27. Random Number Generators

"Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

- John von Neumann

In contemporary computation there is an almost unquenchable thirst for random numbers. One particularly intemperate class of customers is comprised of the diverse Monte Carlo methods.¹ Or one may want to study a problem (or control a process) that depends on several parameters which one doesn’t know how to choose. In such cases random choices are often preferred, or simply convenient. Finally, in system analysis (including biological systems) random “noise” is often a preferred test signal. And, of course, random numbers are useful – to say the least – in cryptography.

In using arithmetical methods for generating “random” numbers great care must be exercised to avoid falling into deterministic traps. Such algorithms never produce truly random events (such as the clicks of a Geiger counter near a radioactive source), but give only pseudorandom effects. This is nicely illustrated by the following (true) story.

An associate of the author, for a study in human vision, wanted to generate visual noise or “snow” (as seen on the screen of a broken TV set). The unwary researcher selected a widely used random number routine available from a renowned software source. The numbers produced by this algorithm had passed a battery of sophisticated tests with random colors flying. Yet when the researcher selected alternative output samples from the “random source” to determine abscissae and ordinates of white dots on a TV screen, the result was not the expected random “snow” but just a few diagonal streaks!

In the following we will sketch some fundamental facts about both random and pseudorandom number generators. We will be interested in both discrete and (quasi-)continuous distributions of random numbers. (Remember that nothing is really continuous in digital computation.) Our main tools will turn out to be congruences and recursions.

¹ The reader may recall that many analytically intractable problems can be solved (albeit with a statistical error) by simulation with a random process and “tabulating” the results of one or more simulation runs.
27.1 Pseudorandom Galois Sequences

In Chap. 25 we became acquainted with a method of generating pseudorandom sequences with elements from $\text{GF}(p)$ and period length $p^m - 1$ [27.1]. For $p = 2$, we obtain binary-valued sequences [27.2] with elements 0 and 1 or, as often preferred, 1 and $-1$.

These sequences $a_n$ are generated by linear recursions [27.3]:

$$a_n = \langle f_1a_{n-1} + \ldots + f_ma_{n-m} \rangle_p$$

where addition is modulo $p$, as indicated by the acute brackets, which signify least positive (or nonnegative) remainders. The coefficients $f_k$ are determined by primitive polynomials in $\text{GF}(p^m)$ (Chap. 26).

These sequences are, of course, not truly random. Each period contains precisely all $m$-tuples but one (the all-zero tuple). Also, their correlation is not that of a truly random sequence (like, say, the pulses from a Geiger counter near a radioactive source). In fact, the periodic autocorrelation has only two possible values (a highly valued property in numerous applications; see Chap. 26). Nevertheless, by choosing $m$ very large, say $m = 168$, and selecting a relatively small excerpt from a full period, pseudorandomness can approach true randomness under many statistical tests. For $m = 168$ and $p = 2$, for example, the period length exceeds $10^{50}$ samples, and even as many as $10^{10}$ samples from such a sequence constitute a very small portion of the entire sequence.

If used in application where higher-order correlations are important, however, trouble can still occur. As an illustration, for $m = 5$ and $p = 2$, and the recursion

$$a_n = \langle a_{n-4} + a_{n-5} \rangle_2$$

or, for $b_n = 1 - 2a_n$,

$$b_n = b_{n-4} \cdot b_{n-5}$$

the third-order correlation coefficient becomes

$$c_{4,5} = \sum b_n \cdot b_{n-4} \cdot b_{n-5} = \sum 1$$

Thus, $c_{4,5}$, instead of being appropriately randomly small, becomes catastrophically large.\(^2\)

The problem with higher-order correlations can be somewhat alleviated by basing the recursion on primitive polynomials with a maximum number of terms. In our example ($m = 5$, $p = 2$), this would mean replacing (27.3) by a recursion involving four terms instead of only two:

$$b_n = b_{n-1} \cdot b_{n-2} \cdot b_{n-3} \cdot b_{n-5}$$

\(^2\)Overlooking the importance of higher-order correlation in analyzing nonlinear systems has led to some very misleading results in the animal neurophysiology of hearing when binary pseudorandom sequences of the kind described here have been used as an acoustic input to the animal's ear.