A Redesigned C++ Library to Test the Lattice Structure of Linear Generators and Search for Good Ones

Pierre L'Ecuyer
DIRO, Université de Montréal, Canada
lecuyer@iro.umontreal.ca

Coauthor(s): Christian F. Weiss

Special session: Testing and analysis of pseudo-random number generators

The spectral test, introduced in [1] and popularized by [2], remains the gold standard to measure the uniformity of point sets produced by linear random number generators by assessing the quality of their lattice structure. Multiple recursive generators, multiply-with-carry, matrix linear congruential generators, and combined generators of these types, for example, can be constructed and analyzed by this type of test [3, 4, 5]. A software tool named LatMRG was written about 30 years ago in the Modula-2 language to perform the spectral test and search for generators with a good lattice structure [4], but this tool can no longer be used because Modula-2 is no longer supported. We are aware of no other similar tool currently available.

In this talk, we present a completely redesigned version of LatMRG, written in C++, and using NTL to handle computations with large numbers. Some of the underlying algorithms have been improved compared with the Modula-2 version. We illustrate what the software can do and its performance via several examples.

[1] R. R. Coveyou and R. D. MacPherson. Fourier analysis of uniform random number generators. *Journal of the ACM*, 14:100–119, 1967.

[2] D. E. Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms.* Addison-Wesley, Reading, MA, second edition, 1981.

[3] P. L'Ecuyer. Good parameters and implementations for combined multiple recursive random number generators. *Operations Research*, 47(1):159–164, 1999.

[4] P. L'Ecuyer and R. Couture. An implementation of the lattice and spectral tests for multiple recursive linear random number generators. *INFORMS Journal on Computing*, 9(2):206–217, 1997.

[5] P. L'Ecuyer, P. Wambergue, and E. Bourceret. Spectral analysis of the MIXMAX random number generators. *INFORMS Journal on Computing*, 32(1):135–144, 2020.