Simulation of life distributions

Enrico Zio

Simulation of life distributions

ENRICO ZIO[1, 2]

[1] *Chair on Systems Science and the Energetic Challenge, European Foundation for New Energy-Electricite' de France, at Ecole Centrale Paris- Supelec, France*

[2]*Energy Department, Politecnico di Milano, Italy*

ABSTRACT

This chapter introduces the principles underlying the simulation of the probability distributions which typically characterize the life/failure processes of engineered components and systems. The basic procedures for sampling random numbers are reviewed and examples are provided with reference to the uniform, exponential and Weibull distributions which are among the most commonly used probability distributions in reliability, availability and safety analysis.

Keywords: Random Variable, Uniform Distribution, Exponential Distribution, Weibull Distribution, Markov Process, Sampling, Inverse Transform Method.

SAMPLING RANDOM VALUES FROM A PROBABILITY DISTRIBUTION

Let X be a random variable (rv) obeying a cumulative distribution function (cdf) $\mathrm{F_X(x)}$ defined as

$$\mathrm{Prob}\{\mathrm{X \le x}\} = \mathrm{F_X(x)}\ ;\ \mathrm{F_X}(-\infty) = 0;\ F_X(\infty) = 1 \qquad (1)$$

In the following, if the rv X obeys a cdf $\mathrm{F_X(x)}$ we shall write $X \sim F_X(x)$.

From the definition it follows that $\mathrm{F_X(x)}$ is a non-decreasing function and we further assume that it is continuous and differentiable as well. The corresponding probability density function (pdf) is then

$$f_X(x) = \frac{dF_x(x)}{dx} \quad ; \quad f_X(x) \ge 0 \quad ; \quad \int_{-\infty}^{\infty} f_x(x)dx = 1 \qquad (2)$$

We now aim at sampling numbers from the cdf $\mathrm{F_X(x)}$. This amounts to obtaining a sequence of $N >> 1$ values $\{X\} \equiv \{X_1, X_2, ..., X_N\}$ such that:

i) the number n of sampled values within an interval $\Delta x << X_{\max} - X_{\min}$, where $X_{\min}$ and $X_{\max}$ are the minimum and maximum values in $\{X\}$, should be such that

$$\frac{n}{N} \approx \int_{\Delta x} f_X(x)dx \qquad (3)$$

In other words, we require that the histogram of the sampled data should approximate $f_X(x)$;

ii) the X_i values should be uncorrelated;

iii) if the sequence *{X}* is periodic, the period after which the numbers start repeating should be as large as possible.

Among all the distributions, the uniform distribution in the interval [0,1), denoted as U[0,1) or, more simply U(0,1), plays a role of fundamental importance since sampling from this distribution allows obtaining rv's obeying any other distribution.

Sampling from the uniform distribution U[0,1)

The cdf and pdf of the distribution U[0,1) are:

$$U_R(r) = r \quad ; \quad u_R(r) = 1 \quad 0 \le r < 1 \qquad (4)$$

The generation of random numbers *R* uniformly distributed in [0,1) has represented, and still represents, an important subject of active research. In the beginning, the outcomes of intrinsically random phenomena were used (e.g. throwing a coin or dice,

spinning the roulette, counting of radioactive sources of constant intensity, etc.), but soon it was realized that, apart from the disuniformity due to imperfections of the mechanisms of generation or detection, the frequency of data thus obtained was too low and the sequences could not be reproduced, so that it was difficult to find and fix the errors in the computer codes in which the sampled random numbers were used.

To overcome these difficulties, the next idea was to fill Tables of random numbers to store in the computers (in 1955 RAND corporation published a Table with 10^6 numbers), but the access to the computer memory decreased the calculation speed and, above all, the sequences that had been memorized were always too short with respect to the growing necessities.

Finally, in 1956 von Neumann proposed to have the computer directly generate the "random" numbers by means of an appropriate function $g(\cdot)$ which should allow one to find the next number R_{k+1} from the preceding one R_k, i.e. $R_{k+1} = g(R_k)$.

The sequence thus generated is inevitably periodic: in the course of the sequence, when a number is obtained that had been obtained before, the subsequence between these two numbers repeats itself cyclically, i.e. the sequence enters a loop. Furthermore, the sequence itself can be reproduced so that it is obviously not "random", but rather deterministic. However, if the function $g(\cdot)$ is chosen correctly, it can be said to have a pseudo-random character if it satisfies a number of randomness tests.

In particular, Von Neumann proposed to obtain R_{k+1} by taking the central digits of the square of R_k. For example, for a computer with a four digit word, if $R_k = 4567$, then $R_k^2 = 20857489$ and $R_{k+1} = 8574$, $R_{k+2} = 5134$, and so on. This function turns out to be lengthy

to calculate and to give rise to rather short periods; furthermore, if one obtains R_k = 0000, then all the following numbers are also zero.

Presently the most commonly used methods for generating sequences $\{R\}$ of numbers from a uniform distribution are inspired by the Monte Carlo roulette game [1-6].

In a real roulette game, the ball, thrown with high initial speed, performs a large number of revolutions around the wheel and finally it comes to rest within one of the numbered compartments. In an ideal machine nobody would doubt that the final compartment, or its associated number, is actually uniformly sampled among all the possible compartments or numbers. In the domain of the real numbers, within the interval [0,1), the game could be modelled by throwing a point on the positive *x*-axis very far from the origin, utilizing a method having an intrinsic dispersion much larger than unity. The difference between the value so obtained and the largest integer smaller than this value may then be reasonably assumed as sampled from *U*[0,1). Obviously this statement is true if a *suitable* method is utilized for throwing the point. In a computer, the above procedure is performed by means of a mixed congruential relationship of the kind

$$R_{k+1} = (aR_k + c) \bmod m. \tag{5}$$

In words, the new number R_{k+1} is the remainder, modulo *m* (a positive integer), of an affine transform of the old R_k, with non-negative integer coefficients *a* and *c*. The above expression in some way resembles the uniform sampling in the roulette game, $aR_k + c$ playing the role of the distance travelled by the ball and *m* that of the wheel

circumference. The sequence so obtained is made up of numbers $R_k < m$ and it is periodic with period $p<m$. For example, if we choose $R_0 = a = c = 5$ and $m=7$, the sequence is {5, 2, 1, 3, 6, 0, 5, …}, with a period $p=6$. The sequences generated with the above described method are actually deterministic so that the sampled numbers are more appropriately called *pseudorandom* numbers. However, the constants *a,c,m* may be selected so that:

- the sequence satisfies essentially all randomness tests;
- the period *p* is very large.

Clearly the numbers generated by the above procedure are always smaller than *m* so that, when divided by *m*, they lie in the interval [0,1).

Research to develop algorithms for generating pseudorandom numbers is still ongoing. Good statistical properties, low speed in numbers generation and reproducibility are central requirements for these algorithms to be suitable for MC simulation.

Other Pseudo - Random Number Generation (PRNG) algorithms include the Niederreiter [7], Sobol [8], and Mersenne Twister [9] algorithms. For example, this latter allows generating numbers with an almost uniform distribution in the range [0, 2k-1], where k is the computer word length (nowadays, k=32 or 64). Further details on other methods are given in [10]-[21], with wide bibliographies which we suggest to the interested reader.

Before leaving this issue, it is important to note that for the generation of pseudo-random numbers U[0,1) many computer codes do not make use of machine subroutines, but use congruential subroutines which are part of the program itself. Thus, for example, it is possible that an excellent program executed on a machine with a word of length

different from the one it was written for, gives absurd results. In this case it should not be concluded that the program is 'garbage', but it would be sufficient to appropriately modify the subroutine that generates the random numbers.

Sampling by the inverse transform method: continuous distributions

Let $X \in (-\infty,+\infty)$ be a rv with cdf $F_X(x)$ and pdf $f_X(x)$, viz.,

$$F_X(x) = \int_{-\infty}^{x} f_X(x')dx' = \Pr\{X \le x\} \tag{6}$$

Since $F_X(x)$ is a non decreasing function, for any $y \in [0,1)$, its inverse may be defined as

$$F_X^{-1}(y) = \inf\{x : F_X(x) \ge y\} \tag{7}$$

With this definition, we take into account the possibility that in some interval $[x_s, x_d]$ $F_X(x)$ is constant (and $f_X(x)$ zero), that is

$$F_X(x) = \gamma \qquad \text{for} \quad x_s \le x \le x_d \tag{8}$$

In this case, from the definition in equation (1) it follows that corresponding to the value γ, the minimum value x_s is assigned to the inverse function $F_X^{-1}(\gamma)$. This is actually as if $F_X(x)$ were not defined in (x_s,x_d]: however, this does not represent a disadvantage, since values in this interval can never be sampled because the pdf $f_X(x)$ is zero in that interval. Thus, in the following, we will suppose that the intervals (x_s,x_d] (open to the left and closed to the right), in which $F_X(x)$ is constant, are excluded from the definition domain of the rv *X*. By so doing the $F_X(x)$ will always be increasing (instead of non-decreasing).

We now show that it is always possible to obtain values $X \sim F_X(x)$ starting from values *R* sampled from the uniform distribution U_R[0,1) [1-6]. In fact, if *R* is uniformly distributed in [0,1), we have

$$Pr\{R \le r\} = U_R(r) = r \tag{9}$$

Corresponding to a *number R* extracted from a $U_R(r)$, we calculate the *number X* = $F_X^{-1}(R)$ and wish to determine its distribution. As it can be seen in Figure 1, for the variable *X* we have

$$Pr\{X \le x\} = Pr\{F_X^{-1}(R) \le x\} \tag{10}$$

Because F_X is an increasing function, by applying F_X to the arguments at the rhs of equation (6), the inequality is conserved and from equation (8) we have

$$Pr\{X \le x\} = Pr\{R \le F_X(x)\} = F_X(x) \tag{11}$$

It follows that $X = F_X^{-1}(R)$ is extracted from $F_X(x)$. Furthermore, because $F_X(x) = r$

$$Pr\{X \le x\} = Pr\{R \le r\} \tag{12}$$

In terms of cdf,

$$U_R(R) = F_X(X) \quad \text{and} \quad R = \int_{-\infty}^{X} f_X(x')dx' \tag{13}$$

This is the fundamental relationship of the inverse transform method which for any *R* value sampled from the uniform distribution $U_R[0,1)$ gives the corresponding *X* value sampled from the $F_X(x)$ distribution (Figure 1). However, it often occurs that the cdf $F_X(x)$ is non-invertibile analytically, so that from (13) it is not possible to find $X \sim F_X(x)$ as a function of $R \sim U[0,1)$. An approximate procedure that is often employed in these cases consists in interpolating $F_X(x)$ with a polygonal function and in performing the inversion of equation (13) by using the polygonal. Clearly, the precision of this procedure

increases with the number of points of $F_X(x)$ through which the polygonal passes. The calculation of the polygonal is performed as follows:

- if the interval of variation of *x* is infinite, it is approximated by the finite interval (x_a, x_b) in which the values of the pdf $f_X(x)$ are sensibly different from zero: for example, in case of the univariate s-normal distribution $N(\mu, \sigma^2)$ with mean value μ and variance σ^2, this interval may be chosen as $(\mu - 5\sigma, \mu + 5\sigma)$;
- the interval (0,1) in which both $F_X(x)$ and $U_R(r)$ are defined is divided in *n* equal subintervals of length 1/*n* and the points $x_0 = x_a, x_1, x_2, ..., x_n = x_b$ such that $F_X(x_i) = i/n$, (i=0, 1,…, n) are found, e.g. by a numerical procedure.

At this point the sampling may start: for each *R* sampled from the distribution U_R[0,1) we compute the integer $i^* = Int(R \cdot n)$ and then obtain the corresponding *X* value by interpolating between the points $(x_{i^*}, \frac{i^*}{n})$ and $(x_{i^*+1}, \frac{i^*+1}{n})$. For example, in case of a linear interpolation we have

$$X = x_{i^*} + (x_{i^*+1} - x_{i^*})(R \cdot n - i^*) \qquad (14)$$

For a fixed number *n* of points x_i upon which the interpolation is applied, the described procedure can be improved by interpolating with arcs of parabolas in place of line segments. The arcs can be obtained by imposing continuity conditions of the function and its derivatives at the points x_i (cubic splines). The expression of *X* as a function of *R* is in

this case more precise, but more burdensome to calculate. Currently, given the ease with which it is possible to increase the RAM memory of the computers, to increase the precision it is possibly preferable to increase the number *n* of points and to use the polygonal interpolation: as a rule of thumb, a good choice is often *n*=500.

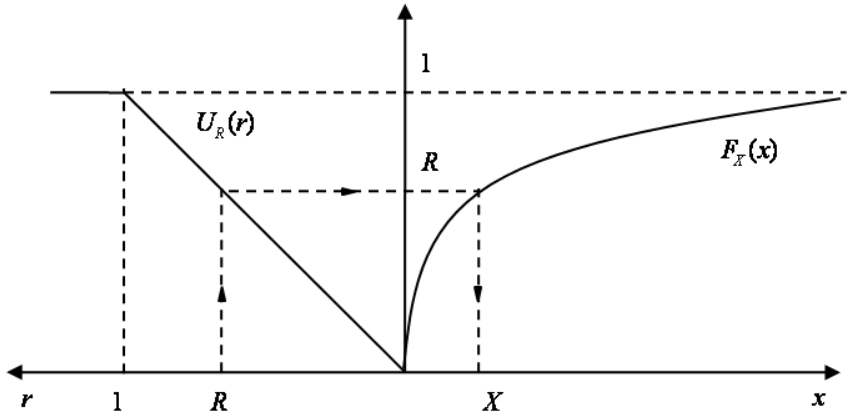

Figure 1: Inverse transform method: continuous distributions

$$R \sim U_R(r) = r \text{ in [0, 1)} \Rightarrow \mathrm{X} \sim \mathrm{F_X}(x)$$

Sampling by the inverse transform method: discrete distributions

Let *X* be a rv which can only have the discrete values x_k , *k*=0,1,…, with probabilities

$$f_k = \mathrm{P}(X = x_k) \geq 0 \quad k=0,1,\ldots \text{ (}\textbf{Erreur ! Il n'y a pas de texte répondant à ce style dans ce document.}\text{.1)}$$

Ordering the {*x*} sequence so that $x_{k-1} < x_k$, the cdf is

$$F_k = \mathrm{P}(X \leq x_k) = \sum_{i=0}^{k} f_i = F_{k-1} + f_k \qquad k=0,1,\ldots \text{ (}\textbf{Erreur ! Il n'y a pas de texte répondant à ce style dans ce document.}\text{.2)}$$

where, by definition, $F_{-1} = 0$. The normalization condition of the cdf (Equation Erreur ! **Il n'y a pas de texte répondant à ce style dans ce document.**.2) now reads

$$\lim_{k\to\infty} F_k = 1 \text{ (}\textbf{Erreur ! Il n'y a pas de texte répondant à ce style dans ce document.}\text{.3)}$$

Following the scheme of the inverse transform method, given a value R sampled from the uniform distribution, the probability that R falls within the interval $(F_{k-1}, F_k]$ is, in the discrete case

$$\mathrm{P}\left(F_{k-1} < R \le F_k\right) = \int_{F_{k-1}}^{F_k} dr = F_k - F_{k-1} = f_k = \mathrm{P}\left(X = x_k\right)$$ (**Erreur ! Il n'y a pas de texte répondant à ce style dans ce document.**.4)

In words, for any $R \sim U[0,1)$, we get the realization $X = x_k$ where k is the index for which $F_{k-1} < R \le F_k$ (Figure.1).

In practice, a realization of X is sampled from the cdf F_k through the following steps:

a) Sample an $R \sim U[0,1)$;
b) Set k=0; $F = f_o$;
c) If $R \le F$, proceed to e);
d) Viceversa, i.e., if $R > F$, set $k \leftarrow k+1$ and then $F \leftarrow F + f_k$ and proceed to c);
e) The required realization is $X = x_k$.

If the F_k values can be precomputed, e.g., if their number is finite, the cycle c)-d) may be simplified by comparing R and F_k at step c) and increasing only k at step d).

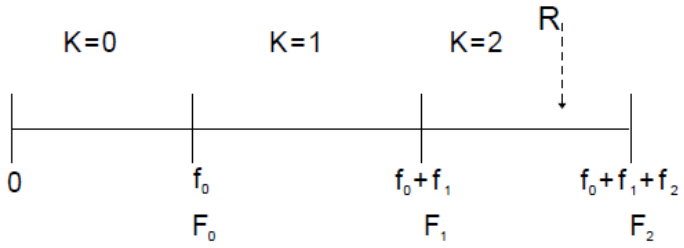

Figure.1: Inverse transform method: discrete distributions, $k = 2 \Rightarrow X = x_2$.

EXAMPLES OF APPLICATIONS OF THE SAMPLING METHODS

Uniform distribution in the interval (a,b)

A rv X is uniformly distributed in the interval (a,b) if

$$F_X(x) = \frac{x-a}{b-a} \qquad f_X(x) = \frac{1}{b-a} \quad \text{for} \quad a \le x \le b$$

$$= 0 \qquad = 0 \qquad \text{for} \quad x < a \tag{15}$$

$$= 1 \qquad = 0 \qquad \text{for} \quad x > b$$

Substitution in equation (13) and solving with respect to X yields

$$X = a + (b-a)R \tag{16}$$

Exponential distribution

A rv $X \in (0,\infty)$ is said to be exponentially distributed if its cdf $F_x(x)$ and pdf $f_x(x)$ are

$$F_X(x) = 1 - e^{-\int_0^x \lambda(u)du} \; ; \; f_X(x) = \lambda(x) e^{-\int_0^x \lambda(u)du} \quad \text{for} \quad 0 \le x \le \infty \tag{17}$$

$$= 0 \qquad = 0 \qquad \text{otherwise}$$

where $\lambda(\cdot)$ is the transition rate.

In the following, we shall refer to the exponential distribution of the random variable *T*, representing the *time to failure* of a component, with cdf $F_T(t)$ and pdf $f_T(t)$. In this case, the transition rate $\lambda(t)$ is called *hazard function* or *failure rate*.

With regards to sampling a realization *t* of the exponentially distributed random variable *T*, this can be obtained by solving equation (13), i.e.

$$\int_0^t \lambda(u)\,du = -\log(1-R) \tag{18}$$

where the realization $R \sim U[0,1)$.

Let us first consider the time homogeneous case, i.e. with constant λ. This situation corresponds to the *useful life* for which the component has been designed to operate and during which failures occur at random with no influence from the usage time. Correspondingly, equation (17) becomes:

$$F(t) = 1 - e^{-\lambda t} \quad ; \quad f(t) = \lambda e^{-\lambda t} \tag{19}$$

Realizations of the associated exponentially distributed rv T are easily obtained from the inverse transform method. The sampling of a given number $N >> 1$ of realizations is performed by repeating N times the following procedure:

- sample a realization of $R \sim U[0,1)$
- compute $t = -\frac{1}{\lambda}\log(1-R)$

Weibull distribution

A generalization of the above case in the time domain consists in assuming that the probability density of occurrence of an event (e.g. the failure of a component), namely λ, is time dependent. A case, commonly considered in practice is that in which the pdf is of the kind

$$\lambda(t) = \beta \alpha t^{\alpha-1} \tag{20}$$

with $\beta > 0, \alpha > 0$. The corresponding distribution is called Weibull's distribution and was proposed in the 1950's by W. Weibull in the course of his studies on the strength of materials. The cdf and pdf of the Weibull distribution are:

$$F(t) = 1 - e^{-\beta t^{\alpha}} \quad , \quad f(t) = \alpha \beta t^{\alpha-1} e^{-\beta t^{\alpha}} \tag{21}$$

In the particular case of $\alpha = 1$ the Weibull distribution reduces to the exponential distribution with constant transition rate $\lambda = \beta$.

In practice, a realization *t* of the random variable *T* is sampled from the Weibull distribution through the following steps:

- sampling of a realization of the random variable $R \sim U[0,1)$
- computation of $t = \left(-\frac{1}{\beta} \ln(1-R) \right)^{1/\alpha}$

REFERENCES

1. Cashwell E. D., Everett C. J., A practical manual on the Monte Carlo method for random walk problems, Pergamon Press, N. Y., 1959.
2. Rubinstein R. Y., Simulation and the Monte Carlo method, Wiley, 1981.
3. Kalos M. H., Whitlock P. A., Monte Carlo methods. Volume I: basics, Wiley, 1986.
4. Lux I., Koblinger L., Monte Carlo particle transport methods: neutron and photon calculations, CRC Press, 1991.
5. Dubi A., Monte Carlo Applications in Systems Engineering, Wiley, 1999.
6. E. Zio, *"The Monte Carlo Simulation Method for System Reliability and Risk Analysis"*, Springer, 2013.
7. Niederreiter, H., *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM, 1992.

8. Sobol, I.M., *The distribution of points in a cube and the approximate evaluation of integrals*, USSR Comput. Math. Math. Phys., 7 (4), 86-112, 1967.
9. Matsumoto, M., Nishimura, T., *Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator*. ACM Transactions on Modeling and Computer Simulation, 8 (1), 1998, 3–30.
10. Rubinstein, R.Y., *Simulation and the Monte Carlo Method*, Wiley, 1981.
11. Kalos, M.H., Whitlock, P.A., *Monte Carlo Methods*, Wiley, 2008.
12. Fang, K.T., Hickernell, F.J., Niederreiter H., *Monte Carlo and Quasi-Monte Carlo Methods 2000*, Springer-Verlag, 2002.
13. Niederreiter, H., Spanier, J., *Monte Carlo and Quasi-Monte Carlo Methods*, Springer, 1998.
14. Liu, J.S., *Monte Carlo Strategies in Scientific Computing*, Springer Series in Statistics, Springer-Verlag, 2001.
15. L'Ecuyer, P., *Uniform random number generation***,** Annals of Operational Research, 53, 1994, 77-120.
16. L'Ecuyer, P., *Uniform Random Number Generators*, in International Encyclopedia of Statistical Science, M. Lovric Ed., Springer-Verlag, 2011, Part 21, 1625-1630.
17. Shortle, J. F., L'Ecuyer, P., *Introduction to Rare-Event Simulation*, in Wiley Encyclopedia of Operations Research and Management Science, John Wiley, 2011.
18. L'Ecuyer, P., *Random Number Generators*, in Wiley Encyclopedia of Operations Research and Management Science, Third Edition, S.I. Gass and M.C. Fu Eds., Springer-Verlag, 2011.
19. L'Ecuyer, P., Mandjes, M., Turn, B., *Importance Sampling and Rare Event Simulation*, in Rare Event Simulation Using Monte Carlo Methods, G. Rubino and B. Turn Eds, Wiley, 2009, 17-38.
20. L'Ecuyer, P., *Pseudorandom Number Generators*, in Wiley Encyclopedia of Quantitative Finance, R. Cont, Ed., Wiley, 2010, 1431-1437.
21. L'Ecuyer, P., *Random Number Generation and Quasi-Monte Carlo*, in Wiley Encyclopedia of Actuarial Science, J. Teugels and B Sundt Eds., John Wiley, 2004, 3, 1363-1369.
22.

FIGURE CAPTIONS

Figure 1: Inverse transform method: continuous distributions

$R \sim U_R(r) = r$ *in [0, 1)* $\Rightarrow$ $X \sim F_X(x)$

Figure.2: Inverse transform method: discrete distributions, $k = 2 \Rightarrow X = x_2$.