Generating Nonhomogeneous Poisson Processes

Raghu Pasupathy, Email: pasupath@vt.edu
Department of Industrial & Systems Engineering, Virginia Tech, Blacksburg, VA 24061, USA

We present an overview of existing methods to generate pseudorandom numbers from a nonhomogeneous Poisson process. We start with various definitions of the nonhomogeneous Poisson process, present theoretical results (sometimes with a proof) that form the basis of existing generation algorithms, and provide algorithm listings. Whenever available, we also provide links to sources containing computer codes. With the intent of guiding users seeking an appropriate algorithm for a given setting, we emphasize computationally burdensome operations within each algorithm. Our treatment includes both one-dimensional and two-dimensional Poisson processes.

Key words: statistics; simulation; random process generation; Poisson processes.

Recall that a *counting process* $\{N_t, t \geq 0\}$ is a stochastic process defined on a sample space Ω such that for each $\omega \in \Omega$, the function $N_t(\omega)$ is a "realization" of the number of "events" happening in the interval $(0, t]$, with $N_0(\omega) = 0$. By this definition, $N_t(\omega)$ is automatically integer valued, non-decreasing, and right-continuous for each ω. A nonhomogeneous Poisson process is a type of counting process that is characterized as follows.

Definition 1. *A counting process $\{N_t, t \geq 0\}$ is called a nonhomogeneous Poisson process if:*

(i) $\forall t, s \geq 0$, *and* $0 \leq u \leq t$, $N_{t+s} - N_t$ *is independent of* N_u*;*

(ii) $\forall t, s \geq 0, Pr\{N_{t+s} - N_t \geq 2\} = o(s)$*; and*

(iii) $\forall t, s \geq 0, Pr\{N_{t+s} - N_t = 1\} = \lambda(t)s + o(s)$, *where* $\lambda(t)$ *is some positive-valued function.*

Definition 1 has been adopted from Billingsley [1, pp. 297], with the notation $o(s)$ used in the usual sense — to denote a function $v(s)$ that satisfies $\lim_{s\to 0} v(s)/s = 0$ [19, pp. 1]. The function $\lambda(t)$ appearing in Definition 1, called the rate function, completely characterizes the Poisson process. Various other definitions of a Poisson process are available and quite prevalent. See for instance Cox and Lewis [4], Ross [16], Gnedenko and Kovalenko [6], and Çinlar [3]. (Section 23 in [1] discusses the equivalence of various definitions of the Poisson process.) For example, Çinlar [3] provides the following definition based on the sample-paths of N_t that is particularly relevant to the current context.

Definition 2. *A counting process $\{N_t, t \geq 0\}$ is called a nonhomogeneous Poisson process if:*

(i) $\forall t, s \geq 0$, *and* $0 \leq u \leq t$, $N_{t+s} - N_t$ *is independent of* N_u*;*

(ii) $\forall t \geq 0$ *and for almost all* ω*, the mapping* $t \to N_t(\omega)$ *has jumps of unit magnitude only, i.e.,* $N_t(\omega) - \lim_{s \uparrow t} N_s(\omega) = 0$ *or* 1 $\forall t \geq 0$ *and for almost all* ω*.*

It is worth noting that Definitions 1 and 2 are not equivalent — the latter definition allows (unit) jumps of positive probability in the sample-paths, in which case the rate function $\lambda(t)$ appearing in Definition 1 simply does not exist.

In what follows, we discuss existing methods to generate pseudorandom numbers from a nonhomogeneous Poisson process. By this we mean generating, on a digital computer, a realization of event times $\{t_i, i = 1, 2, \ldots\}$ from a Poisson process. The Poisson process is specified either through its rate function $\lambda(t)$ (when it exists), or more generally through its expectation function $\Lambda(t) \equiv \mathrm{E}[N_t]$. When the rate function $\lambda(t)$ exists, $\Lambda(t) = \int_0^t \lambda(y)\, dy$.

1 Generating Nonhomogeneous Poisson Processes

As discussed in "Generating Homogeneous Poisson Processes," homogeneous Poisson processes can be generated very efficiently and in a fairly straightforward fashion. This is in some contrast with nonhomogeneous Poisson processes, where generation methods tend to be much less straightforward. We categorize the available methods for generating nonhomogeneous Poisson processes into three broad groups: (i) inversion methods, (ii) order-statistics methods, and (ii) acceptence-rejection methods. We discuss each of these in what follows.

1.1 Inversion

The earliest known inversion method seems to be the technique devised by Çinlar [3, pp. 96]. It is based on an interesting property of nonhomogeneous Poisson processes with a continuous expectation function $\Lambda(t)$.

Theorem 1 (Çinlar, 1975)**.** *Let* $\Lambda(t), t \geq 0$ *be a positive-valued, continuous, nondecreasing function. Then the random variables* $T_1, T_2, \ldots$ *are event times corresponding to a nonhomogeneous Poisson process with expectation function* $\Lambda(t)$ *if and only if* $\Lambda(T_1), \Lambda(T_2), \ldots$ *are the event times corresponding to a homogeneous Poisson process with rate one.*

Theorem 1 provides a method to generate event times from a nonhomogeneous Poisson process that is straightforward in principle. First generate event times from a homogenous Poisson process with rate one, and then invert $\Lambda(\cdot)$ to obtain the event times of the required process.

Algorithm 3 (Çinlar's Method)

```
(0) Initialize $s = 0$.
(1) Generate $u \sim U(0,1)$.
(2) Set $s \leftarrow s - \log(u)$.
(3) Set $t \leftarrow \inf\{v : \Lambda(v) \geq s\}$.
(4) Deliver $t$.
(5) Go to Step (1).
```

When the expectation function $\Lambda(t)$ falls in a particular family of functions that facilitates efficient analytic inversion in Step (3), Algorithm 3 tends to be very fast. This happens only infrequently, however, and implementation often involves employing expensive numerical quadrature for inversion [14] in Step (3). Adoption of Çinlar's method should thus be considered accordingly. Code for a rudimentary version of Çinlar's method can be found through the website `<https://filebox.vt.edu/users/pasupath/pasupath.htm>`.

Theorem 1 assumes the continuity of the expectation function $\Lambda(t)$. When $\Lambda(t)$ has discontinuities, Algorithm 3 can be adapted in a somewhat straightforward fashion. See, for instance Çinlar [3, pp. 100] and Resnick [15] for different treatments.

Another inversion approach to generating nonhomogeneous Poisson processes stems from the distribution of inter-event times. Specifically, consider the ith inter-event time $X_i = T_{i+1} - T_i$ conditional on the first i event times $T_1 = t_1, T_2 = t_2, \ldots, T_i = t_i$. We can derive the cdf of X_i (conditional on $T_1, T_2, \ldots, T_i$) as follows.

$$\begin{aligned}
F_{t_i}(x) &= \Pr\{X_i \leq x | T_j = t_j, j = 1, 2, \ldots, i\} \\
&= \Pr\{N_{t_i+x} - N_{t_i} \geq 1 | T_j = t_j, j = 1, 2, \ldots, i\} \\
&= \Pr\{N_{t_i+x} - N_{t_i} \geq 1\} \\
&= 1 - \Pr\{N_{t_i+x} - N_{t_i} = 0\} \\
&= 1 - \exp(-\Lambda(t_i + x) + \Lambda(t_i)), \qquad (1)
\end{aligned}$$

where the third equality is from the independent increments property of the Poisson process. (We again note here that Equation (1) is not true in general if $\Lambda(t)$ has jumps. It can, however, be shown that (1) will still hold if the location of the jumps has a Poisson number of events with mean equal to the value of the jump [15].)

The cdf appearing in (1) provides a natural method of generating a nonhomogeneous Poisson process — given the previous i event times, generate the $(i+1)$th event time as the sum of the ith event time and the ith inter-event time distributed according to F_{t_i}.

Algorithm 4

```
(0) Initialize $t = 0$.
(1) Generate $x \sim F_t$ given by (1).
(2) Set $t \leftarrow t + x$.
(3) Deliver $t$.
(4) Go to Step (1).
```

Like Çinlar's method, Algorithm 4 is direct, exact, and elegant. However, and again like Çinlar's method, Algorithm 4 is efficient only when the form of the expectation function $\Lambda(t)$ renders easy generation from F_t. Specifically, suppose the cdf-inverse method is used to generate from F_t in Step (1). If the nonhomogeneous Poisson process is specified through the rate function $\lambda(t)$, this amounts to finding x satisfying

$$\int_t^{t+x} \lambda(y)\, dy = -\ln(1-u), \text{ where } u \sim U(0,1). \tag{2}$$

In general, the problem appearing in (2) can be solved only using numerical quadrature [14], and can hence be computationally expensive. (Public domain software is available for this purpose [9].)

By contrast, if enough structure is present and known, efficient tailor-made techniques can be devised to solve (2). One such example is the piecewise-linear rate function considered by Klein and Roberts [8], and by Lee and Wilson [10], with success. Specifically, Klein and Roberts [8] assume that $\lambda(t)$ is a continuous, piecewise-linear function connecting the points $(0, \lambda_0), (t_1, \lambda_1), (t_2, \lambda_2), \ldots$ Therefore, generating from the distribution F_t in Step (1) of Algorithm 4 amounts to identifying x such that $-\log(1-u) = \Lambda(t_i, x)$ where $\Lambda(t_i, x) \equiv \Lambda(t_i + x) - \Lambda(t_i)$. Through straightforward integration of the piecewise-linear function, Klein and Roberts [8] come up with an expression for the function $\Lambda(t_i, x)$ and the corresponding root x that satisfies $-\log(1-u) = \Lambda(t_i, x)$.

One important point needs to be noted in implementing the inversion methods that we have discussed. Depending on the nature of the expectation function $\Lambda(t)$, the distribution of the time until the next event may not be well defined. In particular, there may be a positive probability of observing no events past a particular point in time, i.e., there exists $\delta > 0$ such that $\Pr\{X_i > x\} > \delta$ for all $x > 0$. In such a case, Step (3) in Çinlar's method may have no solution, and F_t in (1) may not be a proper cdf.

1.2 Order Statistics

Let us now state a general result on the distribution of the event times of a nonhomogeneous Poisson process with expectation function $\Lambda(t)$.

Theorem 2 (Cox and Lewis, 1962). *Let $T_1, T_2, \ldots$ be random variables representing the event times of a nonhomogeneous Poisson process with continuous expectation function $\Lambda(t)$, and let N_t represent the total number of events occurring before time t in the process. Then, conditional on the number of events $N_{t_0} = n$, the event times $T_1, T_2, \ldots, T_n$ are distributed as order statistics from a sample with distribution function $F(t) = \Lambda(t)/\Lambda(t_0)$ for $t \in [0, t_0]$.*

Theorem 2 is a generalization of the result for homogeneous Poisson processes that was used in Algorithm 2. It naturally gives rise to Algorithm 5 for generating random variates from a nonhomogeneous Poisson process with expectation function $\Lambda(t)$ in a fixed interval $[0, t_0]$.

Algorithm 5

```
(1) Generate $n \sim \text{Poisson}(\Lambda(t_0))$.
(2) Independently generate $n$ random variates $t'_1, t'_2, \ldots, t'_n$ from the cdf $F(t) = \Lambda(t)/\Lambda(t_0)$.
(3) Order $t'_1, t'_2, \ldots, t'_n$ to obtain $t_1 = t'_{(1)}, t_2 = t'_{(2)}, \ldots, t_n = t'_{(n)}$.
(4) Deliver $t_1, t_2, \ldots, t_n$.
```

The efficiency of Algorithm 5 depends critically on Step (2), where n random variates are to be generated from the cdf $F(t)$. If $\lambda(t)$ is assumed to be log-linear ($\lambda(t) = \lambda e^{\alpha_1 t}$) as in [11], Algorithm 5 becomes very efficient since $F(t) = (e^{\alpha_1 t} - 1)/(e^{\alpha_1 t_0} - 1), t \in [0, t_0], \alpha_1 \neq 0$ is easily invertible. Lewis and Shedler, in the same article [11], provide an even faster variant based on gap statistics. This is extended to the case of second degree exponential polynomial rate functions (i.e., $\lambda(t) = \exp(\alpha_0 + \alpha_1 t + \alpha_2 t^2)$) in [13].

1.3 Acceptance-Rejection

Currently the most popular method for generating nonhomogenous Poisson processes is the "process analogue" of acceptance-rejection called *thinning* [12]. The intuitive idea behind thinning is to first find a constant rate function $\lambda_u(t) = \lambda_u$ which dominates the desired rate function $\lambda(t)$, next generate from the implied homogeneous Poisson process with rate $\lambda_u(t) = \lambda_u$, and then reject an appropriate fraction of the generated events so that the desired rate $\lambda(t)$ is achieved. The following theorem is the basis for this procedure.

Theorem 3 (Lewis and Shedler, 1979). *Consider a nonhomogeneous Poisson process with rate function $\lambda_u(t), t \geq 0$. Suppose that $T_1^*, T_2^*, \ldots, T_n^*$ are random variables representing event times from the nonhomogeneous Poisson process with rate function $\lambda_u(t)$, and lying in the fixed interval $(0, t_0]$. Let $\lambda(t)$ be a rate function such that $0 \leq \lambda(t) \leq \lambda_u(t)$ for all $t \in [0, t_0]$. If the ith event time T_i^* is independently deleted with probability $1 - \lambda(t)/\lambda_u(t)$ for $i = 1, 2, \ldots, n$, then the remaining event times form a nonhomogeneous Poisson process with rate function $\lambda(t)$ in the interval $(0, t_0]$.*

A thinning algorithm for generating random variates from a nonhomogeneous Poisson process follows immediately.

Algorithm 6

```
(0) Initialize $t = 0$.
(1) Generate $u_1 \sim U(0,1)$.
(2) Set $t \leftarrow t - \frac{1}{\lambda_u} \log u_1$.
(3) Generate $u_2 \sim U(0,1)$ independent of $u_1$.
(4) If $u_2 \leq \lambda(t)/\lambda_u$ then deliver $t$.
(5) Goto Step (1).
```

In the above thinning algorithm, at any time t, a variate generated from the majorizing rate function $\lambda_u(t) = \lambda_u$ is accepted with probability $\lambda(t)/\lambda_u$. The efficiency thus depends critically on how "snugly" λ_u approximates $\lambda(t)$ — rate functions $\lambda(t)$ that exhibit heavy fluctuations in time will render the thinning algorithm inefficient. A proof that the thinning algorithm in fact produces the desired stochastic process can be found in [12].

Ross [17] provides a straightforward modification to the thinning method with the objective of mitigating excessive rejection. (Also see [2, pp. 179].) The intuitive idea behind this extension is *piecewise thinning* — majorize the desired rate function using an appropriately chosen piecewise constant function having k pieces, and then perform regular thinning within each piece. Specifically, first divide the horizon of interest $[0, t_0]$ into k intervals $[s_{j-1}, s_j), j = 1, 2, \ldots, k$, and choose constants λ_j satisfying $\lambda_j \geq \sup_{t \in [s_{j-1}, s_j)} \{\lambda(t)\}$. Random variates from a homogeneous Poisson process are then generated in the interval $[s_{j-1}, s_j), j = 1, 2, \ldots, k$ by generating exponential random variates with mean $1/\lambda_j, j = 1, 2, \ldots, k$. The resulting event times are each accepted independently with probability $\lambda(t)/\lambda_j, j = 1, 2, \ldots, k$. Certain simple edge corrections are required when a generated exponential inter-event time straddles two pieces of the majorizing function. A complete algorithm listing can be found in [17, pp. 85].

While piecewise thinning alleviates excessive rejection, it can still be inefficient depending on how well the majoring step function approximates the rate function. Modern methods thus go one step further than piecewise thinning and construct a majorizing function $\lambda_u(t)$ that is both a good approximation of the rate function and has a form that lends itself to easy generation. This results in a method that is a hybrid of two methods — usually an inversion method such as Algorithm 4 used to generate random variates from the constructed majorizing function $\lambda_u(t)$, and a thinning method to reject a correct fraction of the generated random variates. We now state this in the form of an algorithm.

Algorithm 7

```
(0) Initialize $t = 0$.
(1) Generate $x \sim F_t$ where $F_t(x)$ is given in (1) with $\Lambda(t) = \int_0^t \lambda_u(t)\,dt$.
(2) Set $t \leftarrow t + x$.
(3) Generate $u \sim U(0,1)$ independent of $x$.
(4) If $u \leq \lambda(t+x)/\lambda_u(t+x)$ then deliver $t$.
(5) Goto Step (1).
```

Such a hybrid algorithm has been used with much success to generate a nonhomogeneous Poisson process on a fixed interval $[0, t_0]$ in [10]. The majorizing function $\lambda_u(t)$ used in [10] is a continuous piecewise-linear function that is optimal in a certain precise sense.

2 Generating Two-Dimensional Poisson Processes

A counting process $\{N(t), t \geq 0\}$ is said to constitute a two-dimensional nonhomogeneous Poisson process on $C \subseteq \mathbb{R}^2$ with rate function $\lambda(x,y) > 0$ if

(i) The number of events in a region $R \subseteq C$ is Poisson distributed with parameter $\Lambda(R) = \iint_R \lambda(x,y)\,dx\,dy$.

(ii) The number of events occurring in any finite set of nonoverlapping regions are mutually independent.

(We do not go into details about the complete characterization of the nature of C or the subsets R for which (i) and (ii) should be satisfied. See [15, pp. 303] for a more rigorous treatment.) The Poisson process is said to be a homogeneous Poisson process if $\lambda(x,y)$ in the above definition is constant on

C, i.e., $\lambda(x, y) = \lambda$ for $(x, y) \in C$ [7]. For generation from a two-dimensional homogeneous Poisson process, consult "Generation of Homogeneous Poisson Processes."

For generating nonhomogeneous Poisson processes in a bounded region C, a simple first algorithm is based on the idea that the location (X, Y) of an event in C, conditional on the number of events $N = n$ that occur in C, is distributed over the region C according to the probability density function

$$f(x, y) = \frac{\lambda(x, y)}{\iint_C \lambda(x, y)\, dx\, dy}, \quad (x, y) \in C. \tag{3}$$

(See [15, Chapter 4] for more on this. Particularly, the above idea extends seamlessly even into higher dimensions.) This, combined with the fact that the number of events N in C is Poisson distributed with mean $\iint_C \lambda(x, y)\, dx\, dy$, gives rise to a conceptually simple algorithm for generating events.

Algorithm 8

(1) Generate $n \sim \text{Poisson}(\iint_C \lambda(x, y)\, dx\, dy)$.

(2) If $n = 0$ then exit. Otherwise independently generate n events $(x_i, y_i), i = 1, 2, \ldots, n$ that are distributed in C according to the density $f(x, y)$ given in (3).

(3) Deliver $(x_i, y_i), i = 1, 2, \ldots, n$.

Of course, the ease with which the above algorithm is implemented will depend on the nature of the rate function $\lambda(x, y)$. Particularly, it will depend on how easily $\lambda(x, y)$ is integrated over C, and whether $f(x, y)$ lends itself to easy generation. A lot is known about generation from continuous densities with arbitrary form — see [5] for more on this.

Another popular method for generating nonhomogeneous Poisson processes is the multidimensional analogue of thinning, and stems from a theorem by Lewis and Shedler [12].

Theorem 4 (Lewis and Shedler, 1979)**.** *Let $(X_i, Y_i), i = 1, 2, \ldots$ be the Cartesian coordinates of a two-dimensional Poisson process on $C \subseteq I\!R^2$ with rate function $\lambda^*(x, y) \geq 0$. If the ith event is independently deleted with probability $1 - \lambda(x, y)/\lambda^*(x, y)$ where $0 \leq \lambda(x, y) \leq \lambda^*(x, y)$ for all $(x, y) \in C$, the resulting process is a Poisson process on C with rate function $\lambda(x, y)$.*

A basic thinning algorithm would thus involve identifying λ_u such that $\lambda_u \geq \lambda(x, y)$ for all $(x, y) \in C$, generating random variates from a homogeneous Poisson process with rate λ_u (see "Generation of Homogeneous Poisson Processes"), and then deleting events with probability $1 - \lambda(x, y)/\lambda_u$. As in one-dimensional thinning, such an algorithm would be very inefficient in contexts where the rate function $\lambda(x, y)$ exhibits major fluctuations. Methods that are analogous to piecewise thinning and the hybrid method for one-dimensional thinning, if devised, will potentially prove much more efficient.

A more recent idea for generating nonhomogeneous Poisson processes is provided by Saltzman, Drew, and Leemis [18], and is based on the idea of conditioning. (See also [15, Section 4.10] for a deeper treatment of this idea.) Specifically, they note the following two facts about a two-dimensional nonhomogeneous Poisson process with rate function $\lambda(x, y) > 0$ on $C \subset \mathbb{R}^2$.

(i) If $\{(X_i, Y_i)\}$ are events corresponding to the two-dimensional process, then the abcissae $\{X_i\}$ correspond to a one-dimensional nonhomogeneous Poisson process with rate function $\lambda_X(x) = \int_{C(x)} \lambda(x, y)\, dy$, where $C(x) = \{y : (x, y) \in C\}$.

(ii) If (X, Y) denotes the location of an event from the two-dimensional process, the conditional random variable $Y|X = x$ has the probability density function $\lambda(x, y)/\lambda_X(x)$.

The above two facts give rise to a generation algorithm that first generates the abcissa of an event and then generates its ordinate through the conditional density function provided in (ii). Like Algorithm 8, the efficiency of the resulting algorithm depends on the tractability of the rate functions $\lambda(x, y)$ and $\lambda_X(x)$. We summarize this as Algorithm 9 below.

Algorithm 9

```
(1) Initialize $i = 0$.
(2) Generate $x_i$ according to the one-dimensional nonhomogeneous Poisson process with rate function $\lambda_X(x)$.
(3) Generate $y_i$ according to the probability density function $\lambda(x_i, y)/\lambda_X(x_i)$.
(4) Deliver $(x_i, y_i)$.
(5) Set $i = i + 1$ and go to Step (2).
```

References

[1] P. Billingsley. *Probability and Measure.* Wiley, New York, NY., 1995.

[2] P. Bratley, B. L. Fox, and L. E. Schrage. *A Guide to Simulation.* Springer-Verlag, New York, 1987.

[3] E. Çinlar. *Introduction to Stochastic Processes.* Prentice-Hall, New Jersey, 1975.

[4] D. R. Cox and P. A. W. Lewis. *The Statistical Analysis of Series of Events.* Methuen, London, United Kingdom, 1966.

[5] L. Devroye. *Non-uniform Random Variate Generation.* Springer, New York, NY., 1986.

[6] B. V. Gnedenko and I. Kovalenko. *Introduction to Queueing Theory.* Jerusalem: Israel Program for Scientific Translations, Jerusalem, Israel, 1968.

[7] S. Karlin and H. M. Taylor. *A First Course in Stochastic Processes.* Academic Press, New York, NY., 1975.

[8] R. W. Klein and S. D. Roberts. A time-varying Poisson arrival process generator. *Simulation*, 43(4):193–195, 1984.

[9] M. E. Kuhl and J. R. Wilson. User's manual for mp3mle and mp3sim: software for estimating and simulating nonhomogeneous Poisson processes. Technical report, North Carolina State University, Department of Industrial Engineering, 1996.

[10] S. H. Lee, M. M. Crawford, and J. R. Wilson. Modeling and simulation of a nonhomogeneous Poisson process having cyclic behavior. *Communications in Statistics — Simulation*, 20(2 & 3):777–809, 1991.

[11] P. A. W. Lewis and G. S. Shedler. Simulation of nonhomogenous Poisson processes with log-linear rate function. *Biometrika*, 63:501–505, 1976.

[12] P. A. W. Lewis and G. S. Shedler. Simulation of nonhomogenous Poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979.

[13] P. A. W. Lewis and G. S. Shedler. Simulation of nonhomogenous Poisson processes with degree-two exponential polynomial rate function. *Operations Research*, 27(5):1026–1039, 1979.

[14] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables.* Academic Press, New York, NY., 1970.

[15] S. Resnick. *Adventures in Stochastic Processes.* Birkhäuser, Boston, MA., 1992.

[16] S. Ross. *Stochastic Processes.* Wiley, New York, NY., 1995.

[17] S. Ross. *Simulation.* Elsevier, Burlington, MA., 2006.

[18] E. Saltzman, J. Drew, and L. Leemis. Simulation of multivariate nonhomogeneous poisson processes by inversion and composition. *ACM Transactions on Modeling and Computer Simulation.* Under Review.

[19] R. J. Serfling. *Approximation Theorems of Mathematical Statistics.* John Wiley & Sons, Inc., New York, NY., 1980.