 4$ in Table 5.1, while $\tilde{\nu} = -1$ and $\nu^* = 1$ in Table 5.2. The GoodLCGs fail because their structure is too regular (the left p-values are too small because there are too few collisions), whereas the BadLCG2s have the opposite

TABLE 5.2

The log-p-values ℓ for the collision test, with the same setup as in Table 5.1, but for the BadLCG2 generators. Here, $\tilde{\nu} = -1$ and $\nu^* = 1$.

e	$\nu = -1$	$\nu = 0$	$\nu = 1$	$\nu = 2$
12	2	2	→	→
13		3	9	→
14	5	4	→	→
15	3	4	→	→
16	3	12	→	→
17		7	→	→
18		13	→	→
19		3	→	→
20		4	→	→
21		4	→	→
22	2	11	→	→
23	2	4	11	→
24		11	→	→
25		4	→	→
26	3		→	→
27	3	4	→	→
28		13	→	→
29	2	2	→	→
30	2	→	→	→

TABLE 5.3

Collision tests for RNG families, in t dimensions, with $k \approx 2^e$. Recall that $\tilde{\nu}$ (resp., ν^*) is the smallest integer ν for which $|\ell| \geq 2$ (resp., $|\ell| \geq 14$) for a majority of values of e in tests with sample size $n = 2^{\gamma e + \nu}$.

RNG family	γ	t	$\tilde{\nu}$	ν^*
GoodLCG	1/2	2	2	4
		4	3	5
BadLCG2	1/2	2	−1	1
		4	3	5
LFSR3, d odd	1/2	2	3	5
		4	4	6
LFSR3, d power of 2	1/2	2	2	4
		4	3	4
MRG2	1/2	2	2	4
		4	3	5
CombL2	1/2	2	3	5
		4	5	7
InvExpl	1	2	1	1
		4	1	1

behavior (the right p-values are too small because there are too many collisions; their behavior corresponds to the *split* alternative described in section 4).

Table 5.3 gives the values of $\tilde{\nu}$ and ν^* for the selected RNG families for the collision test in 2 and 4 dimensions. All families, except InvExpl, fail at a sample size proportional to the square root of the period length ρ. At $n = 2^{\nu^*}\rho^{1/2}$, the left or right p-value is less than 10^{-14} most of the time. The BadLCG2s in 2 dimensions are the first to fail: they were chosen to be particularly mediocre in 2 dimensions and the test detects it. Apart from the BadLCG2s, the generators always fail the tests

TABLE 5.4
Collision tests with $d = 4$ divisions in each dimension and $t = \lfloor e/2 \rfloor$ dimensions.

Generators	γ	$\tilde{\nu}$	ν^*
GoodLCG	2/3	2	3
BadLCG2	2/3	2	4
LFSR3	1/2	2	4
MRG2	1/2	7	8
CombL2	1/2	5	6
InvExpl	1	1	1

due to excessive regularity. For the GoodLCGs and LFSR3s, for example, there was never a cell with more than 2 points in it. For the LFSR3s, we distinguish two cases: one where d was chosen always odd and one where it was always the smallest power of 2 such that $k = d^t \geq 2^e$. In the latter case, the number of collisions is always 0, since no cell contains more than a single point over the entire period of the generator as a consequence of the "maximal equidistribution" property of these generators [19]. The left p-values then behave as described at the beginning of section 4. The InvExpl resist the tests until after their period length is exhausted. The point sets Ψ_t of these generators appear random instead of very evenly distributed. However, they are much slower than the linear generators.

We applied the power divergence tests with $\delta = -1/2, 0, 1, 2, 4$, and in most cases the p-values were very close to those of the collision test. In fact, when no cell count X_i exceeds 2 (i.e., $W_3 = 0$, which we have observed frequently), there is a one-to-one correspondence between the values of C and D_δ for all $\delta > -1$. Therefore, all these statistics should have similar p-values if both $E[W_3]$ and the observed value of W_3 are small (the very sparse situation). For the overlapping versions of the tests, the values of γ, $\tilde{\nu}$, and ν^* are exactly the same as those given in Table 5.3. This means that the overlapping tests are more efficient than the nonoverlapping ones because they call the RNG t times less often.

We applied the same tests with smaller and larger numbers of cells, such as $k = 2^e/64$, $k = 2^e/8$, $k = 8 \cdot 2^e$, $k = 64 \cdot 2^e$, and found that $\tilde{\nu}$ and ν^* increase when k moves away from 2^e. A typical example is that for the GoodLCGs with $t = 2$, we have $\nu^* = 7$, 6, 5, and 7 for the four choices of k given above, respectively, whereas $\nu^* = 4$ when $k = 2^e$. The *classical* way of applying the serial test for RNG testing uses a large average number of points per cell (dense case). We applied the test based on X^2 to the GoodLCGs, with $k \approx n/8$ and found empirically $\gamma = 2/3$, $\tilde{\nu} = 3$, and $\nu^* = 4$. This means that the required sample size now increases as $O(\rho^{2/3})$ instead of $O(\rho^{1/2})$ as before; i.e., the dense setup with the chi-square approximation is much less efficient than the sparse setup. We observed the same for D_δ with other values of δ and t and observed a similar behavior for other RNG families.

For the results just described, t was fixed and d varied with e. We now fix $d = 4$ (i.e., we take the first two bits of each number) and vary the dimension as $t = \lfloor e/2 \rfloor$. Table 5.4 gives the results of the collision test in this setup. Note the change in γ for the GoodLCGs and BadLCG2s: the tests are less sensitive for these large values of t.

We also experimented with two-level tests, where a test of sample size n is replicated independently N times. For the collision test, we use the test statistic C_T, the total number of collisions over N replications, which is approximately Poisson with mean $Nn^2e^{-n/k}/(2k)$ under $\mathcal{H}_0$. For the power divergence tests, we use as test

TABLE 6.1
List of selected generators.

LCG1.	LCG with $m = 2^{31} - 1$ and $a = 950706376$, $c = 0$.
LCG2.	LCG with $m = 2^{31} - 1$ and $a = 742938285$, $c = 0$.
LCG3.	LCG with $m = 2^{31} - 1$ and $a = 630360016$, $c = 0$.
LCG4.	LCG with $m = 2^{31} - 1$ and $a = 16807$, $c = 0$.
LCG5.	LCG with $m = 2^{31}$, $a = 1103515245$, $c = 12345$.
LCG6.	LCG with $m = 2^{32}$, $a = 69069$, and $c = 1$.
LCG7.	LCG with $m = 2^{48}$, $a = 68909602460261$, $c = 0$.
LCG8.	LCG with $m = 2^{48}$, $a = 44485709377909$, $c = 0$.
LCG9.	LCG with $m = 2^{48}$, $a = 25214903917$, $c = 11$.
RLUX.	RANLUX with $L = 24$ (see [12]).
WEY1.	Nested Weyl with $\alpha = \sqrt{2}$ (see [10]).
WEY2.	Shuffled nested Weyl with $\alpha = \sqrt{2}$ (see [10]).
CLCG4.	Combined LCG of [25].
CMRG96.	Combined MRG in Fig. 1 of [18].
CMRG99.	Combined MRG in Fig. 1 of [23].

statistics the sum of values of $D_\delta^{(N)}$ and $D_\delta^{(C)}$, which are approximately $N(0, N)$ and $\chi^2(N(k-1))$ under $\mathcal{H}_0$, respectively. We observed the following: The power of a test with (N, n) typically is roughly the same as that of the same test at level one ($N = 1$) and with sample size $n\sqrt{N}$. Single-level tests thus need a smaller total sample size than the two-level tests to achieve the same power. On the other hand, two-level tests are justified when the sample size n is limited by the memory size of the computer at hand. (For $n \ll k$, the counters X_j are implemented via a *hashing* technique for which the required memory is proportional to n instead of k.) Another way of doing a two-level test with D_δ is to compute the p-values for the N replicates and compare their distribution with the uniform distribution via a Kolmogorov–Smirnov or Anderson–Darling goodness-of-fit test, for example. We experimented extensively with this as well and found no advantage in terms of efficiency for all the RNG families that we tried.

6. What about real-life LCGs? From the results of the preceding section one can easily predict, conservatively, at which sample size a specific RNG from a given family will begin to fail. We verify this with a few commonly used RNGs listed in Table 6.1. (Of course, this list is far from exhaustive.)

Generators LCG1 to LCG9 are well-known LCGs, based on the recurrence $x_i = (ax_{i-1} + c) \bmod m$, with output $u_i = x_i/m$ at step i. LCG1 and LCG2 are recommended by Fishman and Moore [7], and a FORTRAN implementation of LCG1 is given by Fishman [6]. LCG3 is recommended in [14], among others, and is used in the SIMSCRIPT II.5 and INSIGHT simulation languages. LCG4 is in numerous software systems, including the IBM and Macintosh operating systems, the Arena and SLAM II simulation languages (note that the Arena RNG was replaced by CMRG99 after we wrote this paper), MATLAB, the IMSL library (which also provides LCG1 and LCG5), the Numerical Recipes [38], etc., and is suggested in several books and papers (e.g., [3, 36, 40]). LCG6 is used in the VAX/VMS operating system and on Convex computers. LCG5 and LCG9 are the `rand` and `rand48` functions in the standard libraries of the C programming language [37]. LCG7 is taken from [6] and LCG8 is used in the CRAY system library. LCG1 to LCG4 have period length $2^{31} - 2$; LCG5, LCG6, AND LCG9 have period length m; and LCG7 and LCG8 have period length $m/4 = 2^{46}$.

TABLE 6.2
The log-p-values for the collision test in $t = 2$ dimensions, with $k = m$ cells, and sample size $n = 2^\nu\sqrt{m}$.

Generator	$\nu = 1$	$\nu = 2$	$\nu = 3$	$\nu = 4$	$\nu = 5$
LCG1		-2	-11	$\leftarrow$	$\leftarrow$
LCG2		-3	-8	$\leftarrow$	$\leftarrow$
LCG3			-3	$\leftarrow$	$\leftarrow$
LCG4		2	4	$\rightarrow$	$\rightarrow$
LCG5		-3	-13	$\leftarrow$	$\leftarrow$
LCG6		-3	-7	$\leftarrow$	$\leftarrow$

TABLE 6.3
The log-p-values for the two-level collision test (based on C_T) in $t = 2$ dimensions, with $k = 2^{46}$ cells, sample size $n = 2^{24+\nu}$ for each replication, and $N = 32$ replications.

Generator	$\nu = -2$	$\nu = -1$	$\nu = 0$
LCG7		-7	$\leftarrow$
LCG8		-7	$\leftarrow$
LCG9		-3	-3

RLUX is the RANLUX generator implemented by James [12], with luxury level $L = 24$. At this luxury level, RANLUX is equivalent to the subtract-with-borrow generator with modulus $b = 2^{32} - 5$ and lags $r = 43$ and $s = 22$ proposed in [31] and used, for example, in *Mathematica* (according to its documentation). WEY1 is a generator based on the nested Weyl sequence defined by $u_i = (i(i\alpha \bmod 1)) \bmod 1$, where $\alpha = \sqrt{2}$ (see [10]). WEY2 implements the shuffled nested Weyl sequence proposed in [10], defined by $\sqrt{i} = M(i(i\alpha \bmod 1) \bmod 1) + 1/2$ and $u_i = \sqrt{i}(\sqrt{i}\alpha \bmod 1) \bmod 1$, with $\alpha = \sqrt{2}$ and $M = 12345$. CLCG4, CMRG96, and CMRG99 are the combined LCG of [25], the combined MRG given in Figure 1 of [18], and the combined MRG given in Figure 1 of [23].

Table 6.2 gives the log-p-values for the collision test in two dimensions, for LCG1 to LCG6, with $k \approx m$ and $n = 2^\nu\sqrt{m}$. As expected, suspect values begin to appear at sample size $n \approx 4\sqrt{m}$ and all these LCGs are definitely rejected with $n \approx 16\sqrt{m}$. LCG4 has too many collisions, whereas the others have too few. By extrapolation, LCG7 to LCG9 are expected to begin failing with n around 2^{26}, which is just a bit more than what the memory size of our current computer allowed when we wrote this paper. However, we applied the two-level collision test with $N = 32$, $t = 2$, $k = 2^{46}$, and $n = 2^{24+\nu}$. Here, the total number of collisions C_T is approximately Poisson with mean $32n^2/(2k) \approx 64 \cdot 4^\nu$ under $\mathcal{H}_0$. The log-p-values are shown in Table 6.3. With a total sample size of $32 \cdot 2^{24}$, LCG7 and LCG8 fail decisively; they have too few collisions. We also tried $t = 4$, as well as the collision test with overlapping, and the results were similar.

We tested the other RNGs (the last 5 in the table) for several values of t ranging from 2 to 25. RLUX passed all the tests for $t \leq 24$ but failed spectacularly in 25 dimensions. With $d = 3$, $t = 25$ (so $k = 3^{25}$), and $n = 2^{24}$, the log-p-value for the collision test is $\ell = 8$ (there are 239 collisions, while $E[C|\mathcal{H}_0] \approx 166$). For a two-level test with $N = 32$, $d = 3$, $t = 25$, $n = 2^{23}$, the total number of collisions was $C_T = 1859$, much more than $32\,E[C|\mathcal{H}_0] \approx 1329$ ($\ell \geq 14$). This result is not surprising, because for this generator all the points V_i in 25 dimensions or more lie in a family of equidistant hyperplanes that are $1/\sqrt{3}$ apart (see [20, 42]). Note that

RANLUX with a larger value of L passes these tests, at least for $t \leq 25$. WEY1 passed the tests in 2 dimensions but failed spectacularly for all $t \geq 3$: the points are concentrated in a small number of boxes. For example, with $t = 3$, $k = 1000$, and a sample size as small as $n = 1024$, we observed $C = 735$, whereas $E[C|\mathcal{H}_0] \approx 383$ ($\ell \geq 14$). WEY2, CLCG4, CMRG96, and CMRG99 passed all the tests that we tried.

7. Conclusion. We compared several variants of serial tests to detect regularities in RNGs. We found that the sparse tests perform better than the usual (dense) tests in this context. The choice of the function $f_{n,k}$ does not seem to matter much. In particular, collisions count, Pearson, loglikelihood ratio, and other statistics from the power divergence family perform approximately the same in the sparse case. The overlapping tests require about the same sample size n as the nonoverlapping tests to reject a generator. The overlapping tests are thus more efficient in terms of the quantity of random numbers that need to be generated.

It is not the purpose of this paper to recommend specific RNGs. For that, we refer the reader to [22, 23, 27, 33], for example. However, our test results certainly eliminate many contenders. All LCGs and LFSRs fail these simple serial tests as soon as the sample size exceeds a few times the *square root* of their period length, regardless of the choice of their parameters. Thus, when their period length is less than 2^{50} or so, which is the case for the LCGs still encountered in many popular software products, they are easy to crack with these tests. These small generators should no longer be used. Among the generators listed in Table 6.1, only the last four passed the tests described in this paper with the sample sizes that we have tried. All others should certainly be discarded.

REFERENCES

[1] N. S. Altman, *Bit-wise behavior of random number generators*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 941–949.

[2] A. D. Barbour, L. Holst, and S. Janson, *Poisson Approximation*, Oxford Science Publications, Oxford, UK, 1992.

[3] P. Bratley, B. L. Fox, and L. E. Schrage, *A Guide to Simulation*, 2nd ed., Springer-Verlag, New York, 1987.

[4] I. Csiszár, *Information type measures of difference of probability distributions and indirect observations*, Studia Sci. Math. Hungar., 2 (1967), pp. 299–318.

[5] J. Eichenauer-Herrmann, *Inversive congruential pseudorandom numbers: A tutorial*, International Statistical Reviews, 60 (1992), pp. 167–176.

[6] G. S. Fishman, *Monte Carlo: Concepts, Algorithms, and Applications*, Springer Series in Operations Research, Springer, New York, 1996.

[7] G. S. Fishman and L. R. Moore III, *An exhaustive analysis of multiplicative congruential random number generators with modulus* $2^{31} - 1$, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 24–45.

[8] I. J. Good, *The serial test for sampling numbers and other tests for randomness*, Proc. Cambridge Philos. Soc., 49 (1953), pp. 276–284.

[9] P. E. Greenwood and M. S. Nikulin, *A Guide to Chi-Squared Testing*, Wiley, New York, 1996.

[10] B. L. Holian, O. E. Percus, T. T. Warnock, and P. A. Whitlock, *Pseudorandom number generator for massively parallel molecular-dynamics simulations*, Phys. Rev. E, 50 (1994), pp. 1607–1615.

[11] L. Holst, *Asymptotic normality and efficiency for certain goodness-of-fit tests*, Biometrika, 59 (1972), pp. 137–145.

[12] F. James, *RANLUX: A Fortran implementation of the high-quality pseudorandom number generator of Lüscher*, Comput. Phys. Comm., 79 (1994), pp. 111–114.

[13] D. E. Knuth, *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*, 3rd ed., Addison-Wesley, Reading, MA, 1998.

[14] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*, 2nd ed., McGraw-Hill, New York, 1991.
[15] P. L'Ecuyer, *Efficient and portable combined random number generators*, Comm. ACM, 31 (1988), pp. 742–749; 774. See also the correspondence in the same journal, 32 (1989), pp. 1019–1024.
[16] P. L'Ecuyer, *Testing random number generators*, in Proceedings of the 1992 Winter Simulation Conference, IEEE Press, Piscataway, NJ, 1992, pp. 305–313.
[17] P. L'Ecuyer, *Uniform random number generation*, Ann. Oper. Res., 53 (1994), pp. 77–120.
[18] P. L'Ecuyer, *Combined multiple recursive random number generators*, Oper. Res., 44 (1996), pp. 816–822.
[19] P. L'Ecuyer, *Maximally equidistributed combined Tausworthe generators*, Math. Comp., 65 (1996), pp. 203–213.
[20] P. L'Ecuyer, *Bad lattice structures for vectors of non-successive values produced by some linear recurrences*, INFORMS J. Comput., 9 (1997), pp. 57–60.
[21] P. L'Ecuyer, *Random number generation*, in Handbook of Simulation, J. Banks, ed., Wiley, New York, 1998, pp. 93–137.
[22] P. L'Ecuyer, *Uniform random number generators*, in Proceedings of the 1998 Winter Simulation Conference, IEEE Press, Piscataway, NJ, 1998, pp. 97–104.
[23] P. L'Ecuyer, *Good parameters and implementations for combined multiple recursive random number generators*, Oper. Res., 47 (1999), pp. 159–164.
[24] P. L'Ecuyer, *Tables of linear congruential generators of different sizes and good lattice structure*, Math. Comp., 68 (1999), pp. 249–260.
[25] P. L'Ecuyer and T. H. Andres, *A random number generator based on the combination of four LCGs*, Math. Comput. Simulation, 44 (1997), pp. 99–107.
[26] P. L'Ecuyer and P. Hellekalek, *Random number generators: Selection criteria and testing*, in Random and Quasi-Random Point Sets, P. Hellekalek and G. Larcher, eds., Lecture Notes in Statistics 138, Springer, New York, 1998, pp. 223–265.
[27] P. L'Ecuyer, R. Simard, E. J. Chen, and W. D. Kelton, *An object-oriented random-number package with many long streams and substreams,* Oper. Res., to appear.
[28] P. L'Ecuyer and S. Tezuka, *Structural properties for two classes of combined random number generators*, Math. Comp., 57 (1991), pp. 735–746.
[29] H. Leeb and S. Wegenkittl, *Inversive and linear congruential pseudorandom number generators in empirical tests*, ACM Trans. Modeling Comput. Simulation, 7 (1997), pp. 272–286.
[30] G. Marsaglia, *A current view of random number generators*, in Computer Science and Statistics: Proceedings of the 16th Symposium on the Interface, L. Billard, ed., Elsevier Science Publishers, North-Holland, Amsterdam, 1985, pp. 3–10.
[31] G. Marsaglia and A. Zaman, *A new class of random number generators*, Ann. Appl. Probab., 1 (1991), pp. 462–480.
[32] G. Marsaglia and A. Zaman, *Monkey tests for random number generators*, Comput. Math. Appl., 26 (1993), pp. 1–10.
[33] M. Matsumoto and T. Nishimura, *Mersenne twister: A* 623*-dimensionally equidistributed uniform pseudo-random number generator*, ACM Trans. Modeling Comput. Simulation, 8 (1998), pp. 3–30.
[34] D. Morales, L. Pardo, and I. Vajda, *Asymptotic divergence of estimates of discrete distributions*, J. Stat. Plann. Inference, 48 (1995), pp. 347–369.
[35] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF Reg. Conf. Ser. Appl. Math. 63, SIAM, Philadelphia, 1992.
[36] S. K. Park and K. W. Miller, *Random number generators: Good ones are hard to find*, Comm. ACM, 31 (1988), pp. 1192–1201.
[37] P. J. Plauger, *The Standard C Library*, Prentice-Hall, Englewood Cliffs, NJ, 1992.
[38] W. H. Press and S. A. Teukolsky, *Portable random number generators*, Comp. Phys., 6 (1992), pp. 522–524.
[39] T. R. C. Read and N. A. C. Cressie, *Goodness-of-Fit Statistics for Discrete Multivariate Data*, Springer Series in Statistics, Springer, New York, 1988.
[40] B. D. Ripley, *Thoughts on pseudorandom number generators*, J. Comput. Appl. Math., 31 (1990), pp. 153–163.
[41] M. S. Stephens, *Tests for the uniform distribution*, in Goodness-of-Fit Techniques, R. B. D'Agostino and M. S. Stephens, eds., Marcel Dekker, New York, Basel, 1986, pp. 331–366.
[42] S. Tezuka, P. L'Ecuyer, and R. Couture, *On the add-with-carry and subtract-with-borrow random number generators*, ACM Trans. Modeling Comp. Simulation, 3 (1994), pp. 315–331.
[43] S. Wegenkittl, *A generalized ϕ-divergence for asymptotically multivariate normal models*, J. Multivariate Anal., to appear.