Chapter 25
Uniform Random Number Generation With Jumping Facilities

E. Jack Chen
BASF Corporation, USA

ABSTRACT

A facility for generating sequences of pseudorandom numbers is a fundamental part of computer simulation systems. Furthermore, multiple independent streams of random numbers are often required in simulation studies, for instance, to facilitate synchronization for variance-reduction purposes, and for making independent replications. A portable set of software utilities is described for uniform random-number generation. It provides for multiple generators (streams) running simultaneously, and each generator (stream) has its sequence of numbers partitioned into many long disjoint contiguous substreams. Simple procedure calls allow the user to make any generator "jump" ahead/back v steps (random numbers). Implementation issues are discussed. An efficient and portable code is also provided to implement the package. The basic underlying generator CMRG (combined multiple recursive generator) combines two multiple recursive random number generators with a period length of approximately 2^{191} ($\approx 3.1\times 10^{57}$), good speed, and excellent theoretical properties.

INTRODUCTION

As computer capacities and simulation technologies advance, simulation has become the method of choice for modeling and analysis. The fundamental advantage of simulation is that it can tolerate far less restrictive modeling assumptions, leading to an underlying model that is more reflective of reality and thus more valid, leading to better decisions (Lucas et al. 2015). Simulation studies are typically proceed by transforming in a more or less complicated way of a sequence of numbers between 0 and 1 produced by a pseudorandom generator into an observations of the measure of interest. A facility for generating sequences of pseudorandom numbers is a fundamental part of computer simulation systems. Furthermore, random number generators also play an important role in cryptography. A collection of random

DOI: 10.4018/978-1-5225-7368-5.ch025

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

variables $x_1, x_2, \ldots, x_n$ is a random sample if they are independent and identically distributed. True random numbers cannot be produced by a deterministic algorithm, and hence, random numbers generated by using a recursive equation are referred to as pseudorandom numbers. The deterministic nature of these techniques is important because it can be reproduced in computations. A facility for generating sequences of pseudorandom numbers is a fundamental part of computer simulation systems. Usually, in practice, such a facility produces a deterministic sequence of values, but externally these values should appear to be drawn independently from a uniform distribution between 0 and 1, i.e., they are independent and statistically indistinguishable from a truly random sequence. Furthermore, multiple independent streams of random numbers are often required in simulation studies, for instance, to facilitate synchronization for variance-reduction purposes, and for making independent replications.

A random number generator (RNG) is an algorithm that starting from an initial seed (or seeds), produces a stream of numbers that behaves as if it were a random sample when analyzed using statistical tests. The RNG is closely related to the Deterministic Random Bit Generators (DRBGs). See L'Ecuyer (1990, 2013) and references therein for more information on RNGs. We describe a portable set of software utilities for uniform random-number generation. It provides for multiple generators (streams) running simultaneously, and each generator (stream) has its sequence of numbers partitioned into many long disjoint contiguous substreams, see L'Ecuyer et al. (2002). Simple procedure calls allow the user to make any generator "jump" ahead/back v steps (random numbers). Implementation issues are discussed. The basic underlying generator CMRG (Combined Multiple Recursive Generator) combines two multiple recursive random number generators with a period length of approximately 2^{191} ($\approx 3.1\times 10^{57}$), good speed, and excellent theoretical properties, e.g., the lattice structure, see Kroese et al. (2011) for a list of desired properties.

BACKGROUND

There are a number of methods for generating the random numbers, of which the most popular are the congruential methods (mixed, multiplicative, and additive). The (mixed) linear congruential generators (LCGs) are defined by

$$x_i = \left(ax_{i-1} + c\right) MOD\ m, \quad u_i = \frac{x_i}{m}, x_0 \in \{1,\ldots,m-1\}\ i > 0.$$

where m (the modulus) is a positive integer (usually a very large primary number), a (the multiplier) $\in \{1,\ldots,m-1\}$ and c (the increment) is a nonnegative integer. This mathematical notation signifies that x_i is the remainder of $(ax_{i-1}+c)$ divided by m. Hence, $x_i \in \{1,\ldots,m-1\}$. Thus, random variable u_i is a uniform 0, 1 variable. Note that

$$x_{i+v} = \left(a^v x_i + \frac{c\left(a^v - 1\right)}{a - 1}\right) \mathrm{MOD}\ m\,.$$

11 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/uniform-random-number-generation-with-jumping-facilities/213139

Related Content

Shapes and Patterns in Visualizing Human Knowledge: Part 1: Origin From a Historical and Cognitive Perspective
Grzegorz Osinskiand Veslava Osinska (2018). *Information Visualization Techniques in the Social Sciences and Humanities (pp. 1-16).*
www.irma-international.org/chapter/shapes-and-patterns-in-visualizing-human-knowledge/201299

A Qualitative Case Study for Technology Acceptance Using TAM and the Kübler-Ross Models
Benjamin Soteloand Richard Alan Livingood (2018). *Technology Adoption and Social Issues: Concepts, Methodologies, Tools, and Applications (pp. 664-674).*
www.irma-international.org/chapter/a-qualitative-case-study-for-technology-acceptance-using-tam-and-the-kbler-ross-models/196698

Adding Emotions to Models in a Viewpoint Modelling Framework From Agent-Oriented Software Engineering: A Case Study With Emergency Alarms
Leon Sterling, Alex Lopez-Lorcaand Maheswaree Kissoon-Curumsing (2018). *Innovative Methods, User-Friendly Tools, Coding, and Design Approaches in People-Oriented Programming (pp. 324-367).*
www.irma-international.org/chapter/adding-emotions-to-models-in-a-viewpoint-modelling-framework-from-agent-oriented-software-engineering/203849

Anti-Models for Architectural Graphic Expression and UX Education
Francisco V. Cipolla-Ficarra, Jim Carré, Alejandra Quirogaand Valeria M. Ficarra (2018). *Technology-Enhanced Human Interaction in Modern Society (pp. 218-233).*
www.irma-international.org/chapter/anti-models-for-architectural-graphic-expression-and-ux-education/189845

Developing and Measuring the Business Case for Health Information Technology
Duncan Wade Unwin, Louis Sanzogniand Kuldeep Sandhu (2018). *Technology Adoption and Social Issues: Concepts, Methodologies, Tools, and Applications (pp. 1628-1656).*
www.irma-international.org/chapter/developing-and-measuring-the-business-case-for-health-information-technology/196748