Stochastic Empirical Loading and Dilution Model (SELDM) Version 1.0.0—Appendix 1. Monte Carlo Methods

By Gregory E. Granato

Prepared in cooperation with the
U.S. Department of Transportation, Federal Highway Administration,
Office of Project Development and Environmental Review

Techniques and Methods 4–C3

U.S. Department of the Interior
U.S. Geological Survey

U.S. Department of the Interior
KEN SALAZAR, Secretary

U.S. Geological Survey
Suzette M. Kimball, Acting Director

U.S. Geological Survey, Reston, Virginia: 2013

For more information on the USGS—the Federal source for science about the Earth, its natural and living resources, natural hazards, and the environment, visit http://www.usgs.gov or call 1–888–ASK–USGS.

For an overview of USGS information products, including maps, imagery, and publications, visit http://www.usgs.gov/pubprod

To order this and other USGS information products, visit http://store.usgs.gov

Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

Suggested citation:
Granato, G.E., 2013, Stochastic empirical loading and dilution model (SELDM) Version 1.0.0—Appendix 1, Monte Carlo methods: U.S. Geological Survey Techniques and Methods, book 4, chap. C3, 26 p., CD–ROM. (Also available at http://pubs.usgs.gov/tm/04/c03/.)

Contents

Figures

Tables

Abbreviations

BMP	best management practice
CMRRNG	combined multiple recursive random-number generator
CDF	cumulative distribution function
INT	integer function
K_N	normal variate, a pseudorandom decimal value that is generated to represent a random variable with a normal distribution
K_P	Pearson type III variate, a pseudorandom decimal value that is generated to represent a random variable with a Pearson type III distribution
ln	natural logarithm
LCG	linear congruential generator
MRRNG	multiple recursive random-number generator
PC	personal computer
PPCC	probability-plot correlation coefficient
PRNG	pseudorandom-number generator
RAM	random access memory
SELDM	Stochastic Empirical Loading and Dilution Model
U_{01}	uniform random variate, a pseudorandom decimal value that is generated to represent a random variable with a uniform distribution between 0 and 1
VB	Microsoft Visual Basic®
VBA	Microsoft Visual Basic for Applications®

Stochastic Empirical Loading and Dilution Model (SELDM) Version 1.0.0—Appendix 1. Monte Carlo Methods

By Gregory E. Granato

Introduction

The Stochastic Empirical Loading and Dilution Model (SELDM) uses Monte Carlo methods to generate large stochastic datasets for the variables needed to provide planning-level estimates of runoff quality at a site of interest. The results of the stochastic analysis indicate the risk for exceeding water-quality objectives and therefore the potential need for mitigation measures to reduce effects of runoff on receiving waters. In SELDM, Monte Carlo methods are used with input statistics to generate multiyear samples of precipitation, prestorm streamflow, runoff coefficients, water-quality data, and metrics for the performance of best management practices (BMPs). Monte Carlo methods are used because the interplay of these variables is complex. Theoretical and analytical solutions cannot characterize the potential combinations of input variables because each variable may be best characterized by a different probability distribution.

The purpose of this appendix is to document the numerical methods that are implemented in SELDM with references to literature that describes the theory and development of such methods. This appendix provides the information and equations needed to generate stochastic data for each statistical distribution used in SELDM. This appendix documents methods for adjusting these data to represent correlations between variables by using rank correlation coefficients or regression equations. Methods for adjusting data to represent conditional probabilities also are documented. Uses for different methods are described as examples in this appendix, but the applications of each method for individual hydrologic variables in the model are described in the main body of this report.

Generating Random Numbers from a Uniform Distribution

SELDM uses a pseudorandom-number generator (PRNG) to generate seemingly random numbers that simulate a uniform distribution. Computer-based random-number generators are identified as PRNGs because computers are, by design, precise and deterministic calculators that cannot produce a set of truly random numbers without an external random signal (Press and others, 1992; L'Ecuyer, 1999; Saucier, 2000; Gentle, 2003). A variate is commonly defined as a single pseudorandom number that is generated to represent a random variable (Devroye, 1986). A series of variates generated by a PRNG represents a sample of values from a specified probability distribution. All values within the specified range of a uniform distribution are equally likely to occur. Hydrologic processes do not commonly produce values that fit a uniform distribution (Haan, 1977; Chow and others, 1988; Stedinger and others, 1993), but the other distributions of interest for modeling hydrologic data are generated by using one or more variates generated from a uniform distribution (Devroye, 1986; Press and others, 1992; Salas, 1993; Saucier, 2000; Gentle, 2003; L'Ecuyer and Simard, 2007).

A high-quality PRNG is needed for defensible stochastic simulation models. There are, however, persistent concerns about the quality of many available PRNGs (Press and others, 1992; Hellekalek, 1998; Saucier, 2000; Gentle, 2003; McCullough and Wilson, 2005; L'Ecuyer and Simard, 2007; McCullough, 2008). SELDM was developed as a Microsoft Access® database software application by using Visual Basic for Applications® (VBA). Research shows that the PRNGs native to Microsoft Visual Basic® (VB) and VBA may not meet criteria for high-quality PRNGs (McCullough and Wilson, 2005; L'Ecuyer and Simard, 2007; McCullough, 2008). Thus, a high-quality PRNG known as MRG32k3a (L'Ecuyer, 1999) was implemented in VBA to generate variates from a uniform distribution for use with SELDM.

A high-quality PRNG is commonly defined as a generator that passes a number of numerical tests and is found suitable for intended uses (L'Ecuyer, 1988; Press and others, 1992; Hellekalek, 1998; L'Ecuyer, 1998; L'Ecuyer, 1999; Gentle, 2003; L'Ecuyer and Simard, 2007). Statistical theory dictates that random numbers are independent and identically distributed, but PRNGs have several characteristics that are at odds with statistical theory. PRNGs are deterministic; given a starting position, they will produce the same string of output values with each value being a function of the previous value(s). For this reason, subsequent values are not independent. Paradoxically, however, this characteristic of PRNGs also is advantageous for stochastic simulation models because the implementation of the generator can be verified, and stochastic simulations can be repeated by using the same starting values. A true uniform random-number set will, in theory, completely fill the interval from 0 to 1. A PRNG, however, cannot produce every possible value in the interval. The mathematical structure of a PRNG causes the series of numbers to fall on lattice planes (L'Ecuyer, 1988; L'Ecuyer, 1998; Gentle, 2003). High-quality PRNGs, however, are designed so that the lattice planes are of such a high order that serial correlations within the series are not readily detected. The distances between lattice planes also are small enough so that they are effectively zero. Output values from a high-quality PRNG should be almost indistinguishable from the theoretical distribution. As such, the output from a high quality PRNG will pass goodness-of-fit tests such as the Anderson-Darling, chi-square, and Kolmogorov-Smirnov tests (L'Ecuyer and Simard, 2007). PRNGs have a set period equal to the length of the sequence of output values that are generated before the PRNG returns to the initial value. Once a PRNG progresses through its period, it repeats the same string of output values. Thus the period of a high-quality PRNG should be several orders of magnitude greater than the longest series needed for an analysis. The length of the period is especially important if a Monte Carlo analysis includes more than one variable so that the string of pseudorandom numbers for one variable does not substantially overlap the string for another variable.

The MRG32k3a Multiple Recursive Pseudorandom-Number Generator

Many PRNGs are based on one or more linear congruential generators (Press and others, 1992; L'Ecuyer, 1999; Saucier, 2000; Gentle, 2003). These PRNGs produce seemingly random numbers by using the integer remainder from the division of two numbers. A linear congruential generator (LCG) is implemented by using the equation

$$X_{(i+1)} = \left(a \times X_{(i)} + c\right) \bmod m\,, \qquad (1\text{–}1)$$

where

$X_{(i+1)}$ is an integer value produced by the current iteration i of the LCG;

$X_{(i)}$ is an integer value input to the current iteration i of the LCG—the first value of $X_{(i)}$ that is used to start a random sequence is commonly referred to as the random seed;

a is the constant multiplier, which should be a large prime-integer value;

c is the constant increment, which must be an integer; and

m is the constant modulus, which should be a large prime-integer value.

A multiple recursive random-number generator (MRRNG) uses more than one previous value of $X_{(i)}$ to generate each subsequent seed value of $X_{(i+1)}$. L'Ecuyer (1999) selected a second-order generator of the form

$$X_{(j)} = \left(a_1 \times X_{(j-1)} + a_2 \times X_{(j-2)} + c\right) \bmod m\,, \qquad (1\text{–}2)$$

with the X, a, c, and m values having the same definitions as in equation 1–1, and j the same definition as i in equation 1–1. The integer values of X produced by equation 1–1 or 1–2 are converted to uniform random variates between 0 and 1 (U_{01}) by using equation 1–3, which is

$$U_{01} = \frac{X}{m}\,. \qquad (1\text{–}3)$$

Good values of a, c, and m are critical because they define the spacing of values, the apparent randomness of the series, and the correlations among successive values of U_{01} generated in the series (Press and others, 1992; Hellekalek, 1998; L'Ecuyer, 1999; Gentle, 2003). Selecting a good value of m is critical because this value controls the period of the generator and the extent to which the generator fills the interval from 0 to 1.

MRG32k3a (L'Ecuyer, 1999) was selected for use in SELDM because it passed several batteries of tests commonly used to identify high-quality PRNGs (Marsaglia and Tsang, 2002; L'Ecuyer and Simard, 2007, 2009). MRG32k3a is a combined multiple recursive random-number generator (CMRRNG), which uses two instances of equation 1–2 with different values of a, c and m. If X_j and Y_j are the numbers produced by the CMRRNG with two instances of equation 1–2, then the combined generator would have the form

$$U_{01} = \frac{(X_j - Y_j)}{(m_x + 1)} \tag{1–4}$$

if X_j is greater than Y_j and

$$U_{01} = \frac{(X_j - Y_j + m_x)}{(m_x + 1)} \tag{1–5}$$

if X_j is less than or equal to Y_j (L'Ecuyer, 1999). Using equations 1–4 and 1–5 with m_x+ 1 as the modulus of the combined generator ensures that the uniform numbers produced fall in the range between 0 and 1. The parameters, *a*, *c*, and *m* for this generator are listed in table 1–1.

Table 1–1. Parameters for the MRG32k3a combined multiple recursive random-number generator (CMRRNG) by L'Ecuyer (1999).

[Variables *a*, *c*, and *m* shown in equation 1–2]

Generator	Multiplier a_1	Multiplier a_2	Increment *c*	Modulus *(m)*	
				Real number	Base 2 expression
LCG1	1,403,580	810,728	0	4,294,967,087	2^{32}–209
LCG2	527,612	1,370,589	0	4,294,944,443	2^{32}–22,853

L'Ecuyer (1999) developed several criteria for selecting parameters that would result in a high-quality PRNG that could be implemented on a 32-bit computer and conducted a 20–40 hour random search for each set of constraints. The MRG32k3a CMRRNG had the best properties from among the 32-bit PRNGs that were tested. The period of this generator is 2^{191} (about 3.14×10^{57}), which means that randomly selected seed values have a small chance of producing an overlapping series of output values. The large divisor also means that the minimum distance between two uniform random numbers is about 2.3×10^{-11}, which is a good approximation for a continuous random variable. L'Ecuyer (1999) also checked the lattice structure for this generator and described it as a "well behaved" lattice structure of 45 dimensions.

Although PRNGs are theoretically built on integer operations, MRG32k3a uses double-precision floating-point (real) numbers. The largest (long) integer that can be represented on a 32-bit processor is 2,147,483,647, which is 2^{31}-1. The double-precision floating-point data type, however, can represent numbers on the order of 1.79×10^{308}, which is almost $2^{1,024}$. The parameters, *a*, *c*, and *m* also are selected so that values for X_j and Y_j in equation 1–2 would maintain integer values, even though floating-point variables and operators are used to generate uniform random numbers with the MRG32k3a algorithm (L'Ecuyer, 1999).

Random-Seed Management in SELDM

A random-seed management algorithm was developed for SELDM. This algorithm was designed to generate reproducible results, to facilitate sensitivity analysis, and to preclude manual seed selections that could be used to strategically affect simulation results. The initial random-seed values determine the series of random numbers that are produced for each runoff-quality analysis. The random-seed management algorithm uses a master seed that is generated for an analysis, and the random-number generator selects seed values for each simulated parameter from a random-seed lookup table. The management algorithm also was designed so that the relatively small stochastic datasets generated for SELDM runs would not have uniform random variates that substantially depart from theoretical U_{01} values (an average of 0.5, a standard deviation equal to the square root of 1/12, (about 0.288675), and a coefficient of skew equal to 0). In many Monte Carlo applications, a single parameter is run hundreds or thousands of times to generate many realizations to simulate multiple samplings of an underlying population. In such cases, random-seed management is undesirable because it is the variation in sample statistics or outcomes for one variable that is of interest. In SELDM, however, Monte Carlo methods are used to simulate one realization that results from a large number of random storm events. SELDM is designed to evaluate the effects of different combinations of environmental variables based on statistics input by the user. Each simulation in SELDM produces stochastic data for about 1,000 to 2,000 runoff events. This number of events is about 1 to 2 orders of magnitude larger than the number of monitored events in the highway-runoff datasets collected for each site in the database (Granato and Cazenas, 2009). Each simulation, however, is only one stochastic realization for the site of interest. The U_{01} statistics are controlled within tight tolerances so that the output values will represent hydrologic variability rather than statistical sampling variability. If the input statistics are assumed to be representative, each simulation should represent the permutations and combinations of the variables being simulated.

SELDM can be run multiple times with different master seeds to assess the potential for statistical sampling variability. The user can reselect the master seed and rerun the analysis or copy the analysis, reselect the master seed, and rerun the analysis. Copying the analysis and resetting the master-seed values will allow the user to document a stochastic sensitivity analysis. Copying the analysis, keeping the master seed, and varying input specifications will allow the user to document a hydrologic-sensitivity analysis.

SELDM uses a random-seed table (tblURNSeeds) with 11,003 random-seed pairs that are used as the initial condition for the MRG32k3a algorithm. These random-seed pairs were selected to ensure that the MRG32k3a algorithm would produce unbiased samples for datasets with 500 to 3,000 variates. The seeds were selected so that the average and standard deviation of the samples of U_{01} values would be within the range of about 0.5 ± 0.00535 and 0.288675 ± 0.011545, respectively. Selecting random-seed values that would produce a set of U_{01} values within these tight tolerance limits was a two-step process that consisted of using a VB program (SeedMaker.exe) to generate seeds and a Microsoft Access® database (RandomSeeds.mdb) to identify values that met the user-defined criteria. The SeedMaker.exe program is available on the computer media accompanying this report. The RandomSeeds.mdb database is available as a compressed file (RandomSeedsMDB.zip) on the SELDM Web site (http:/ma.water.usgs.gov/FHWA/SELDM.htm).

SeedMaker.exe was written to generate U_{01} values, calculate statistics, and output each random-seed pair with the maximum deviation from theoretical U_{01} statistics. Each time the program was run, it initialized the intrinsic VB 6.0 random-number generator to select one random-seed pair as the starting position for the MRG32k3a algorithm. The program used one subroutine with the MRG32k3a algorithm to generate the user-defined number of random-seed pairs from the initial starting position. For each pair, the program used a second subroutine with an independent copy of the MRG32k3a algorithm to generate up to 3,000 U_{01} values and to calculate the cumulative average, standard deviation, and skew as each new value was generated. The largest deviation of each statistic was recorded as the U_{01} values were generated and saved with the value of the starting seed pair. Several copies of the SeedMaker.exe program were used to generate 13,448,833 random-seed pairs; this process took about 660 hours of processing time on five personal computers (PCs) with 2-gigahertz processors and at least 1 gigabyte of random access memory (RAM).

The output files from the SeedMaker.exe program were imported into the RandomSeeds.mdb database. Queries to the compiled input table indicated that, because each of the 13,448,833 random-seed pairs was unique, the random-search algorithm was successfully implemented. The RandomSeeds.mdb database was queried to select random-seed pairs that would produce stochastic samples of U_{01} values with maximum difference of plus-or-minus 1.7 percent of the average and plus-or-minus 4 percent of the standard deviation as sample size increased from 500 to 3,000 uniform random variates. Inspection of the random-seed statistics revealed that controlling the maximum difference from the theoretical average and standard deviation resulted in coefficients of skew that were within the range of plus or minus 0.183, which is sufficient for use with SELDM. The application of these statistical criteria yielded the 11,003 random-seed pairs that are used in SELDM.

SELDM calculates a series of seeds from the master seed value and uses the series to generate each stochastic sample. The seed values are randomly selected from a series of U_{01} values, are rescaled to the range of index values (from 1 to 11,003) in table tblURNSeeds, and are converted to the integer values equal to the index numbers identifying the random-seed pairs. The following equation (Saucier, 2000)

$$U_{\min-\max} = i_{\min} + \left(\left[i_{\max} - i_{\min}\right] \times U_{01}\right) \tag{1–6}$$

is used to rescale a U_{01} value to any range, where

U_{01} is the uniform random variate, which is a decimal value between 0 and 1;

$i_{\min}$ is the lower bound of the generated variates, which was set equal to 1 in SELDM;

$i_{\max}$ is the upper bound of the generated variates, which was set equal to 11,003 in SELDM; and

$U_{\min-\max}$ is the uniform random variate in the specified range.

The double precision real number $U_{\min-\max}$ is truncated to an integer by the VBA integer function (INT).

A random-seed-pair population of 11,003 values was considered to be sufficient for use in SELDM because each simulation requires relatively few pairs. There are 7 random-number samples used to simulate precipitation, prestorm flow, and runoff and 2 random-number samples used to simulate the hydraulic BMP treatment variables. There are 1 to 4 random-number samples used to simulate each water-quality constituent. If all 116 constituents in the highway-runoff database (Granato and Cazenas, 2009) were included in a SELDM analysis four times (if different datasets are tested), this analysis would use about 1,865 seed values (about 17 percent of the initial seed values in the database). According to Bayes' theorem, the number of ways 1,865 values could be drawn from 11,003 values without replacement is almost infinite (Haan, 1977). The factorial values for this calculation are beyond the ability to calculate on a 32-bit computer.

Generating Random Numbers by Using the Inverse Cumulative Distribution Function

The inverse cumulative distribution function (CDF) method (also known as the inverse transformation method) is a simple, efficient technique for generating random numbers from a specified probability distribution by using a set of U_{01} random numbers (Press and others, 1992; Saucier, 2000; Gentle, 2003; Cheng and others, 2007). The inverse CDF method is based upon the property that a random variable X has a CDF $F_x(\ )$, and that substituting the values of X will yield uniform random numbers in the range between 0 and 1 ($F_x(X) = U_{01}$). Thus, the inverse CDF can be used to generate values of X from U_{01} values ($F_x^{-1}(U_{01}) = X$). The inverse CDF method is simple and efficient, but it cannot be applied to all distributions because an analytical solution is needed to invert the CDF for generating the X values. In SELDM, the inverse CDF method is used to generate stochastic data from the exponential distribution, the trapezoidal distribution, and the triangular distribution, which is a special case of the trapezoidal distribution.

The One- or Two-Parameter Exponential Distribution

SELDM generates one- and two-parameter exponential variates from the uniform variates by using the inverse transform method with the theoretical inverse CDF of the exponential distribution. A theoretical function can be used for the exponential distribution because there is an analytical solution for this function (Devroye, 1986; Press and others, 1992; Saucier, 2000; Gentle, 2003). The exponential distribution is commonly fully characterized by the mean value and so is referred to as a one-parameter distribution. In some cases, however, a lower bound that is greater than zero is needed to characterize hydrologic data. For example, if runoff events are defined as having a minimum volume of 0.1 in., a minimum duration of 1 hour, and an interevent time of 6 hours (Driscoll and others, 1989), with each variable being exponentially distributed, then the volume, duration, and interevent time would be modeled as two-parameter exponential distributions. The defined minimum values would be selected as the lower bounds of the distribution. To implement the exponential-distribution algorithm, take the uniform variate U_{01} generated by using the MRG32k3a algorithm (L'Ecuyer, 1999) and calculate the exponential variate

$$X_e = X_{\min} - \left(\bar{X} - X_{\min}\right) \times \ln(1 - U_{01})\,, \tag{1–7}$$

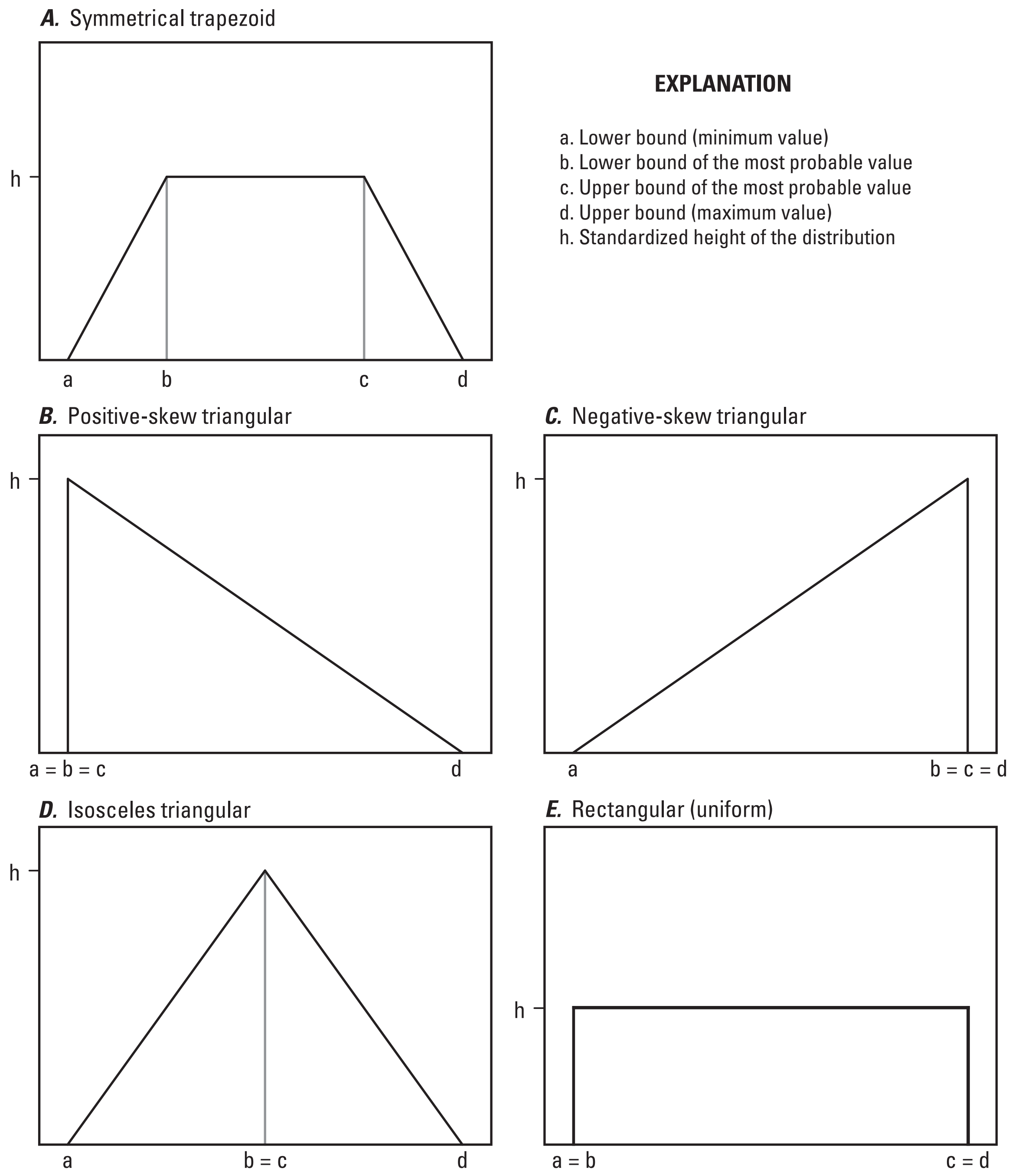
where

X_e is the exponential variate;
$X_{\min}$ is the lower bound of the sample to be generated;
$\bar{X}$ is the average value; and
U_{01} is the uniform random variate, which is a decimal value between 0 and 1.

If $X_{\min}$ is set to zero, the equation reverts to the one-parameter exponential distribution. The natural logarithm (ln) of one minus the uniform variate is used in equation 1–7 so that values of the exponential variates increase with increasing values of the uniform variate; this method is consistent with methods used by SELDM to generate variates from other distributions.

The Trapezoidal Distribution

SELDM generates random numbers that follow trapezoidal distributions by using the inverse CDF with an algorithm developed by Kacker and Lawrence (2007). The trapezoidal distribution is defined by a lower bound (the minimum value), a lower bound of the most probable value, an upper bound of the most probable value, and an upper bound (the maximum value), all of which are shown in figure 1–1. The trapezoidal distribution is very flexible and can assume a variety of shapes, including a positive-skewed triangular distribution, a negative-skewed triangular distribution, a symmetric (isosceles) triangular distribution, and a rectangular (uniform) distribution (fig. 1–1).



A. Symmetrical trapezoid

B. Positive-skew triangular

C. Negative-skew triangular

D. Isosceles triangular

E. Rectangular (uniform)

EXPLANATION
a. Lower bound (minimum value)
b. Lower bound of the most probable value
c. Upper bound of the most probable value
d. Upper bound (maximum value)
h. Standardized height of the distribution
h
a
b
c
d
h
a = b = c
d
h
a
b = c = d
h
a
b = c
d
h
a = b
c = d

Figure 1–1. Five possible probability-density functions of the trapezoidal distribution as defined by the location variables. The height of each trapezoid is calculated to normalize the area under the probability-density function to equal one.

Random numbers from a trapezoidal distribution can be calculated by using the inversion method. First, the standardized height, or maximum probability (*h*), of a standardized trapezoidal distribution is calculated as

$$h = \frac{2}{(d-a)+(c-b)}, \tag{1–8}$$

where

a is the lower bound (minimum value),
b is the lower bound of the most probable value,
c is the upper bound of the most probable value, and
d is the upper bound (maximum value).

To generate random numbers within a trapezoidal distribution (X_T), take uniform variates (U_{01}) generated by using the MRG32k3a algorithm (L'Ecuyer, 1999) and use the algorithm

if

$$0 \le U_{01} \le \frac{h}{2} \times (b-a),$$

then

$$X_T = a + \sqrt{\frac{2(b-a)}{h}} \times \sqrt{U_{01}}; \tag{1–9}$$

else if

$$\frac{h}{2} \times (b-a) \le U_{01} \le 1 - \left(\frac{h}{2} \times (d-c)\right),$$

then

$$X_T = \frac{a+b}{2} + \frac{U_{01}}{h}; \tag{1–10}$$

else if

$$1 - \left(\frac{h}{2} \times (d-c)\right) \le U_{01} \le 1,$$

then

$$X_T = d - \sqrt{\frac{2(d-c)}{h}} \times \sqrt{1-U_{01}} \tag{1–11}$$

to generate values from a trapezoidal distribution (Kacker and Lawrence, 2007).

Currently (2012), the trapezoidal distribution is not commonly used because use of this generalized distribution to model data is a recent development (Kacker and Lawrence, 2007). However, the rectangular (uniform) and triangular distributions, which are special cases of the trapezoidal distribution, are widely used for Monte Carlo analysis (Devroye, 1986; Johnson, 1997; Back and others, 2000; Saucier, 2000; U.S. Environmental Protection Agency, 2001). These distributions are commonly used because they approximate complex distributions and can be parameterized by using expert judgment or by fitting the distribution to data (Johnson, 1997; Back and others, 2000; U.S. Environmental Protection Agency, 2001). In SELDM, the trapezoidal distribution is used to generate data for the ratio of the falling limb to the rising limb of a runoff hydrograph, the BMP flow-reduction ratio, the BMP hydrograph-extension duration, the ratio of outflow to inflow values of constituent concentrations in the BMP, and the adverse-effect ratio for constituent concentrations in receiving waters. Each of these variables is hydrologically complex and is not well quantified in the literature, but can be characterized by using expert judgment to estimate reasonable minimum, maximum, and most probable values.

Generating Random Numbers by Using the Frequency Factor Method

SELDM can generate stochastic data from the normal, lognormal, Pearson type III, and log-Pearson type III distributions by using the frequency factor method to calculate values of a variable from the average and standard deviation of the data:

$$X_d = \bar{X} + S \times K_d, \tag{1–12}$$

where

X_d is a value from distribution d,
$\bar{X}$ is the average value used to generate stochastic data,
S is the standard deviation used to generate stochastic data, and
K_d is the variate associated with the value X_d for the selected distribution (d).

The frequency factor method commonly is used to model a number of distributions with inverse CDFs that are mathematically intractable (Chow, 1954; Haan, 1977; Chow and others, 1988; Stedinger and others, 1993; Cheng and others, 2007). In the frequency factor equation, the variate is the variable that relates the probability of occurrence to the number of standard deviations above or below the mean. The relation between the probability of occurrence and K_d depends on the distribution being modeled. The frequency factor methods used in SELDM are algebraic approximations of the inverse CDF. K_d values can be generated from uniform variates (U_{01}) by using an interpolation table or algebraic approximations for the inverse CDF.

The Normal or Lognormal Distribution

Values from either the normal or lognormal distribution can be calculated by the frequency factor method with normal variates (K_N). In SELDM, values of K_N are generated from U_{01} variates that were previously generated by use of the MRG32k3a algorithm (L'Ecuyer, 1999) by using algorithm AS241 described in the article entitled "The percentage points of the normal distribution" as a polynomial approximation of the inverse CDF of the normal distribution (Wichura, 1988). This polynomial approximation is used because there is no closed-form expression for the CDF of the normal distribution (Haan, 1977; Chow and others, 1988; Stedinger and others, 1993). This approximated inversion method was selected because other methods in common use, such as the sum of uniform variates or the Box-Muller method, can be computationally intensive and can introduce numerical artifacts that affect the quality of the variate population (Devroye, 1986; Press and others, 1992; Salas, 1993; Saucier, 2000; Gentle, 2003). Various polynomials have been used to calculate normal variates from uniform variates (Abramowitz and Stegun, 1964; Chow, 1988; Salas, 1993). Algorithm AS241 was selected for use with SELDM because the equations provide uniform to normal transformations with a maximum error of about 6×10^{-16}.

The first step needed to implement algorithm AS241 (Wichura, 1988) is to calculate an intermediate variable (q) from the U_{01} variate value:

$$q = U_{01} - 0.5. \tag{1–13}$$

If q is approximately zero (in the range between -1×10^{-15} and 1×10^{-15} inclusive), then

$$K_N = 0.0. \tag{1–14}$$

Otherwise, if q is in the range between -0.425 and 0.425, then AS241 calculates a second intermediate variable (R) from q

$$R = 0.425^2 - q^2 \tag{1–15}$$

and calculates K_N

$$K_N = \frac{((((((((A_7 \times R + A_6) \times R + A_5) \times R + A_4) \times R + A_3) \times R + A_2) \times R + A_1) \times R + A_0)}{((((((((B_7 \times R + B_6) \times R + B_5) \times R + B_4) \times R + B_3) \times R + B_2) \times R + B_1) \times R + B_0)}, \tag{1–16}$$

where the variables A_0 through A_7 and B_0 through B_7 are the coefficients of the polynomial listed in table 1–2. Otherwise, if q is less than or equal to -0.425, then AS241 calculates the second intermediate variable (R) from the U_{01} value:

$$R = \sqrt{-LN(U_{01})}\ , \tag{1–17}$$

where ln(U_{01}) is the natural logarithm of U_{01}. Otherwise, if q is greater than or equal to 0.425, then AS241 calculates the second intermediate variable (R) from the U_{01} value:

$$R = \sqrt{-LN(1-U_{01})}\ . \tag{1–18}$$

Then, if R = 5, AS241 adjusts R:

$$R = R - 1.6 \tag{1–19}$$

Table 1–2. Coefficients for the polynomials used to implement algorithm AS241 by Wichura (1988).

[p, cumulative probability of a value in the interval 0 to 1; q = p-0.5; s =1.3887943864964E-11; values are expressed in scientific notation]

Coefficients for q values in the range -0.425 to 0.425 (eq. 1–16)			
A_0	3.38713287279637E+00	B_0	1.00000000000000E+00
A_1	1.33141667891784E+02	B_1	4.23133307016009E+01
A_2	1.97159095030655E+03	B_2	6.87187007492058E+02
A_3	1.37316937655095E+04	B_3	5.39419602142475E+03
A_4	4.59219539315499E+04	B_4	2.12137943015866E+04
A_5	6.72657709270087E+04	B_5	3.93078958000927E+04
A_6	3.34305755835880E+04	B_6	2.87290857357219E+04
A_7	2.50908092873012E+03	B_7	5.22649527885285E+03
Coefficients for q in the ranges from 0.425 to s and -0.425 to -s (eq. 1–20)			
C_0	1.42343711074968E+00	D_0	1.00000000000000E+00
C_1	4.63033784615654E+00	D_1	2.05319162663776E+00
C_2	5.76949722146069E+00	D_2	1.67638483018380E+00
C_3	3.64784832476320E+00	D_3	6.89767334985100E-01
C_4	1.27045825245237E+00	D_4	1.48103976427480E-01
C_5	2.41780725177451E-01	D_5	1.51986665636165E-02
C_6	2.27238449892692E-02	D_6	5.47593808499534E-04
C_7	7.74545014278341E-04	D_7	1.05075007164442E-09
Coefficients for q in the ranges beyond ±s (eq. 1–22)			
E_0	6.65790464350110E+00	F_0	1.00000000000000E+00
E_1	5.46378491116411E+00	F_1	5.99832206555887E-01
E_2	1.78482653991729E+00	F_2	1.36929880922735E-01
E_3	2.96560571828504E-01	F_3	1.48753612908506E-02
E_4	2.65321895265761E-02	F_4	7.86869131145613E-04
E_5	1.24266094738807E-03	F_5	1.84631831751005E-05
E_6	2.71155556874348E-05	F_6	1.42151175831644E-07
E_7	2.01033439929228E-07	F_7	2.04426310338993E-15

and calculates K_N

$$K_N = \frac{(((((((C_7 \times R + C_6) \times R + C_5) \times R + C_4) \times R + C_3) \times R + C_2) \times R + C_1) \times R + C_0)}{(((((((D_7 \times R + D_6) \times R + D_5) \times R + D_4) \times R + D_3) \times R + D_2) \times R + D_1) \times R + D_0)}, \qquad (1\text{–}20)$$

where the variables C_0 through C_7 and D_0 through D_7 are the coefficients of the polynomial listed in table 1–2. Otherwise, if R is greater than 5, AS241 adjusts R:

$$R = R - 5.0 \qquad (1\text{–}21)$$

and calculates K_N

$$K_N = \frac{(((((((E_7 \times R + E_6) \times R + E_5) \times R + E_4) \times R + E_3) \times R + E_2) \times R + E_1) \times R + E_0)}{(((((((F_7 \times R + F_6) \times R + F_5) \times R + F_4) \times R + F_3) \times R + F_2) \times R + F_1) \times R + F_0)}, \qquad (1\text{–}22)$$

where the variables E_0 through E_7 and F_0 through F_7 are the coefficients of the polynomial listed in table 1–2.

Figure 1–2 is an example of a normal variate dataset that was generated by using AS241. In the stochastic sample of 12,000 points shown in the figure, about 85.2 percent of the sample was converted by using equation 1–16, about 13.6 percent of the sample by using equation 1–20, and about 1.2 percent of the sample by using equation 1–22. By chance, there are no values with a uniform variate within the range of $0.5 \pm 1 \times 10^{-15}$ in this sample, so the value from equation 1–14 is shown for reference only. Although equation 1–14 provides a convenient approximation, it is rarely used because of the small chance of obtaining a double-precision uniform variate value from VBA within this small range.

The polynomial approximation simulates K_N values as if they were generated as samples from a standard normal distribution. The standard normal population has a mean equal to zero, a standard deviation equal to one, and a coefficient of skew equal to zero. The values of these statistics approach theoretical values for large samples, but the statistics for smaller samples may substantially diverge from theoretical values. For example, figure 1–3 shows the theoretical statistics of the standard normal distribution, the 95-percent confidence intervals for each statistic, and the running average, standard deviation, and skew of three random samples. For the most part, the average and standard deviation are within the 95-percent confidence limits, and the samples converge toward the theoretical values if the number of values exceeds 1,000 or 2,000. The coefficients of skew of the samples, however, have larger departures from the theoretical value and do not converge as readily. The random sample with a coefficient of skew that reached 2.06 was caused by a far outlier ($K_N = 3.58$) in the eighth generated value followed by three small negative values. The coefficient of skew of this sample remained outside the 95-percent confidence interval for another 32 generated values until another far outlier ($K_N = -4.19$) balanced out the first far outlier.

Once a standard normal variate (K_N) has been generated, this value can be used to generate a value for a stochastic sample of a normal or lognormal variable by using the frequency factor method (equation 1–12) with the average $\bar{X}$ and standard deviation S defined by the user (Chow, 1954; Haan, 1977; Chow and others, 1988; Stedinger and others, 1993). If the numbers are normally distributed, the average and standard deviation used in equation 1–12 are calculated from the raw data. If the numbers are lognormally distributed, the average and standard deviation used in equation 1–12 are calculated from the logarithms of the data, the frequency factor method is applied, and the individual values are retransformed.

The Pearson Type III or Log-Pearson Type III Distribution

Values from either the Pearson type III or log-Pearson type III distribution can be calculated by the frequency factor method with normal variates (K_N) that have been adjusted for nonzero skews in the data or logarithms of data, respectively. In SELDM, the K_N values are generated from the U_{01} values calculated with MRG32k3a (L'Ecuyer, 1999) and AS241 (Wichura, 1988). These K_N values are then adjusted to Pearson type III (K_P) variates by using the modified Wilson-Hilferty transformation algorithm developed by Kirby (1972). The Wilson-Hilferty transformation is an algebraic approximation that converts a normal variate to a Pearson type III/gamma variate (Wilson and Hilferty, 1931). The Wilson-Hilferty transformation can be used because the Pearson type III distribution is a special case of the gamma distribution (Bobee and Ashkar, 1991).

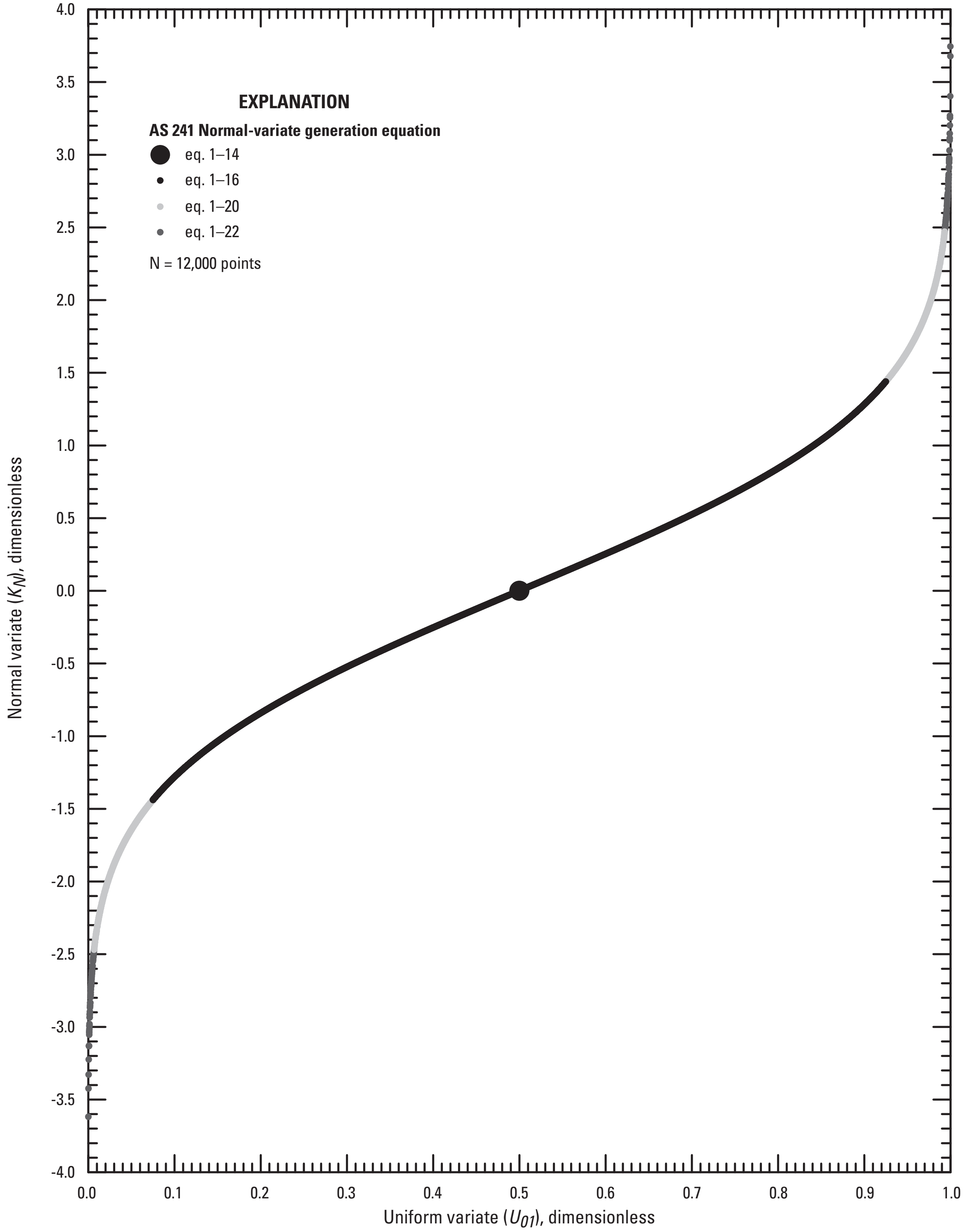


Figure 1–2. Example of implementation of algorithm AS 241 (Wichura, 1988) for producing normal variates from uniform variates for a sample of 12,000 randomly generated points.

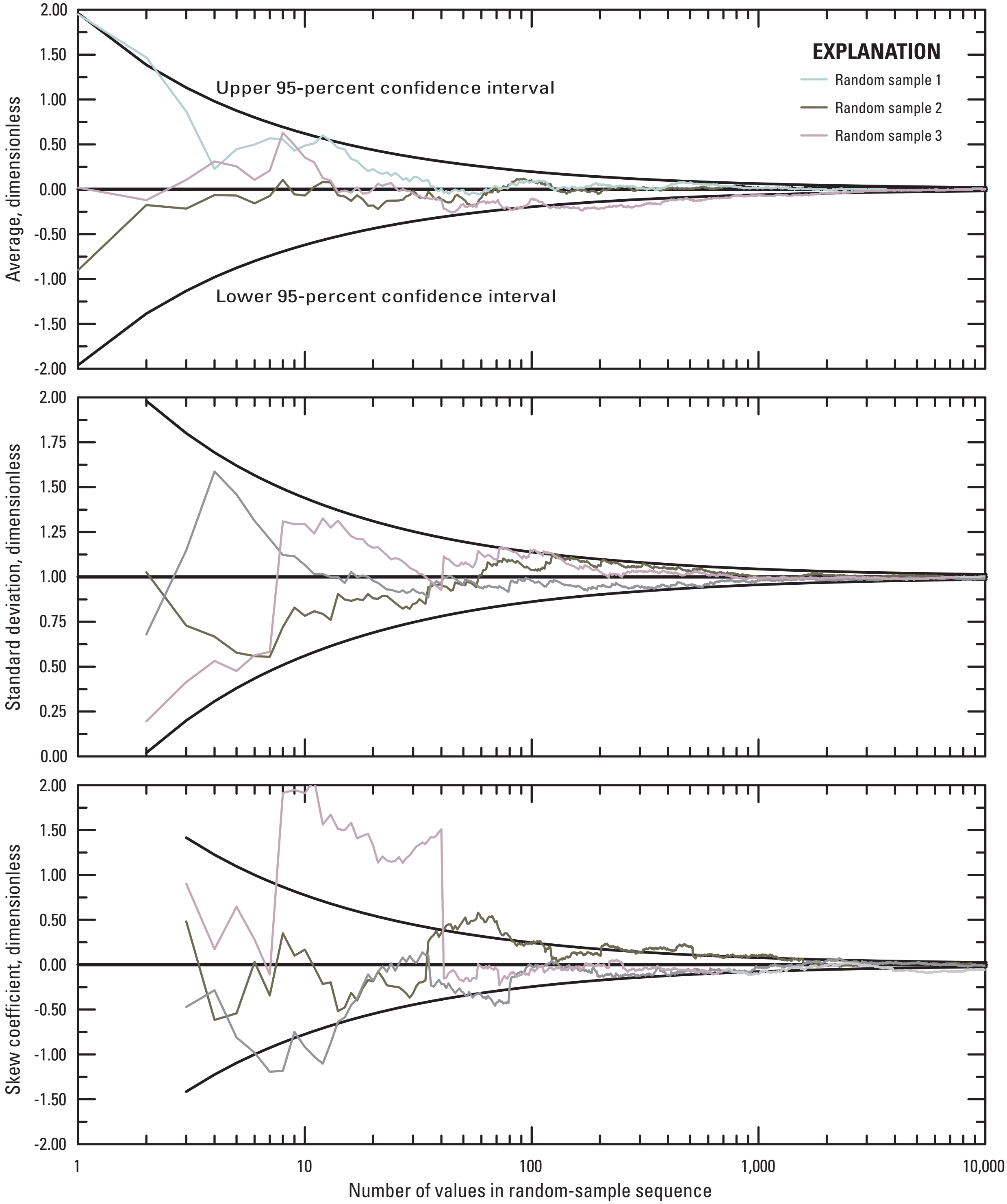


Figure 1–3. Three random samples of the standard normal distribution with the nominal values (average = 0, standard deviation = 1, and coefficient of skew = 0) and the 95-percent confidence intervals for these statistics.

The Wilson-Hilferty transformation is considered to be an accurate approximation if the skew is in the range between -1 and 1 inclusive, and only adequate within the range of skews with absolute values between 1 and 3 (Kirby, 1972; Bobee and Ashkar, 1991):

$$K_{Pw} = \frac{2}{G}\left\{\left[1-\left(\frac{G}{6}\right)^2+\left(\frac{G}{6}\right)K_N\right]^3-1\right\}, \qquad (1\text{–}23)$$

where

K_N is the normal variate,
G is the coefficient of skew, and
K_{Pw} is the Wilson-Hilferty approximation to the Pearson variate (K_P).

The algorithm developed by Kirby (1972) applies adjustment factors to the Wilson-Hilferty transformation to generate variates that approximate theoretical values. Kirby (1972) indicates that these K_P values are “satisfactory throughout the range of hydrologic interest,” which includes skew values in the range of plus or minus 9.75. This algorithm provides variates that preserve the mean, standard deviation, and skew of the standardized Pearson type III distribution. The Kirby (1972) algorithm was selected for use with SELDM because other potential methods are not based on an inverse transform, may require generation of multiple uniform variates for each Pearson type III variate, can be resource intensive, or can be numerically unstable for some values (Haan, 1977; Bobee and Ashkar, 1991; Press and others 1992; Gentle, 2003). Although there are several other approximations and adjustments to the Wilson-Hilferty transformation, the algorithm derived by Kirby (1972) provides the best approximation over the range of skew values that are of hydrologic interest; this range includes skew values that are between -9.75 and +9.75 inclusive (Bobee and Ashkar, 1991). This modified Wilson-Hilferty transformation is

$$K_P = A\left\{\max\left[H, 1-\left(\frac{G_s}{6}\right)^2+\left(\frac{G_s}{6}\right)K_N\right]^3-B\right\}, \qquad (1\text{–}24)$$

where A, B, G_s, and H are intermediate setup variables. Kirby (1972) used an interpolation table to calculate these parameters from the coefficient of skew (G) by using absolute skew values (G_T) that were calculated across intervals of 0.25 (table 1–3).

To use this algorithm, set up the series of intermediate variables based on the input value G for the sample, and calculate a series of K_P values on the basis of input K_N values. The first step to establish the intermediate variables is to calculate the skew interpolation factor (dP):

$$dP = 4\times\left(G_T-|G|\right), \qquad (1\text{–}25)$$

where G_T is the smallest tabled skew that is greater than the absolute value of G. The first variable A, which is the skew adjustment factor, is calculated as

$$A = \max\left(\frac{2}{|G|}, 0.40\right)\times\left((1-dP)\times dA_I + dP\times dA_{(I-1)}\right) \qquad (1\text{–}26)$$

by using values for the interpolation factors dA_I and $dA_{(I-1)}$ for index skew values that bracket the calculated skew (G_I) from table 1–3.

The B term is calculated in two steps. First, an approximate value is calculated, and then a skew adjustment factor is applied. If the absolute value of G is less than or equal to 2.25, the approximate value B_A equals 1; otherwise, it is calculated as

$$B_A = 1+0.0144\times\left(|G|-2.25\right)^2. \qquad (1\text{–}27)$$

Then B is calculated on the basis of the skew interpolation factor (dP) and the values for dB from table 1–3 as

$$B = B_A\times\left((1-dP)\times dB_I + dP\times dB_{(I-1)}\right). \qquad (1\text{–}28)$$

Table 1–3. Interpolation table for the adjusted Wilson-Hilferty approximation by Kirby (1972).

Index number	Absolute value of the coefficient of skew	Interpolation factors		
(I)	*(G_T)*	$dA_{(I)}$	$dB_{(I)}$	$dG_{(I)}$
1	0.00	0.000000	0.000000	0.000000
2	0.25	0.004614	0.000000	-0.000144
3	0.50	0.009159	-0.000001	-0.001137
4	0.75	0.013553	-0.000004	-0.003762
5	1.00	0.017753	-0.000021	-0.008674
6	1.25	0.021764	-0.000075	-0.011555
7	1.50	0.025834	-0.000190	-0.010076
8	1.75	0.030406	-0.000326	-0.006049
9	2.00	0.035710	-0.000317	-0.000921
10	2.25	0.041730	0.000116	0.004189
11	2.50	0.048321	0.000434	0.008515
12	2.75	0.055309	0.000116	0.011584
13	3.00	0.062538	-0.000464	0.013139
14	3.25	0.069873	-0.000981	0.013122
15	3.50	0.077334	-0.001165	0.010945
16	3.75	0.084682	-0.000743	0.007546
17	4.00	0.091926	0.000435	0.002767
18	4.25	0.099028	0.002479	-0.003181
19	4.50	0.105967	0.005462	-0.010089
20	4.75	0.112695	0.009353	-0.017528
21	5.00	0.119245	0.014206	-0.025476
22	5.25	0.106551	0.019964	-0.033609
23	5.50	0.095488	0.026829	-0.042434
24	5.75	0.085671	0.034307	-0.050525
25	6.00	0.076990	0.042495	-0.058192
26	6.25	0.069290	0.051293	-0.065221
27	6.50	0.062443	0.060593	-0.071410
28	6.75	0.056349	0.070324	-0.076638
29	7.00	0.050908	0.080332	-0.080655
30	7.25	0.046047	0.090532	-0.083349
31	7.50	0.041702	0.100831	-0.084584
32	7.75	0.037815	0.111114	-0.084203
33	8.00	0.034339	0.121283	-0.082089
34	8.25	0.031229	0.131245	-0.078126
35	8.50	0.028445	0.140853	-0.072165
36	8.75	0.025964	0.150120	-0.064188
37	9.00	0.023753	0.158901	-0.054059
38	9.25	0.021782	0.167085	-0.041633
39	9.50	0.020043	0.174721	-0.027005
40	9.75	0.018528	0.181994	-0.010188

An intermediate skew variable G_S also is calculated on the basis of the input coefficient of skew (G), tabled skew values (G_T), the skew interpolation factor (dP), and the values for dG from table 1–3. First, the value of G_S is computed as

$$G_S = |G| \times \left(\left(1-dP\right)\times dG_I + dP \times dG_{(I-1)}\right), \tag{1–29}$$

and, if the absolute value of G is greater than one, the value is adjusted as follows:

$$G_S = G_S - 0.063 \times \left(|G|-1\right)^{1.85}. \tag{1–30}$$

The final intermediate variable is H is calculated as

$$H = \left(B - \left(\frac{2}{\left(|G|\times A\right)}\right)\right)^{\frac{1}{3}}. \tag{1–31}$$

A Pearson type III variate (K_P) is calculated for each normal variate (K_N) by using a three-step process. First, the intermediate variable W_{KN} is calculated as

$$W_{KN} = \left(1 - \left(\frac{G_s}{6}\right)^2 + sign(G) \times \left(\frac{G_s}{6}\right) \times K_N\right). \tag{1–32}$$

Second, if W_{KN} is less than H, then this variable is set equal to H. Third, K_P is calculated on the basis of W_{KN} as

$$K_p = A \times \left(W_{KN}^3 - B\right), \tag{1–33}$$

which is the short form of equation 1–24.

The SELDM implementation of the adjusted Wilson-Hilferty algorithm includes three adjustments to improve the results for skews near 0 and skews beyond -9.75 and +9.75. The SELDM algorithm sets the K_P values equal to the K_N values for skews less than 0.005 because the K_N values are a better approximation in this range, and because a skew of 0.005 is within the standard error of a coefficient of skew of 0 for sample sizes smaller than 240,000. The SELDM algorithm uses the unadjusted Wilson-Hilferty equation (equation 1–23) in the ranges between -0.5 and -0.005 and between 0.005 and 0.5 inclusive because the unadjusted equation provides a better approximation in these ranges. The SELDM algorithm truncates skews less than -9.75 and greater than +9.75 to these minimum and maximum values. These extreme skews are truncated because extrapolation errors beyond these values are worse than errors that arise from use of the adjusted Wilson-Hilferty algorithm with truncated skew values. Skew values that approach the limits of -9.75 or +9.75 are very uncommon; datasets with extremely high skew values should be evaluated to assess data-quality concerns.

The results of this approach are demonstrated in figure 1–4, which includes a normal variate (K_N) and Pearson type III (K_P) variates derived for six different skew values. These variates are plotted against the original uniform variates that are rescaled to percentiles and plotted on a probability axis. The probability axis linearizes the normal variate to better show the effects of nonzero skews on the population of the standard distribution. As skews increase from zero, the distributions become increasingly concave up. Values at both ends of the distributions are higher, and values in the center of the distributions are lower, for the Pearson type III variates in comparison to the normal variates for the same values of the uniform variate (fig. 1–4). As skews decrease from zero, the distributions become increasingly concave down. Values at both ends of the distributions are lower and values in the center of the distributions are higher for the Pearson type III variates in comparison to the normal variates for the same values of the uniform variate (fig. 1–4). Pearson type III variates reach asymptotic extreme values once W_{KN} is less than H in the range of uniform variates from 0.0001 to 0.9999 (standard normal variates between -3.719 and +3.719) for positive and negative skew values beyond an absolute value of 1.3. Such asymptotic values are apparent in figure 1–4 for the curves representing Pearson type III variates with skew values of -2 and +2. The asymptotic K_P values of 1 and +1 are reached at the U_{01} values of about 0.025 (the 2.5th percentile) and about 0.975 (the 97.5th percentile) for skew values of 2 and -2, respectively.

Random numbers can be generated by using Pearson type III variates in equation 1–12 to represent a wide range of distributions commonly used in hydrologic studies (Kirby, 1972; Haan, 1977; Chow and others, 1988; Bobee and Ashkar, 1991). The arithmetic or logarithmic mean and standard deviation provide the location and scale of the output CDF, respectively. The Pearson type III variates control the shape of the CDF. For example, if the skew is 0, the variates will fit a normal (or log-normal) distribution; if the skew is equal to 2, the variates will fit an exponential distribution.

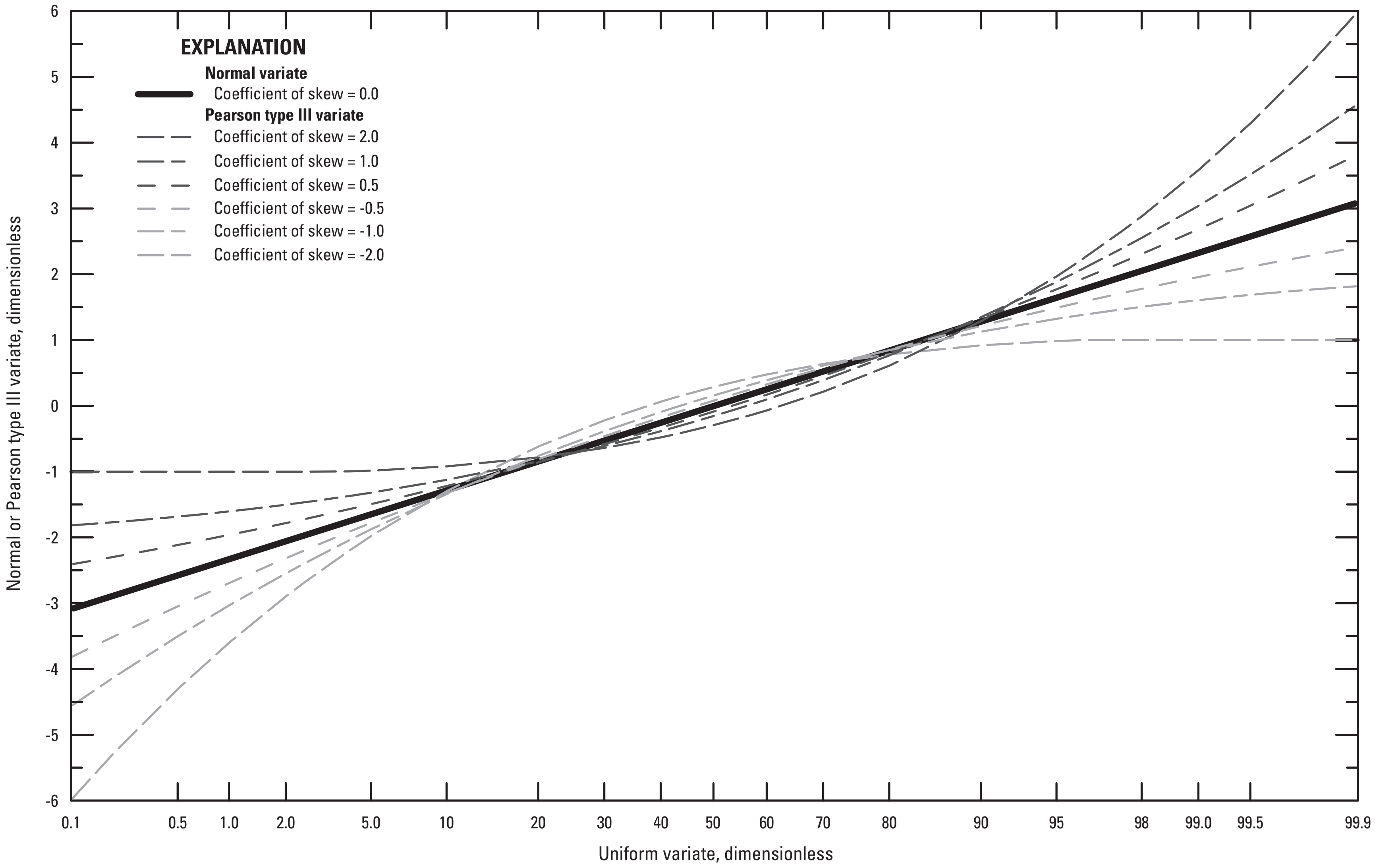


Figure 1–4. The normal variate and six Pearson type III variates generated by using the Kirby (1972) adjusted Wilson-Hilferty approximation. The normal and Pearson type III variates are graphed with respect to the uniform variates, which are scaled to the range 0–100 and plotted on a probability scale.

Generating Random Numbers by Using a Regression Relation

SELDM can generate random numbers by using a regression relation between two variables. Regression relations can be used to calculate dependent water-quality variables from user-input statistics and from stormflows by using transport curves defined by user-input statistics. The regression equation with this stochastic component is

$$Y_i = b + m \times X_i + \left(KN_i \times \sigma_r\right), \qquad (1\text{–}34)$$

where

Y_i is the dependent variable,
b is the intercept of the regression line,
m is the slope of the regression line,
X_i is the independent (or predictor) variable,
KN_i is a random normal variate, and
σ_r is the standard deviation of the residuals from the regression analysis.

The intercept, slope, and standard deviation of the residuals are input by the user, presumably from analysis of available water-quality data. For example, the Kendall-Theil Robust Line program (Granato, 2006) developed for the SELDM project can be used to calculate these regression statistics. The independent (or predictor) variable is another stochastic water-quality variable, or, for a transport curve, the stochastic sample of upstream stormflows (Granato and others, 2009).

The regression method is, essentially, the frequency-factor method. With the regression method, however, the regression equation provides an estimate of the average dependent variable (Y_r) for a given value of the independent variable (X_i). The standard deviation of the residuals and a normal frequency factor calculated from a uniform random variate U_{01} by using the algorithm AS241 (Wichura, 1988) determine the placement of the generated data point (Y_i) above or below the local regression-line value (Y_r).

The SELDM interface provides several options for defining stochastic regression relations, including those with one or more segments and those developed from logarithmic transformations of the independent and dependent variables. Granato and others (2009) provide several examples of regression relations with a stochastic component. In SELDM, regression relations may include 1 to 3 segments with positive slopes, negative slopes, or zero slopes. If a zero-slope segment is specified, the dependent variable is independent of the predictor variable, the intercept approximates the mean of the dependent variable, and the standard deviation of the residuals approximates the standard deviation of the dependent variable in the range covered by the segment.

Logarithmic regression relations may better reflect the characteristics of hydrologic data (Helsel and Hirsch, 2002; Vogel and others, 2005; Granato, 2006). Use of logarithmic regression relations also precludes generation of data that are less than or equal to zero. SELDM does not reject values that are less than zero, so the user must take care to specify untransformed regression statistics that will not produce concentration values less than zero. If a generated concentration is less than or equal to zero, however, SELDM resets the concentration to equal 0.002, which may bias output results.

Generating Random Numbers with an Input Rank Correlation Coefficient

SELDM simulates relations between selected variables by generating random numbers with a defined rank correlation coefficient. The nonparametric rank correlation coefficient (Spearman's rho, ρ) was selected for implementation in SELDM rather than the parametric correlation coefficient (Pearson's *r*) for several reasons. Pearson's *r*, the linear correlation coefficient, is not resistant to outliers and is not effective for quantifying nonlinear monotonic correlation (Haan, 1977; Helsel and Hirsch, 2002). Spearman's rho, however, characterizes monotonic correlation between the ranks of data and is not influenced by the distribution of data (Haan, 1977; Helsel and Hirsch, 2002). Rho can be used to generate two sets of U_{01} variates with a specified degree of (rank) correlation because the ranks are a function of the uniform variates. The resulting variates can be used to generate random numbers as two correlated samples from the same distribution or from different distributions.

An algorithm by Mykytka and Cheng (1994) was selected for generating correlated uniform random numbers in SELDM because the commonly used algorithms have three critical limitations: most critically, these algorithms produce samples with a substantial reduction in variance; the random variates produced by these algorithms may not fit tight tolerances for a uniform random distribution; and these algorithms are designed to produce pairs of values rather than one or more correlated random values from a master input variable. The algorithm by Mykytka and Cheng (1994) was adapted for use with SELDM because it is designed to minimize the reduction in variance, match the properties of a theoretical uniform random distribution, and to facilitate use of a master uniform random variable.

The most commonly used method for generating correlated random numbers with a uniform distribution is based on acceptance-rejection methods (Saucier, 2000). The first step is to generate a pair of two uniform variates X_{U01} and Z_{U01}. The next step is to rescale the pair to the interval from -1 to +1. If the sum of squared values of X_{U-11} and Z_{U-11} is greater than one, the pair is rejected and a new pair is generated. If the sum of squared values is less than or equal to one, the pair is used to calculate the uniform random number (Y_{U01}) that is correlated to X_{U01} with the specified rho (ρ) value. This method is implemented by the equation

$$Y_{U01} = \rho \times X_{U-11} + \sqrt{1-\rho} \times Z_{U-11} \,. \qquad (1–35)$$

This algorithm, however, creates a substantial and systematic reduction in variance in the Y_{U01} sample (fig. 1–5). The stochastic samples of Y_{U01} variates generated with this algorithm do not consistently fit a uniform distribution when the probability-plot correlation coefficient (PPCC) test (Vogel and Kroll, 1989) is applied.

Nawathe and Rao (1979) developed an algorithm for generating correlated uniform random numbers by using the ordinary least squares regression model. Nawathe and Rao (1979) used the equation

$$Y_i = \bar{Y} - \rho \times \left(\frac{\sigma_y}{\sigma_x}\right) \times \left(\bar{X} - X_i\right) + \left(KN_i \times \sqrt{\sigma_y^2\left(1-\rho^2\right)}\right) \qquad (1–36)$$

to calculate a value Y_i from an input uniform random number X_i, where

ρ is the correlation coefficient;
$\bar{X}$ is the average of input values;
$\bar{Y}$ is the average of output values;
σ_x is the standard deviation of input values;
σ_y is the standard deviation of output values; and
KN_i is a normal random variate, which supplies the random-error component.

The theoretical average and standard deviation of the input and output U_{01} variables X_i and Y_i should be 0.5 and about 0.288675, respectively. The reduction in variance for this approach is greater than the reduction in variance of the commonly used algorithm described by Saucier (2000) for absolute values of the correlation coefficient less than 0.6 but is substantially less than the reduction in variance of the commonly used algorithm as the absolute value of the correlation coefficient increases from 0.6 to 1.0 (fig. 1–5). Hirsch and Gilroy (1984) indicate that a reduction in variance in the ordinary least squares regression model is caused by incorporating the correlation coefficient into the slope of the line (as in equation 1–36). Using the line of organic correlation substantially improves but does not eliminate the reduction in variance in the uniform random variates produced (Hirsch and Gilroy, 1984; Helsel and Hirsch, 2002). Nawathe and Rao (1979) include the error component of the regression line in their algorithm; this modification should eliminate the reduction in variance (Helsel and Hirsch, 2002). Nawathe and Rao (1979), however, are creating a dependent dataset rather than fitting a line to existing data. They use the correlation coefficient and the theoretical variance of the output values to calculate the standard deviation of the error component (eq. 1–36), which causes the reduction in variance of Y_i that is a function of the correlation coefficient. Furthermore, the stochastic samples of Y_{U01} variates generated with the regression methods based on either ordinary least squares or the line of organic correlation do not consistently fit a uniform distribution when the PPCC test (Vogel and Kroll, 1989) is applied.

The algorithm by Mykytka and Cheng (1994) uses the correlation coefficient and two independent U_{01} values to generate a third intermediate value, and then adjusts the intermediate value to produce the final correlated U_{01} value. The mean, standard deviation, and PPCC test statistics for the population of adjusted U_{01} values generated by using this algorithm will be within expected statistical limits for large sample sizes. The mean reduction in the standard deviation of the final correlated U_{01} values produced by this algorithm was less than one tenth of a percent over the full range of rho values (fig. 1–5). However, the algorithm by Mykytka and Cheng (1994) does produce a small amount of bias in the rank correlations. More than 154,000 stochastic samples were generated to define this bias, and 4 polynomial equations were developed for use with SELDM to correct for this bias. If the absolute value of the desired value of the correlation coefficient (|ρ|) is less than or equal to 0.2, an adjusted value (ρ^*) is calculated as

$$\rho^* = |\rho| + (0.0578 \times |\rho|) - 0.0012 \,. \qquad (1–37)$$

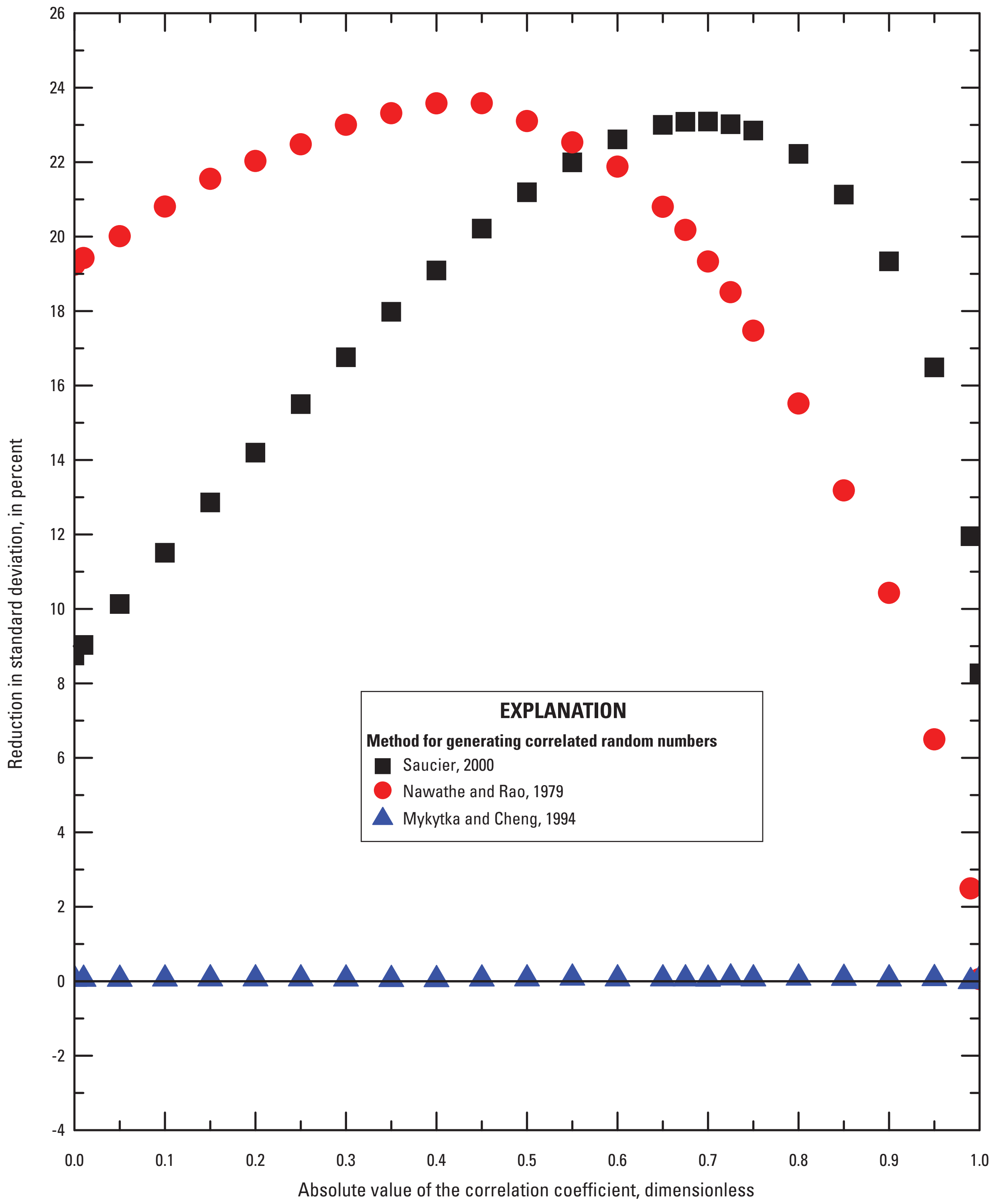


Figure 1–5. The percent reduction in the standard deviation of the dependent uniform random-number sample as a function of the absolute value of the rank correlation coefficient (Spearman's rho). Negative values denote an increase in the standard deviation. The theoretical standard deviation of a sample of uniform random numbers in the range between 0 and 1 is about 0.288675.

If the absolute value of the desired value of the correlation coefficient (|ρ|) is less than or equal to 0.7, an adjusted value (ρ^*) is calculated as

$$\rho^* = |\rho| - (0.3245 \times |\rho|^2) + (0.3155 \times |\rho|) - 0.0527 . \tag{1–38}$$

If the absolute value of the desired value of the correlation coefficient (|ρ|) is less than or equal to 0.77, an adjusted value (ρ^*) is calculated as

$$\rho^* = |\rho| - (0.126 \times |\rho|) + 0.0974 . \tag{1–39}$$

If the absolute value of the desired value of the correlation coefficient (|ρ|) is less than or equal to 0.97, an adjusted value (ρ^*) is calculated as

$$\rho^* = |\rho| - (0.6814 \times |\rho|^3) + (2.2569 \times |\rho|^2) - (2.3823 \times |\rho|) + 0.8078 , \tag{1–40}$$

and if |ρ| is greater than 0.97, then

$$\rho^* = |\rho| . \tag{1–41}$$

Given the value of ρ^*, the algorithm by Mykytka and Cheng (1994) uses three coefficients *A*, *B*, and *C* with the input U_{01} value (X_i) and the independent U_{01} value ($Y01_i$) to calculate the third intermediate value ($Y02_i$). The three coefficients are

$$A = \rho^* , \tag{1–42}$$

$$B = \sqrt{1-(\rho^*)^2} , \tag{1–43}$$

and

$$C = \frac{1-A-B}{2} . \tag{1–44}$$

The equation for calculating the third intermediate value is

$$Y02_i = A \times X_i + B \times Y01_i + C . \tag{1–45}$$

The value of the correlated U_{01} value (Y_i) is calculated by using an equation that depends on the value of ρ* and $Y02_i$:

if $\rho^* \le \frac{1}{\sqrt{2}}$ and $C \le Y02_i \le A+C$, then

$$Y_i = \frac{(Y02_i - C)^2}{2AB}; \tag{1–46}$$

if $\rho^* \le \frac{1}{\sqrt{2}}$ and $A+C < Y02_i \le B+C$, then

$$Y_i = \frac{Y02_i - (A/2) - C}{B}; \tag{1–47}$$

if $\rho^* \le \frac{1}{\sqrt{2}}$ and $B+C < Y02_i \le A+B+C$, then

$$Y_i = 1 - \frac{(A+B+C-Y02_i)^2}{2AB}; \tag{1–48}$$

if $\rho^* > \frac{1}{\sqrt{2}}$ and $C \le Y02_i \le B+C$, then

$$Y_i = \frac{(Y02_i - C)^2}{2AB}; \tag{1–49}$$

if $\rho^* > \frac{1}{\sqrt{2}}$ and $B+C < Y02_i \le A+C$, then

$$Y_i = \frac{Y02_i - (B/2) - C}{A}; \quad \text{and} \tag{1–50}$$

if $\rho^* > \frac{1}{\sqrt{2}}$ and $\mathrm{A}+C < Y02_i \le A+B+C$, then

$$Y_i = 1 - \frac{(A+B+C-Y02_i)^2}{2AB}. \tag{1–51}$$

If the original correlation coefficient (ρ) is negative, then the negatively correlated U_{01} value (Y_i) is calculated as

$$Y_i = 1 - Y_i. \tag{1–52}$$

The SELDM user can specify a rank correlation coefficient (rho) between several pairs of variables in the input dataset, but, as with other stochastic variables, the rho value in the output dataset may vary substantially from the input value. The four polynomial bias-correction equations eliminate bias in the average of resultant correlation coefficients, but they do not eliminate the inevitable variations in the stochastically generated values. The reason for this variation is that SELDM generates a random sample rather than the complete population of stochastic variables, and the rate of convergence of the rank correlation coefficient is slow. These variations are more pronounced for low values of rho than for high values of rho (Haan, 1977, Caruso and Cliff, 1997).

Confidence intervals for rho can be estimated by methods developed for Pearson's *r* because Spearman's rho is analogous to Pearson's *r* between the ranks of the samples (Haan, 1977; Caruso and Cliff, 1997; Helsel and Hirsch, 2002). Examples of the theoretical 95-percent confidence intervals for selected values of Spearman's rank correlation coefficients generated by using the large-sample approximation (Haan, 1977; Caruso and Cliff, 1997) are shown in figure 1–6. These examples demonstrate that the confidence intervals of rho are a function of the value of rho and the sample size. For example, if the user selects a value of rho of 0.25 and a stochastic sample size of 1,000 points, then 95 percent of the samples generated would be expected to have rho values between 0.19 and 0.31 inclusive (fig. 1–6). If a sample size of 2,000 points is selected, then 95 percent of the samples generated would be expected to have rho values between 0.21 and 0.29 inclusive. In comparison, 95 percent of 1,000 point samples with a specified rho value of 0.9 would be expected to have sample rho values between 0.89 and 0.91 inclusive (fig. 1–6). The range of uncertainty of the rho values input by the user is probably larger than the numerical variations in the SELDM model because many environmental-monitoring datasets comprise measurements from relatively few samples (Helsel and Hirsch, 2002). For example, Granato (2010) calculated rho values between prestorm streamflows and the corresponding runoff coefficients from available stormflow-monitoring datasets for 42 sites to see if prestorm streamflow was a good explanatory variable for variability in runoff coefficients. About 67 percent of these datasets comprised measurements from fewer than 50 samples; only one dataset comprised measurements from more than 100 samples. The 95-percent confidence limit for a dataset with 50 samples and a rho value of 0.75 would be expected to include values in the range from 0.60 to 0.85 (fig. 1–6). This is not to say that the output sample is more accurate than the actual input data, but instead that numerical variations in large stochastic samples are expected to be within the range of uncertainty of the input values.

Generating Random Numbers Adjusted for Conditional Probability

SELDM generates random numbers that are adjusted for conditional probability censoring by using an algorithm developed in this study. Conditional probability methods are used when a proportion of data are equal to a censored value, and the rest of the data can be characterized by a probability distribution. A common application of conditional probability methods is to datasets that, with the exception of a few zero values, follow a lognormal or log-Pearson type III distribution (Haan, 1977; Chow and others, 1988; Stedinger and others, 1993). In SELDM, the values of the U_{01} variates between 0 and 1 are rescaled to obtain the full lognormal or log-Pearson type III distribution indicated by the average, standard deviation, and skew of the logarithms of nonzero streamflows. To implement this algorithm, a U_{01} variate is generated. If the value of the variate is less than or equal to the proportion of censored values (in this case, the proportion of zero flows), then the censored value (zero) is returned. Otherwise, the uniform variate is rescaled as follows:

$$U_R = \frac{\left(U_{01} - P_C\right)}{\left(1 - P_C\right)}, \tag{1–53}$$

where

P_c is the proportion of censored measurements,
U_{01} is the original uniform variate (in the range between P_c and 1), and
U_R is the rescaled uniform variate (in the range between 0 and 1).

The rescaled uniform variate is used to generate a normal variate (K_N) or Pearson type III variate (K_P) and is input to equation 1–12 with the mean and standard deviation of the logarithms of nonzero values.

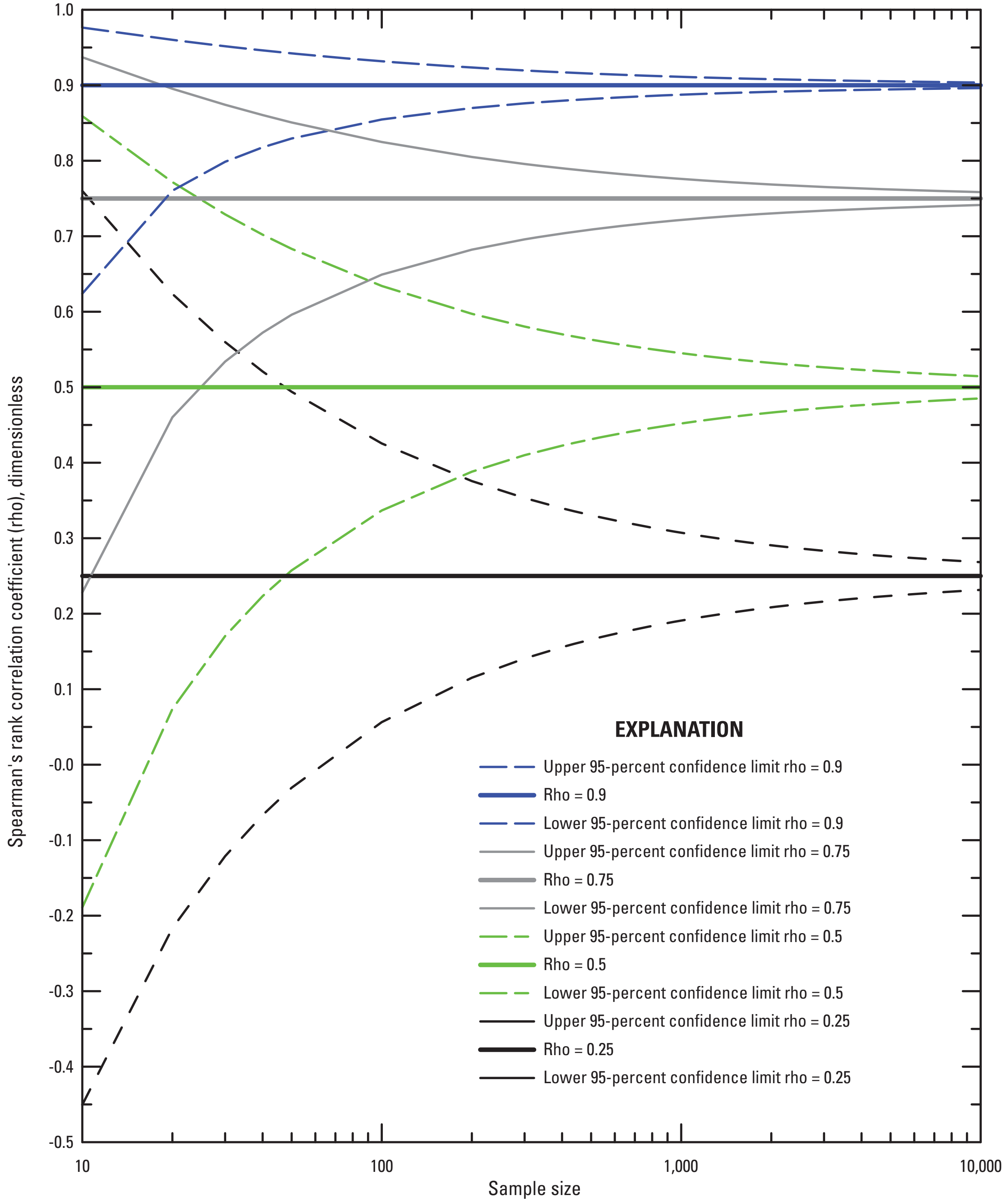


Figure 1–6. Examples of the theoretical 95-percent confidence intervals for selected values of Spearman's rank correlation coefficients generated by using the large-sample approximation (Haan, 1977; Caruso and Cliff, 1997).

Calculating the Number of Storms to be Simulated in a Stochastic Sample

In SELDM, the series of independent runoff events is modeled as a Poisson process (Granato, 2010). A Poisson process is commonly used to model discrete events on a continuous time scale with the time between events modeled as an exponential distribution (Haan, 1977). In SELDM, the time between storm-event midpoints (delta) is modeled as a two-parameter exponential distribution with a minimum value that is equal to the minimum interevent time (Granato, 2010). The number of events per year is not prespecified. SELDM randomly produces the delta values and then successively groups storms into annual-load accounting years where the sum of delta values exceeds the length of a year (8,760 hours for three years and 8,784 hours in each fourth year).

Maximizing the number of stochastic storm events must be balanced with the appearance of uncertainty in model results. If too few storms are generated, stochastic variations in the statistics of the U_{01} variates generated may preclude convergence toward the values of the input statistics (fig. 1–3). If the output values do not converge toward the values of input statistics, then biased results may misrepresent the risks for exceeding water-quality objectives and lead to the implementation of mitigation measures at sites where such measures are unnecessary. If too few storm events are generated, then extreme (large or small) values that define the potential occurrence of water-quality excursions may not be included in the analysis, thereby indicating that mitigation is not necessary at sites where such measures may be necessary. If too many events are generated, however, the large number of events may indicate an accuracy or precision in simulation results that is not warranted given the uncertainty of input data and statistics. Also, if too many storms are generated, stochastic extension of input statistics beyond available data may misrepresent the actual risks for exceeding (or not exceeding) water-quality goals. These principles apply to both the number of runoff-producing storm events and the number of years simulated.

A compromise solution was developed to balance the number of storms and the number of years simulated. Granato (2010) calculated synoptic storm-event statistics for 2,610 hourly-precipitation data stations with at least 25 years of data during the period 1965–2006. Among the 15 U.S. Environmental Protection Agency rain zones, the average annual number of runoff-producing storm events ranged from 17 in the arid Southwest to 62 in the humid Northwest. The average annual number of runoff-producing storm events ranged from 4 at station 048893 in southwestern California to 93 at station 456858 in northwestern Washington State. For example, 1,000 simulated events would represent a 250 year simulation in southwestern California but only 10 year simulation in northwestern Washington State. Although simulation results for 1,000 events would adequately represent statistics for individual storms in a humid area, an annual output dataset consisting of only 10 values would not provide statistics useful for estimating the potential variability in annual loads. Although data generated by the 250-year simulation for southwestern California may seem to be more precise than the data for most other areas, the output may overrepresent available data in this case. A linear equation relating the average number of storm events per year to the minimum number of storms events to be stochastically generated was developed to balance these competing objectives:

$$N_{Min} = 725 + 17 \times \bar{N}_{Annual}\,, \qquad (1\text{–}54)$$

where

N_{Min} is the minimum number of storm events to be used in the stochastic simulation, and

$\bar{N}_{Annual}$ is the average annual number of runoff-producing storm events selected by the user.

This equation yields about 1,014 storm events for the southwestern rain zone (equivalent to about 60 years of record) and about 1,779 storm events for the northwestern rain zone (equivalent to about 29 years of record). The slope (17) and the intercept (725) in equation 1–54 are preset in the SELDM program code, but they are annotated to facilitate modification if this is warranted for a specific application.

References Cited

Abramowitz, Milton, and Stegun, I.A., eds., 1964, Handbook of mathematical functions with formulas, graphs, and mathematical tables: Washington, D.C., U.S. Department of Commerce, National Bureau of Standards, Applied Mathematics Series, v. 55, 1,046 p.

Back, W.E., Boles, W.W., and Fry, G.T., 2000, Defining triangular probability distributions from historical cost data: Journal of Construction Engineering, v. 126, no. 1, p. 29–37.

Bobee, Bernard, and Ashkar, Fahim, 1991, The gamma family and derived distributions applied in hydrology: Littleton, Colo., Water Resources Publications, 203 p.

Caruso, J.C., and Cliff, Norman, 1997, Empirical size, coverage, and power of the confidence intervals for Spearman's rho: Educational and Psychological Measurement, v. 57, no. 4, p. 637–654.

Cheng, K.S., Ciang, J.L., and Hsu, C.W., 2007, Simulation of probability distributions commonly used in hydrological frequency analysis: Hydrological Processes, v. 51, p. 51–60.

Chow, V.T., 1954, The log-probability law and its engineering applications, *in* Proceedings of the American Society of Civil Engineers, v. 80, no. 536, 25 p.

Chow, V.T., Maidment, D.R., and Mays, L.W., 1988, Applied hydrology: New York, McGraw-Hill, Inc., 572 p.

Devroye, Luc, 1986, Non-uniform random variate generation: New York, Springer-Verlag, 843 p.

Driscoll, E.D., Palhegyi, G.E., Strecker, E.W., and Shelley, P.E., 1989, Analysis of storm event characteristics for selected rainfall gages throughout the United States: U.S. Environmental Protection Agency, OCLC 30534890, 43 p.

Gentle, J.E., 2003, Random number generation and Monte Carlo methods (2d ed.): New York, Springer Science+Business Media, Inc., 381 p.

Granato, G.E., 2006, Kendall-Theil Robust Line (KTRLine—version 1.0)—A Visual Basic program for calculating and graphing robust nonparametric estimates of linear-regression coefficients between two continuous variables: U.S. Geological Survey Techniques and Methods, book 4, chap. A7, 31 p., CD–ROM.

Granato, G.E., 2010, Methods for development of planning-level estimates of stormflow at unmonitored sites in the conterminous United States: Federal Highway Administration Report FHWA–HEP–09–005, 90 p., CD–ROM.

Granato, G.E., Carlson, C.S., and Sniderman, B.S., 2009, Methods for development of planning-level estimates of water quality at unmonitored stream sites in the conterminous United States: Federal Highway Administration Report FHWA–HEP–09–003, 53 p., CD–ROM.

Granato, G.E., and Cazenas, P.A., 2009, Highway-Runoff Database (HRDB version 1.0)—A data warehouse and preprocessor for the stochastic empirical loading and dilution model: Federal Highway Administration Report FHWA–HEP–09–004, 57 p.

Haan, C.T., 1977, Statistical methods in hydrology: Ames, Iowa, Iowa State University Press, 378 p.

Hellekalek, Peter, 1998, Good random number generators are (not so) easy to find: Mathematics and Computers in Simulation, v. 46, p. 485–505.

Helsel, D.R., and Hirsch, R.M., 2002, Statistical methods in water resources—Hydrologic analysis and interpretation: U.S. Geological Survey Techniques of Water-Resources Investigations, book 4, chap. A3, 510 p.

Hirsch, R.M., and Gilroy, E.J., 1984, Methods of fitting a straight line to data—Examples in water resources: American Water Resources Association, Water Resources Bulletin, v. 20, no. 5, p. 705–711.

Johnson, David, 1997, The triangular distribution as a proxy for the beta distribution in risk analysis: The Statistician, v. 46, no. 3, p. 387–398.

Kacker, R.N., and Lawrence, J.F., 2007, Trapezoidal and triangular distributions for Type B evaluation of standard uncertainty: Metrologia, v. 44, no. 2, p. 117–127.

Kirby, W.H., 1972, Computer-oriented Wilson-Hilferty transformation that preserves the first three moments and the lower bound of the Pearson type III distribution: Water Resources Research, v. 8, no. 5, p. 1251–1254.

L'Ecuyer, Pierre, 1988, Efficient and portable combined random number generators: Communications of the Association for Computing Machinery, v. 31, no. 6, p. 742–751.

L'Ecuyer, Pierre, 1998, Random number generation, chap. 4 *in* Banks, Jerry, ed., The Handbook on Simulation: New York, John Wiley, Inc., 66 p.

L'Ecuyer, Pierre, 1999, Good parameters and implementations for combined multiple recursive random number generators: Operations Research, v. 47, no. 1, p. 159–164.

L'Ecuyer, Pierre, and Simard, Richard, 2007, TestU01—A C library for empirical testing of random number generators: Association for Computing Machinery, Transactions on Mathematical Software, v. 33, no. 4, article 22, 40 p.

L'Ecuyer, Pierre, and Simard, Richard, 2009, TestU01—A C library for empirical testing of random number generators, user's guide, compact version: Montreal, Canada, University of Montreal, 214 p. Also available at http://www.iro.umontreal.ca/~lecuyer/.

Marsaglia, George, and Tsang, W.W., 2002, Some difficult-to-pass tests of randomness: Journal of Statistical Software, v. 7, no. 3, p. 1–9.

McCullough, B.D., 2008, Microsoft Excel's "Not the Wichmann–Hill" random number generators: Computational Statistics & Data Analysis, v. 52, p. 4587–4593.

McCullough, B.D., and Wilson, Berry, 2005, On the accuracy of statistical procedures in Microsoft Excel 2003: Computational Statistics & Data Analysis, v. 49, p. 1244–1252.

Mykytka, E.F., and Cheng, C.Y., 1994, Generating correlated random variates based on an analogy between correlation and force *in* Tew, J.D., Manivannan, S., Sadowski, D.A., and Seila, A.F., eds., Proceedings of the 1994 Winter Simulation Conference of the Association for Computing Machinery, December 11–14, 1994, Lake Buena Vista, Fla., p. 1413–1416.

Nawathe, S.P., and Rao, B.V., 1979, A simple technique for the generation of correlated random number sequences: Institute of Electrical and Electronics Engineers, Transactions on Systems, Man, and Cybernetics, v. SMC–9, no. 2, p. 96–102.

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T., 1992, Numerical recipes in Fortran 77—The art of scientific computing (2d ed.): New York, Cambridge University Press, 992 p.

Salas, J.D., 1993, Analysis and modeling of hydrologic time series, chap. 19—*in* Maidment, D.R., ed., Handbook of Hydrology: New York, McGraw Hill, p. 19.1–19.72.

Saucier, Richard, 2000, Computer generation of statistical distributions: U.S. Army Research Laboratory Report ARL–TR–2168, 105 p.

Stedinger, J.R., Vogel, R.M., and Foufoula-Georgiou, Efi, 1993, Frequency analysis of extreme events, chap. 18 *in* Maidment, D.R., ed., Handbook of Hydrology: New York, McGraw-Hill, Inc., p. 18.1–18.66.

U.S. Environmental Protection Agency, 2001, Risk assessment guidance for Superfund, v. 3, part A, Process for conducting probabilistic risk assessment: U.S. Environmental Protection Agency, Office of Emergency and Remedial Response, Report EPA 540–R–02–002, 385 p.

Vogel, R.M., and Kroll, C.N., 1989, Low-flow frequency analysis using probability-plot correlation coefficients: Journal of Water Resources Planning and Management, v. 115, no. 3, p. 338–357.

Vogel, R.M., Rudolph, B.E., and Hooper, R.P., 2005, Probabilistic behavior of water-quality loads: Journal of Environmental Engineering, v. 131, no. 7, p. 1081–1089.

Wichura, M.J., 1988, Algorithm AS241—The percentage points of the normal distribution: Applied Statistics, Journal of the Royal Statistical Society, series C, v. 37, no. 3, p. 477–484.

Wilson, E.B., and Hilferty, M.M., 1931, The distribution of chi-square: Proceedings of the National Academy of Science, v. 17, p. 684–688.