



Masuda, N., & Rocha, L. E. C. (2018). A Gillespie Algorithm for Non-Markovian Stochastic Processes. *SIAM Review*, 60(1), 95-115.  
<https://doi.org/10.1137/16M1055876>

Publisher's PDF, also known as Version of record

License (if available):  
CC BY

Link to published version (if available):  
[10.1137/16M1055876](https://doi.org/10.1137/16M1055876)

[Link to publication record in Explore Bristol Research](#)  
PDF-document

## University of Bristol - Explore Bristol Research

### General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:  
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

# A Gillespie Algorithm for Non-Markovian Stochastic Processes\*

Naoki Masuda<sup>†</sup>  
Luis E. C. Rocha<sup>‡</sup>

**Abstract.** The Gillespie algorithm provides statistically exact methods for simulating stochastic dynamics modeled as interacting sequences of discrete events including systems of biochemical reactions or earthquake occurrences, networks of queuing processes or spiking neurons, and epidemic and opinion formation processes on social networks. Empirically, the inter-event times of various phenomena obey long-tailed distributions. The Gillespie algorithm and its variants either assume Poisson processes (i.e., exponentially distributed inter-event times), use particular functions for time courses of the event rate, or work for non-Poissonian renewal processes, including the case of long-tailed distributions of inter-event times, but at a high computational cost. In the present study, we propose an innovative Gillespie algorithm for renewal processes on the basis of the Laplace transform. The algorithm makes use of the fact that a class of point processes is represented as a mixture of Poisson processes with different event rates. The method is applicable to multivariate renewal processes whose survival function of inter-event times is completely monotone. It is an exact algorithm and works faster than a recently proposed Gillespie algorithm for general renewal processes, which is exact only in the limit of infinitely many processes. We also propose a method to generate sequences of event times with a tunable amount of positive correlation between inter-event times. We demonstrate our algorithm with exact simulations of epidemic processes on networks, finding that a realistic amount of positive correlation in inter-event times only slightly affects the epidemic dynamics.

**Key words.** point process, network, numerical simulation, Laplace transform

**AMS subject classifications.** 60-08, 65C

**DOI.** 10.1137/16M1055876

**1. Introduction.** Various complex systems are driven by interactions between subsystems via time-stamped discrete events. For example, in chemical systems, a chemical reaction event changes the number of reagents of particular types in a discrete manner. Stochastic point processes, in particular Poisson processes assuming that events occur independently and at a constant rate over time, are a central tool for emulating the dynamics of chemical systems [67]. They are also useful in simulating epidemic processes in a population [14] and many other systems.

\*Received by the editors January 7, 2016; accepted for publication (in revised form) January 11, 2017; published electronically February 7, 2018.

<http://www.siam.org/journals/sirev/60-1/M105587.html>

**Funding:** The work of the first author was supported by JST, CREST, and JST, ERATO, Kawarabayashi Large Graph Project. L.E.C.R. is a Chargé de recherche of the Fonds de la Recherche Scientifique - FNRS.

<sup>†</sup>Department of Engineering Mathematics, University of Bristol, Bristol, UK (naoki.masuda@bristol.ac.uk).

<sup>‡</sup>Department of Mathematics and naXys, Université de Namur, Namur, Belgium, and Department of Public Health Sciences, Karolinska Institutet, Stockholm, Sweden.

Consider an event-driven system in which events are generated by Poisson processes running in parallel. In chemical reaction systems, each Poisson process (possibly with different rates for each process) is attached to one reaction. In epidemic processes taking place on human or animal contact networks (i.e., graphs), each Poisson process is assigned to an individual or a link, which may potentially transmit the infection. The event rate of some of the Poisson processes may change with the occurrence of a reaction or infection within the entire system. The simplest simulation method is to discretize time and then judge whether or not an event occurs in each time window, for individual processes. This widely used method is suboptimal because the size of the time window must be sufficiently small to obtain high accuracy, which is computationally expensive [70]. The Gillespie algorithm is an efficient and statistically exact algorithm for multivariate Poisson processes [21, 22, 39]. The Gillespie algorithm, or, in particular, the direct method of Gillespie [21, 22], exploits the fact that a superposition of independent Poisson processes is a single Poisson process whose event rate is the sum of those of the constituent Poisson processes. Using this mathematical property, only a single Poisson process needs to be emulated in the Gillespie algorithm.

However, for various real-world systems in which multivariate point processes have been applied (both with and without network structure), event sequences are clearly non-Poissonian. In particular, inter-event times typically obey long-tailed distributions, which are inconsistent with the exponential (i.e., short-tailed) distribution that Poisson processes produce. Examples of long-tailed distributions of inter-event times include earthquake occurrences [5, 12], firing of neurons [4, 64], social networks and the Internet [7, 24, 33, 69], financial transactions [49, 60], and crimes [37, 54]. Therefore, multivariate point processes that are not necessarily Poissonian have been used in these applications, e.g., triggered seismicity [30, 55], networks of spiking neurons [56, 66], epidemic processes [33, 50], opinion formation models [16, 65], finance [10, 17], and criminology [54, 59]. These applications call for numerical methods to efficiently and accurately simulate interacting and non-Markovian point processes.

A reasonable description of event sequences in these phenomena requires, at the very least, renewal processes, in which inter-event times are independently generated from a given distribution [13, 15]. Along these lines, one numerical approach is to use the modified next reaction method [3], which improves on both the so-called Gillespie's first reaction method [21] and the next reaction method [20]. The basic idea behind these methods is to draw the next event time for all processes from predetermined distributions, select the process that has generated the minimum waiting time to the next event, execute the event, and repeat. However, for non-Poissonian renewal processes, it is generally difficult to numerically solve the equations that determine the next event time although these methods call for the generation of only half as many random numbers in comparison to the Gillespie algorithm [70]. In addition, we can easily halve the number of random variates required in the Gillespie algorithm, such that the next reaction method and the Gillespie algorithm demand the same number of random variates (Appendix A). In what follows, we restrict ourselves to the Gillespie algorithm and its variants.

Motivated by applications to chemical reactions, many extensions of the Gillespie algorithm in the case of non-Poissonian processes assume that the dynamical change in the event rate is exogenously driven in particular functional forms [11, 46]. These extensions are not applicable for general renewal processes, because the event rate of a constituent process is a function of the time since the last event of the process, which depends on that process. In other words, we cannot assume a common exogenous

driver. Boguñá and colleagues extended the Gillespie algorithm to be applicable for general renewal processes [9] (also see [70] for further developments). However, the algorithm has practical limitations. First, it is not accurate when the number of ongoing renewal processes is small [9]. This can result in a considerable amount of approximation error in the beginning or final stages of the dynamics of epidemic or opinion formation models, for example, in which only a small number of processes is active, even in a large population [70]. Second, it is necessary to recalculate the instantaneous event rate of each process following the occurrence of every event in the entire population, a procedure that can be computationally expensive.

In the present study, we propose an innovative Gillespie algorithm, the Laplace Gillespie algorithm, which is applicable to non-Poissonian renewal processes. It exploits the mathematical properties of the Laplace transform, is accurate for an arbitrary number of ongoing renewal processes, and runs faster than the previous algorithm [9]. This article is organized as follows. In section 2, we review the original Gillespie algorithm for Poisson processes. In section 3, we review the previous extension of the Gillespie algorithm to general renewal processes [9]. In section 4, we introduce the Laplace Gillespie algorithm, together with theoretical underpinnings and examples. In section 5, we numerically compare the previous algorithm [9] and the Laplace Gillespie algorithm. In section 6, we introduce a method to generate event sequences with positive correlation in inter-event times, as is typically observed in human behavior and natural phenomena [24]. In section 7, we demonstrate our methods by performing exact simulations of an epidemic process in which inter-event times follow a power-law distribution. In section 8, we discuss the results, including limitations of the proposed algorithm. The codes used in our numerical simulations are available in the Supplementary Materials.

**2. Gillespie Algorithm.** The original Gillespie algorithm [21, 22, 39] assumes  $N$  independent Poisson processes with a rate of  $\lambda_i$  ( $1 \leq i \leq N$ ) running in parallel. Because of the independence of the different Poisson processes, the superposition of the  $N$  processes is a Poisson process with rate  $\sum_{i=1}^N \lambda_i$ . Therefore, we first draw  $\Delta t$ , an increment in time to the next event of the superposed Poisson process, from the exponential distribution given by

$$(1) \quad \phi(\Delta t) = \left( \sum_{i=1}^N \lambda_i \right) e^{-(\sum_{i=1}^N \lambda_i) \Delta t}.$$

Because the survival function (i.e., the probability that a random variable is larger than a certain value) of  $\phi(\Delta t)$  is given by  $\int_{\Delta t}^{\infty} \phi(t') dt' = e^{-(\sum_{i=1}^N \lambda_i) \Delta t}$ , we obtain  $\Delta t = -\log u / (\sum_{i=1}^N \lambda_i)$ , where  $u$  is a random variate drawn from the uniform density on the interval  $[0, 1]$ . Then, we determine the process  $i$  that has produced the event with probability

$$(2) \quad \Pi_i = \frac{\lambda_i}{\sum_{i=1}^N \lambda_i}.$$

Finally, we advance the time by  $\Delta t$  and repeat the procedure. Following the occurrence of an event, any  $\lambda_i$  is permitted to change.

**3. Non-Markovian Gillespie Algorithm.** Now consider  $N$  renewal processes running in parallel, and denote by  $\psi_i(\tau)$  the probability density function of inter-event times for the  $i$ th process ( $1 \leq i \leq N$ ). If the process is Poissonian, we obtain

$\psi_i(\tau) = \lambda_i e^{-\lambda_i \tau}$ . For such a population of general renewal processes, Boguñá and colleagues proposed an extension of the Gillespie algorithm, which they called the non-Markovian Gillespie algorithm (nMGA) [9].

To begin with, we explain their exact Gillespie algorithm for general renewal processes, which is the basis of the nMGA. A short derivation of the exact Gillespie algorithm is given in Appendix B. We denote by  $t_i$  the time elapsed since the last event of the  $i$ th process, and by

$$(3) \quad \Psi_i(t_i) = \int_{t_i}^{\infty} \psi_i(\tau') d\tau'$$

the survival function of the  $i$ th process (i.e., the probability that the inter-event time is larger than  $t_i$ ). We also set

$$(4) \quad \Phi(\Delta t | \{t_j\}) = \prod_{j=1}^N \frac{\Psi_j(t_j + \Delta t)}{\Psi_j(t_j)},$$

which is in fact equal to the probability that no process generates an event for time  $\Delta t$  (see Appendix B). The exact Gillespie algorithm for general renewal processes is given as follows:

1. Initialize  $t_j$  ( $1 \leq j \leq N$ ) for all  $j$  (for example,  $t_j = 0$ ).
2. Draw the time until the next event,  $\Delta t$ , by solving  $\Phi(\Delta t | \{t_j\}) = u$ , where  $u$  is a random variate uniformly distributed over  $[0, 1]$ .
3. Select the process  $i$  that has generated the event with probability

$$(5) \quad \Pi_i \equiv \frac{\lambda_i(t_i + \Delta t)}{\sum_{j=1}^N \lambda_j(t_j + \Delta t)},$$

where  $\lambda_i(t_i + \Delta t) \equiv \psi_i(t_i + \Delta t) / \Psi_i(t_i + \Delta t)$  is equal to the instantaneous rate of the  $i$ th process.

4. Update the time since the last event,  $t_j$ , to  $t_j + \Delta t$  ( $j \neq i$ ) and set  $t_i = 0$ .
5. Repeat steps 2–4.

Although this algorithm is statistically exact, step 2 can be time-consuming [9, 70]. To improve performance, Boguñá and colleagues introduced the nMGA. The nMGA is an approximation to the aforementioned algorithm and is exact as  $N \rightarrow \infty$ . When  $\Delta t$  is small, as is the case when  $N$  is large, (4) is approximated as

$$(6) \quad \begin{aligned} \Phi(\Delta t | \{t_j\}) &= \exp \left[ - \sum_{j=1}^N \ln \frac{\Psi_j(t_j)}{\Psi_j(t_j + \Delta t)} \right] \\ &= \exp \left[ - \sum_{j=1}^N \ln \frac{\Psi_j(t_j)}{\Psi_j(t_j) - \psi_j(t_j) \Delta t + O(\Delta t^2)} \right] \\ &\approx \exp \left[ - \Delta t \left( \sum_{j=1}^N \lambda_j(t_j) \right) \right]. \end{aligned}$$

With this approximation, the time until the next event is determined by  $\Phi(\Delta t | \{t_j\}) \approx \exp [ - \Delta t ( \sum_{j=1}^N \lambda_j(t_j) ) ] = u$ , i.e.,  $\Delta t = - \ln u / ( \sum_{j=1}^N \lambda_j(t_j) )$ . The process that generates the event is determined by setting  $\Delta t = 0$  in (5). For a Poisson process, we set  $\lambda_i(t_i) = \lambda_i$  to recover the original Gillespie algorithm (cf. (1) and (2)).

#### 4. Laplace Gillespie Algorithm.

**4.1. Algorithm.** In the nMGA, we update the instantaneous event rates for all the processes  $\lambda_j(t_j)$  ( $1 \leq j \leq N$ ) and their sum following the occurrence of each event. This is because  $t_j$  ( $1 \leq j \leq N$ ) is updated following an event. This procedure is time-consuming when  $N$  is large; we have to update  $\lambda_j(t_j)$  even if the probability density of the inter-event times for the  $j$ th process is not perturbed by an event that has occurred elsewhere.

To construct an efficient Gillespie algorithm for non-Markovian point processes, we start by considering the following renewal process, called the event-modulated Poisson process. When an event occurs, we first draw the rate of the Poisson process, denoted by  $\lambda$ , according to a fixed probability density function  $p(\lambda)$ . Then, we draw the time until the next event according to the Poisson process with rate  $\lambda$ . Upon the occurrence of the next event, we renew the rate  $\lambda$  by redrawing it from  $p(\lambda)$ . We then repeat these steps.

The event-modulated Poisson process is a mixture of Poisson processes of different rates. It is, in general, a non-Poissonian renewal process and is slightly different from a mixed Poisson process, in which a single rate is initially drawn from a random ensemble and used throughout a realization [27, 38]. It is also different from a doubly stochastic Poisson process (also known as a Cox process), in which the rate of the Poisson process is a stochastic process [27, 38, 40], or its subclass called the Markov-modulated Poisson process, in which the event rate switches in time according to a Markov process [18]. In these processes, the dynamics of the event rate are independent of the occurrence of events. In contrast, for event-modulated Poisson processes, the event rate changes upon the occurrence of events.

An event-modulated Poisson process is a Poisson process when conditioned on the current value of  $\lambda$ . Therefore, when we simulate  $N$  event-modulated Poisson processes, they are independent of each other and of the past event sequences if we are given the instantaneous rate of the  $i$ th process, denoted by  $\lambda_i$  for all  $i$  ( $1 \leq i \leq N$ ). This property enables us to construct a Gillespie algorithm. By engineering  $p(\lambda)$ , we can emulate a range of renewal processes with different distributions of inter-event times. The new Gillespie algorithm, which we call the Laplace Gillespie algorithm (the reason for this name will be made clear in section 4.2, where we discuss the algorithm's theoretical basis in the Laplace transform), is defined as the Gillespie algorithm for event-modulated Poisson processes. We denote the density of the event rate for the  $i$ th process by  $p_i(\lambda_i)$ . The Laplace Gillespie algorithm proceeds as follows:

1. Initialize each of the  $N$  processes by drawing the rate  $\lambda_i$  ( $1 \leq i \leq N$ ) according to its density function  $p_i(\lambda_i)$ .
2. Draw the time until the next event  $\Delta t = -\ln u / \sum_{j=1}^N \lambda_j$ , where  $u$  is a random variate uniformly distributed over  $[0, 1]$ .
3. Select the process  $i$  that has generated the event with probability  $\lambda_i / \sum_{j=1}^N \lambda_j$ .
4. Draw a new rate  $\lambda_i$  according to  $p_i(\lambda_i)$ . If there are processes  $j$  ( $1 \leq j \leq N$ ) for which the statistics of inter-event times have changed following the occurrence of the event generated in steps 2 and 3 (e.g., a decrease in the rate of being infected owing to the recovery of an infected neighbor), modify  $p_j(\lambda_j)$  accordingly and draw a new rate  $\lambda_j$  from the modified  $p_j(\lambda_j)$ . The event rates of the remaining processes remain unchanged.
5. Repeat steps 2–4.

**4.2. Theory.** An event-modulated Poisson process is a renewal process. The renewal process is fully characterized by the probability density of inter-event times,  $\psi(\tau)$ . For an event-modulated Poisson process with probability density of the event rate  $p(\lambda)$ , we obtain

$$(7) \quad \psi(\tau) = \int_0^\infty p(\lambda) \lambda e^{-\lambda\tau} d\lambda.$$

Integration of both sides of (7) yields the survival function of the inter-event times as follows:

$$(8) \quad \Psi(\tau) = \int_\tau^\infty \psi(\tau') d\tau' = \int_0^\infty p(\lambda) e^{-\lambda\tau} d\lambda.$$

Equation (8) indicates that  $\Psi(\tau)$  is the Laplace transform of  $p(\lambda)$ . Therefore, the necessary and sufficient condition for a renewal process to be simulated by the Laplace Gillespie algorithm is that  $\Psi(\tau)$  is the Laplace transform of a probability density function of a random variable taking nonnegative values. Although this statement can be made more rigorous if we replace  $p(\lambda)d\lambda$  by the probability distribution function, we will use the probability density representation for simplicity.

A necessary and sufficient condition for the existence of  $p(\lambda)$  is that  $\Psi(\tau)$  is completely monotone and  $\Psi(0) = 1$  [15]. The complete monotonicity is defined by

$$(9) \quad (-1)^n \frac{d^n \Psi(\tau)}{d\tau^n} \geq 0 \quad (\tau \geq 0, n = 0, 1, \dots).$$

Condition  $\Psi(0) = 1$  is satisfied by any survival function. With  $n = 0$ , (9) reads  $\Psi(\tau) \geq 0$ , which all survival functions satisfy. With  $n = 1$ , (9) reads  $\psi(\tau) \geq 0$ , which is also always satisfied. However, (9) with  $n \geq 2$  imposes nontrivial conditions.

**4.3. Examples.** In this section, we give examples of distributions of inter-event times  $\psi(\tau)$  for which the Laplace Gillespie algorithm can be used. These examples are summarized in Table 1.

**Exponential Distribution.** A Poisson process with rate  $\lambda_0$ , i.e.,  $\psi(\tau) = \lambda_0 e^{-\lambda_0\tau}$ , is trivially generated by  $p(\lambda) = \delta(\lambda - \lambda_0)$ , where  $\delta$  is the delta function.

**Power-Law Distribution Derived from a Gamma Distribution of  $\lambda$ .** Consider the case in which  $p(\lambda)$  is the gamma distribution given by

$$(10) \quad p(\lambda) = \frac{\lambda^{\alpha-1} e^{-\lambda/\kappa}}{\Gamma(\alpha) \kappa^\alpha},$$

where  $\Gamma(\alpha)$  is the gamma function,  $\alpha$  is the shape parameter, and  $\kappa$  is the scale parameter. By combining (8) and (10), we obtain

$$(11) \quad \Psi(\tau) = \frac{1}{(1 + \kappa\tau)^\alpha}.$$

The probability density of inter-event times is given by the following power-law distribution:

$$(12) \quad \psi(\tau) = \frac{\kappa\alpha}{(1 + \kappa\tau)^{\alpha+1}}.$$

**Table 1** Distributions of inter-event times,  $\psi(\tau)$ , for which the Laplace Gillespie algorithm can be used.  $H$  is the Heaviside function defined by  $H(x) = 1$  ( $x \geq 0$ ) and  $H(x) = 0$  ( $x < 0$ ).

Distribution	$\psi(\tau)$	$\Psi(\tau)$	Condition	$p(\lambda)(\lambda \geq 0)$
Exponential	$\lambda_0 e^{-\lambda_0 \tau}$	$e^{-\lambda_0 \tau}$		$\delta(\lambda - \lambda_0)$
Power law	$\frac{\lambda^{\alpha-1} e^{-\lambda/\kappa}}{\Gamma(\alpha)\kappa^\alpha}$	$\frac{\kappa}{(1+\kappa\tau)^{\alpha+1}}$		$\frac{1}{(1+\kappa\tau)^\alpha}$
Power law with exponential tail	$\frac{e^{-\lambda_0 \tau}}{(1+\kappa\tau)^\alpha} \left( \lambda_0 + \frac{\kappa\alpha}{1+\kappa\tau} \right)$	$\frac{e^{-\lambda_0 \tau}}{(1+\kappa\tau)^\alpha}$		$\frac{(\lambda-\lambda_0)^{\alpha-1} e^{-\frac{\lambda-\lambda_0}{\kappa}} H(\lambda-\lambda_0)}{\Gamma(\alpha)}$
Power law (with exponential tail)	$\propto \frac{e^{-\lambda_{\min}\tau}}{\tau}$ or $\propto \frac{1}{\tau^2}$	$\frac{e^{-\lambda_{\min}\tau} - e^{-\lambda_{\max}\tau}}{\tau(\lambda_{\max} - \lambda_{\min})}$	$\lambda_{\min} \ll \lambda_{\max}$	uniform on $[\lambda_{\min}, \lambda_{\max}]$
Power law	$\frac{\alpha+1}{\tau^{\alpha+2}} \gamma(\alpha+2, \tau)$	$\frac{\alpha+1}{\tau^{\alpha+1}} \gamma(\alpha+1, \tau)$	$\alpha > -1$	$(\alpha+1)\lambda^\alpha$
Weibull	$\alpha\mu^\alpha \tau^{\alpha-1} e^{-(\mu\tau)^\alpha}$	$e^{-(\mu\tau)^\alpha}$	$0 < \alpha \leq 1$	complicated
Gamma	$\frac{\tau^{\alpha-1} e^{-\tau/\kappa}}{\Gamma(\alpha)\kappa^\alpha}$	complicated	$0 < \alpha \leq 1$	$\frac{H(\lambda-\kappa^{-1})}{\Gamma(\alpha)\Gamma(1-\alpha)\lambda(\kappa\lambda-1)^\alpha}$
Mittag-Leffler	$\approx \frac{\beta \sin(\beta\pi)\Gamma(\beta)}{\pi\tau^{\beta+1}}$	$E_\beta(-\tau^\beta)$	$0 < \beta < 1$	$\frac{1}{\pi} \frac{\lambda^{\beta-1} \sin(\beta\pi)}{\lambda^{2\beta} + 2\lambda^\beta \cos(\beta\pi) + 1}$

When  $\alpha = 1$ ,  $p(\lambda)$  is the exponential distribution and  $\psi(\tau) \propto \tau^{-2}$  [31]. The same mathematical relationship connecting the gamma distribution and a power-law distribution has been used in superstatistics in statistical mechanics [8] and in a reinforcement learning model for generating discount rates that decay with time according to a power law [43].

**Power-Law Distribution with an Exponential Tail Derived from a Gamma Distribution of  $\lambda$ .** Consider a shifted gamma distribution [27] given by

$$(13) \quad p(\lambda) = \begin{cases} \frac{(\lambda-\lambda_0)^{\alpha-1} e^{-(\lambda-\lambda_0)/\kappa}}{\Gamma(\alpha)} & (\lambda \geq \lambda_0), \\ 0 & (0 < \lambda < \lambda_0), \end{cases}$$

where  $\lambda_0$  is a constant. By combining (8) and (13), we obtain

$$(14) \quad \Psi(\tau) = \frac{e^{-\lambda_0 \tau}}{(1 + \kappa\tau)^\alpha}.$$

By differentiating (14), we obtain a power-law distribution with an exponential tail given by

$$(15) \quad \psi(\tau) = \frac{e^{-\lambda_0 \tau}}{(1 + \kappa\tau)^\alpha} \left( \lambda_0 + \frac{\kappa\alpha}{1 + \kappa\tau} \right).$$

**Power-Law Distribution Derived from a Uniform Distribution of  $\lambda$ .** Assume that  $p(\lambda)$  is a uniform density on  $[\lambda_{\min}, \lambda_{\max}]$  [31]. We obtain

$$(16) \quad \Psi(\tau) = \frac{e^{-\lambda_{\min}\tau} - e^{-\lambda_{\max}\tau}}{\tau(\lambda_{\max} - \lambda_{\min})}$$

and

$$(17) \quad \psi(\tau) = \frac{\lambda_{\min} e^{-\lambda_{\min}\tau} - \lambda_{\max} e^{-\lambda_{\max}\tau}}{(\lambda_{\max} - \lambda_{\min})\tau} + \frac{e^{-\lambda_{\min}\tau} - e^{-\lambda_{\max}\tau}}{(\lambda_{\max} - \lambda_{\min})\tau^2}.$$



Now suppose that  $\lambda_{\min} \ll \lambda_{\max}$ . If  $\lambda_{\min} > 0$ , we obtain  $\psi(\tau) \propto e^{-\lambda_{\min}\tau}/\tau$  as  $\tau \rightarrow \infty$ , a power-law distribution with an exponential cutoff. If  $\lambda_{\min} = 0$ , we obtain  $\psi(\tau) \propto 1/\tau^2$  as  $\tau \rightarrow \infty$ .

**Power-Law Distribution Derived from a Power-Law Distribution of  $\lambda$ .** Consider the distribution

$$(18) \quad p(\lambda) = (\alpha + 1)\lambda^\alpha,$$

where  $\alpha > -1$  and  $0 \leq \lambda \leq 1$  [31]. We obtain

$$(19) \quad \Psi(\tau) = \frac{\alpha + 1}{\tau^{\alpha+1}} \gamma(\alpha + 1, \tau),$$

where  $\gamma(\alpha + 1, \tau) = \int_0^\tau x^\alpha e^{-x} dx$  is the incomplete gamma function, and

$$(20) \quad \psi(\tau) = \frac{\alpha + 1}{\tau^{\alpha+2}} \gamma(\alpha + 2, \tau).$$

As  $\tau \rightarrow \infty$ , we obtain  $\psi(\tau) \approx (\alpha + 1)\Gamma(\alpha + 2)/\tau^{\alpha+2}$ .

**Weibull Distribution.** The Weibull distribution is defined by

$$(21) \quad \Psi(\tau) = e^{-(\mu\tau)^\alpha},$$

which yields

$$(22) \quad \psi(\tau) = \alpha\mu^\alpha \tau^{\alpha-1} e^{-(\mu\tau)^\alpha}.$$

The Weibull distribution with  $\alpha = 1$  is an exponential distribution. The Weibull distribution has a longer and shorter tail than the exponential distribution when  $\alpha < 1$  and  $\alpha > 1$ , respectively. The Weibull distribution can be expressed as the Laplace transform of a  $p(\lambda)$  if and only if  $0 < \alpha \leq 1$  [35, 75]. The distribution when  $\alpha = 1/2$  is the so-called stable distribution of order  $1/2$ , for which we obtain [15, 27, 35]

$$(23) \quad p(\lambda) = \frac{m^{\frac{1}{2}} e^{-\frac{m}{4\lambda}}}{2\sqrt{\pi}\lambda^{\frac{3}{2}}}.$$

For other values of  $\alpha$  (i.e.,  $0 < \alpha < 1/2$  or  $1/2 < \alpha < 1$ ), the explicit form of  $p(\lambda)$  is complicated [35] such that the use of the Laplace Gillespie algorithm is impractical. For these  $\alpha$  values, a mixture of a small number of exponential distributions may resemble the Weibull distribution [36], such that we may be able to use  $p(\lambda)$  with point masses at some discrete values of  $\lambda$  to approximate the Weibull distribution of inter-event times.

**Gamma Distribution.** When inter-event times obey the gamma distribution, i.e.,

$$(24) \quad \psi(\tau) = \frac{\tau^{\alpha-1} e^{-\tau/\kappa}}{\Gamma(\alpha)\kappa^\alpha},$$

$\Psi(\tau)$  can be expressed as the Laplace transform of a probability density function  $p(\lambda)$  if and only if  $0 < \alpha \leq 1$  [23, 74]. We obtain [23]

$$(25) \quad p(\lambda) = \begin{cases} 0 & (0 < \lambda < \kappa^{-1}), \\ \frac{1}{\Gamma(\alpha)\Gamma(1-\alpha)\lambda(\kappa\lambda-1)^\alpha} & (\lambda \geq \kappa^{-1}). \end{cases}$$

**Mittag–Leffler Distribution.** Consider the distribution of inter-event times defined in terms of the survival function given by

$$(26) \quad \Psi(\tau) = E_\beta(-\tau^\beta),$$

where

$$(27) \quad E_\beta(z) = \sum_{n=0}^{\infty} \frac{z^n}{\Gamma(1 + \beta n)}$$

is the so-called Mittag–Leffler function. When  $0 < \beta < 1$ ,  $\Psi(\tau)$  is completely monotone, and we obtain [25, 26]

$$(28) \quad p(\lambda) = \frac{1}{\pi} \frac{\lambda^{\beta-1} \sin(\beta\pi)}{\lambda^{2\beta} + 2\lambda^\beta \cos(\beta\pi) + 1}.$$

When  $\beta = 1$ , (26) and (27) imply that  $\Psi(\tau) = e^{-\lambda\tau}$ , yielding a Poisson process. When  $0 < \beta < 1$ ,  $\Psi(\tau)$  is long-tailed with the asymptotics [19, 26]

$$(29) \quad \Psi(\tau) \approx \frac{\sin(\beta\pi)\Gamma(\beta)}{\pi\tau^\beta},$$

or, equivalently,

$$(30) \quad \psi(\tau) \approx \frac{\beta \sin(\beta\pi)\Gamma(\beta)}{\pi\tau^{\beta+1}}.$$

Therefore, this class of  $\psi(\tau)$  produces long-tailed distributions of inter-event times with a power-law exponent lying between one and two. A special case occurs when  $\beta = 1/2$ , in which case  $\Psi(\tau) = e^{-\tau^{2\beta}} [1 - \text{erf}(\tau^\beta)]$ , where  $\text{erf}(z) \equiv (2/\sqrt{\pi}) \int_0^z e^{-z'^2} dz'$  is the error function.

**Integral of a Valid Survival Function.** The function given by

$$(31) \quad \Psi^w(\tau) \equiv \frac{\int_\tau^\infty \Psi(\tau') d\tau'}{\int_0^\infty \Psi(\tau') d\tau'} = \frac{\int_\tau^\infty \Psi(\tau') d\tau'}{\langle \tau \rangle_\psi}$$

is well-defined if and only if  $\langle \tau \rangle_\psi$ , i.e., the mean inter-event time with respect to density  $\psi(\tau)$  is finite. Assume that the renewal process generated by  $\psi(\tau)$  permits use of the Laplace Gillespie algorithm. Because  $\Psi^w(\tau) \geq 0$  ( $\tau \geq 0$ ),  $d^n \Psi^w(\tau)/d\tau^n = - [d^{n-1} \Psi(\tau)/d\tau^{n-1}] / \int_0^\infty \Psi(\tau') d\tau'$  ( $n = 1, 2, \dots$ ), and  $\Psi(\tau)$  is completely monotone, it follows that  $\Psi^w(\tau)$  is completely monotone. In addition, (31) implies that  $\Psi^w(0) = 1$ . Therefore, the renewal process with survival function  $\Psi^w(\tau)$  can also be simulated by the Laplace Gillespie algorithm.

The corresponding probability density of inter-event times is given by

$$(32) \quad \psi^w(\tau) = -\frac{d\Psi^w(\tau)}{d\tau} = \frac{\Psi(\tau)}{\langle \tau \rangle_\psi}.$$

In terms of  $p(\lambda)$ , we obtain

$$(33) \quad \psi^w(\tau) = \frac{\int_0^\infty p(\lambda) e^{-\lambda\tau} d\lambda}{\int_0^\infty \frac{p(\lambda')}{\lambda'} d\lambda'}$$

and

$$(34) \quad \Psi^w(\tau) = \frac{\int_0^\infty \frac{p(\lambda)}{\lambda} e^{-\lambda\tau} d\lambda}{\int_0^\infty \frac{p(\lambda')}{\lambda'} d\lambda'}.$$

Therefore, in each update of the Laplace Gillespie algorithm with the density of inter-event times given by  $\psi^w(\tau)$ , the rate of the Poisson process  $\lambda$  should be sampled according to density  $p^w(\lambda)$ , where

$$(35) \quad p^w(\lambda) = \frac{\frac{p(\lambda)}{\lambda}}{\int_0^\infty \frac{p(\lambda')}{\lambda'} d\lambda'}.$$

For example, if  $\psi(\tau)$  is an exponential distribution, then  $\psi^w(\tau)$  is an exponential distribution of the same mean. If  $\psi(\tau)$  is the power-law distribution given by (12), then  $\psi^w(\tau)$  is a power-law distribution of the same form, with  $\alpha$  replaced by  $\alpha - 1$ .

**Product of Valid Survival Functions.** The product of two completely monotone functions,  $\Psi_1(\tau)$  and  $\Psi_2(\tau)$ , is completely monotone [15]. In addition,  $\Psi_1(0)\Psi_2(0) = 1$  if  $\Psi_1(0) = \Psi_2(0) = 1$ . Therefore, the survival function  $\Psi(\tau) \equiv \Psi_1(\tau)\Psi_2(\tau)$  admits use of the Laplace Gillespie algorithm if  $\Psi_1(\tau)$  and  $\Psi_2(\tau)$  do as well. The probability density of the event rate will be the convolution of  $p_1(\lambda)$  and  $p_2(\lambda)$ , where  $\Psi_i(\tau) = \int_0^\infty p_i(\lambda)e^{-\lambda\tau} d\lambda$  ( $i = 1, 2$ ).

**Numerical Laplace Transform.** Given an arbitrary  $p(\lambda)$ , we can in principle carry out a numerical Laplace transform to derive  $\Psi(\tau)$ . We then obtain a valid  $\psi(\tau)$  by numerically differentiating  $\Psi(\tau)$ .

**4.4. Empirical Distributions of Inter-Event Times.** We are often interested in informing multivariate point processes by empirical data of event sequences. A standard numerical approach is to emulate the dynamics (e.g., epidemic processes) on top of empirical event sequences, i.e., use empirically observed events with time stamps to induce, for example, infection events [33, 50]. There exists a Gillespie algorithm to run dynamical processes on such empirical temporal networks [70]. Another approach is to estimate  $\psi(\tau)$  from empirical data and then use a variant of the Gillespie algorithm (e.g., the nMGA or the Laplace Gillespie algorithm) to simulate stochastic point processes (e.g., epidemic processes).

The Laplace Gillespie algorithm requires the survival function,  $\Psi(\tau)$ , to be completely monotone. Under this condition, we may be able to compute the inverse Laplace transform to obtain  $p(\lambda)$  at a reasonable computational cost [2]. However, because it is likely that an empirical  $\Psi(\tau)$  is not completely monotone, we propose two alternative methods to estimate  $p(\lambda)$  from empirical data. The first method is to fit a completely monotone survival function of inter-event times, such as (11), to given data. The second method is to estimate a finite mixture of exponential distributions of different means to approximate the empirical  $\psi(\tau)$  or  $\Psi(\tau)$ . Likelihood or other cost-function methods are available for performing this estimation [29, 35, 36, 42, 58]. If the empirical  $\Psi(\tau)$  is completely monotone, the approximation error is guaranteed to decay inversely proportional to the number of constituent distributions [44]. For both of these methods, we should be mindful of the bias in the estimation caused by a finite time window of observation [41].

**4.5. Initial Conditions.** When we begin to run  $N$  processes, one approach is to initially draw the inter-event time for each process from  $\psi(\tau)$ . This initial condition

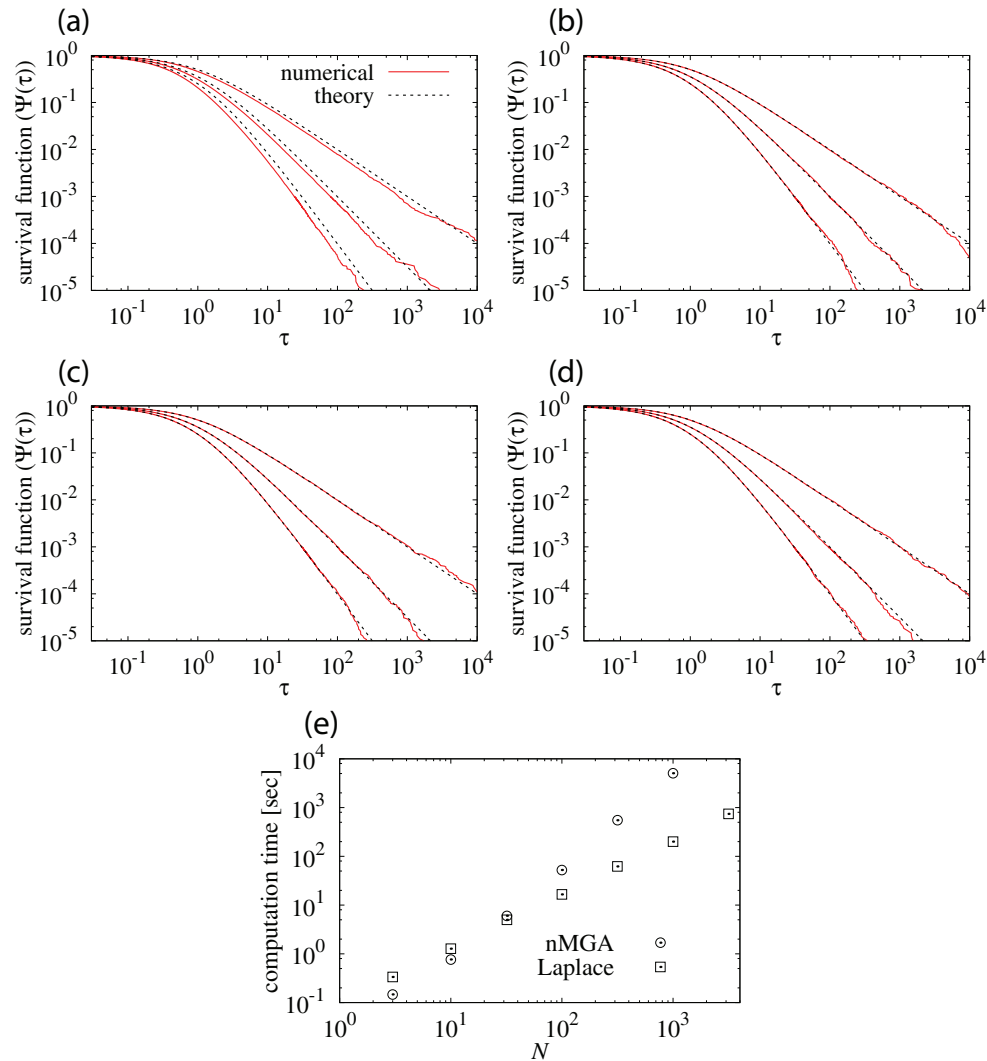
defines the so-called ordinary renewal process [13]. An alternative model, called the equilibrium renewal process, assumes that the process has begun at time  $t = -\infty$ , such that the first inter-event time for each process, drawn at  $t = 0$ , uses the equilibrium distribution of waiting times to the next event, rather than  $\psi(\tau)$  (i.e., the distribution of inter-event times) [13]. In fact, the equilibrium distribution of waiting times to the next event coincides with  $\psi^w(\tau)$  given by (32) [13, 15, 51]. To simulate the equilibrium renewal process, we start by drawing the rates of the Poisson processes according to  $p^w(\lambda)$  given by (35). Afterwards, we draw the event rates according to  $p(\lambda)$ .

**5. Numerical Performance.** In this section, we compare the performances of the nMGA and the Laplace Gillespie algorithm. We use the power-law distribution of inter-event times given by (12). Because  $\kappa$  only controls the scale of inter-event times, we set  $\kappa = 1$  without loss of generality. To generate gamma-distributed random variates, we use a well-known algorithm [48] and adapt an open source code [1] for our purposes. We generate  $\lfloor N/3 \rfloor$  processes by (12) with  $\alpha = 1$ , another  $\lfloor N/3 \rfloor$  processes with  $\alpha = 1.5$ , and another  $N - 2\lfloor N/3 \rfloor \approx N/3$  processes with  $\alpha = 2$ . We continue the simulation until one of the  $N$  processes generates  $10^6$  inter-event times. We employ the ordinary renewal process such that the initial inter-event time for each process is drawn from  $\psi(\tau)$ .

The survival functions for one of the processes with  $\alpha = 1$ , one of those with  $\alpha = 1.5$ , and one of those with  $\alpha = 2$  are shown by the solid curves for the nMGA and the Laplace Gillespie algorithm in Figures 1(a) and 1(b), respectively, for  $N = 10$ . The theoretical survival function (11) is plotted with the dashed curves. The results obtained from the Laplace Gillespie algorithm (Figure 1(b)) are more accurate than those obtained from the nMGA. This is because the nMGA is exact only in the limit of  $N \rightarrow \infty$ , whereas the Laplace Gillespie algorithm is exact for any  $N$ . When  $N = 10^3$ , the nMGA is sufficiently accurate (Figure 1(c)), as is the Laplace Gillespie algorithm (Figure 1(d)). The results shown in Figures 1(a) and 1(c) are consistent with the numerical results obtained in [9].

The nMGA may require a lot of time in updating the instantaneous event rates for all processes every time an event occurs in one of the  $N$  processes. The Laplace Gillespie algorithm avoids this rate recalculation, whereas it might be costly to calculate the gamma variates each time an event occurs. We compare numerically the computation times for the two algorithms by varying  $N$ . The other parameters remain the same as those used in Figures 1(a)–(d). For the Laplace Gillespie algorithm, we use a binary tree data structure to store and update  $\lambda_i$  ( $1 \leq i \leq N$ ) to accelerate the selection of the  $i$  value with probability  $\Pi_i$  upon the occurrence of each event [20]. In short, each  $\lambda_i$  occupies a leaf of the tree, and each non-leaf node stores the sum of its left child and its right child. This data structure is useful when only a small number of  $\lambda_i$  are changed following the occurrence of each event [20]. This is not the case for the nMGA, for which all the  $N$  instantaneous event rates must be updated upon each event. Therefore, for the nMGA, we use a simple linear search, which is computationally less expensive than updating a binary tree every time an event occurs. We use codes written in C++, compiled with a standard g++ compiler without an optimization option on a Mac Book Air with 1.7 GHz Intel Core i7 processor and 8GB 1600 MHz DDR3 RAM. The computation time in seconds plotted as a function of  $N$  in Figure 1(e) indicates that the Laplace Gillespie algorithm runs substantially faster than the nMGA as  $N$  increases.

Both the nMGA and the Laplace Gillespie algorithm require two uniformly distributed random variates per event, as does the standard Gillespie algorithm. In



**Fig. 1** Comparison between the nMGA and the Laplace Gillespie algorithm. The distribution of inter-event times is power law (12) with  $\kappa = 1$ . Among the  $N$  processes,  $\lfloor N/3 \rfloor$  processes are simulated with  $\alpha = 1$ , another  $\lfloor N/3 \rfloor$  processes with  $\alpha = 1.5$ , and another  $N - 2\lfloor N/3 \rfloor \approx N/3$  processes with  $\alpha = 2$ . (a)–(d) Survival function of inter-event times for one process with  $\alpha = 1$ , another with  $\alpha = 1.5$ , and another with  $\alpha = 2$ , from the top to the bottom. (a) nMGA when  $N = 10$ . (b) Laplace Gillespie algorithm when  $N = 10$ . (c) nMGA when  $N = 10^3$ . (d) Laplace Gillespie algorithm when  $N = 10^3$ . (e) Computation time as a function of the number of processes,  $N$ .

addition, for each generated event, the nMGA demands  $O(N)$  time to search for the process to fire and update the instantaneous event rates for all processes. In contrast, the Laplace Gillespie algorithm demands  $O(k \log N)$  time per event on average, where  $k$  is the typical number of processes that are affected by the firing of the  $i$ th process. The search for what process to fire requires  $O(\log N)$  time, given the binary tree data structure [20]. The updating of  $\lambda_j$  due to the event of the  $i$ th process requires  $O(k)$  time, including the generation of  $k$  gamma-distributed random variates. The updating of the binary tree requires  $O(k \log N)$  time because it consumes  $O(\log N)$  time

for each updated  $\lambda_j$ . Because  $k$  is usually much smaller than  $N$  in practice [20, 33], the Laplace Gillespie algorithm is expected to run faster than the nMGA, which is consistent with the numerical results shown in Figure 1(e).

Additionally, the Laplace Gillespie algorithm outperforms the nMGA in the sense that the Laplace Gillespie algorithm is exact for any  $N$ , whereas the nMGA is not. On the other hand, for the Laplace Gillespie algorithm, the form of  $\psi(\tau)$  is limited, whereas the nMGA allows for any  $\psi(\tau)$  to be used.

**6. Positively Correlated Sequences of Inter-Event Times.** We have considered renewal processes, i.e., stationary point processes without correlation between inter-event times. However, inter-event times are positively correlated for human activity and earthquakes [24, 52]. The Laplace Gillespie algorithm provides a method for generating point processes with positive correlation, without changing  $\psi(\tau)$ . To generate such event sequences, we redraw a new event rate for the Poisson process,  $\lambda_i$ , with probability  $1 - q$  ( $0 \leq q < 1$ ), when the  $i$ th process has generated an event. With probability  $q$ , we continue to use the same value of  $\lambda_i$  until the  $i$ th process generates another event. We call this algorithm the correlated Laplace Gillespie algorithm. The standard Laplace Gillespie algorithm is recovered when  $q = 0$ . The correlation between inter-event times grows as  $q$  increases. Although the same  $\lambda_i$  value may be used for generating consecutive inter-event times, the corresponding inter-event times are different because they are generated from a Poisson process. The computation time for the correlated Laplace Gillespie algorithm decreases as  $q$  increases because the number of times that  $\lambda_i$  is redrawn is proportional to  $1 - q$ .

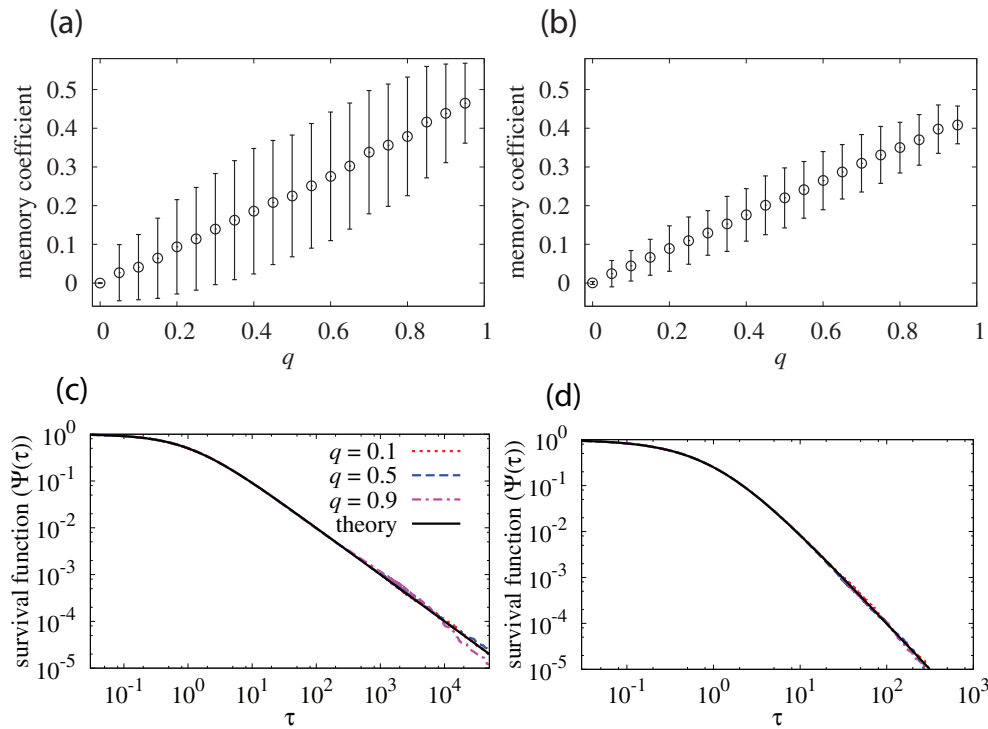
In a continuous-time Markov process with a state-dependent hopping rate, the inter-event time defined as the time between two consecutive hops, regardless of the state, is generally correlated across inter-event times [63]. The correlated Laplace Gillespie algorithm can be interpreted as a special case of this scenario such that the state is continuous, the process hops back to the current state with probability  $q$ , and it hops to any other state with probability proportional to  $(1 - q) \times p(\lambda)$ . The correlated Laplace Gillespie algorithm can be alternatively built on top of a finite-state [63] or an infinite-state [42] Markov process with a general transition probability between states. This variant of the correlated Laplace Gillespie algorithm is similar to a two-state cascading Poisson process in which the two states correspond to different event rates [47].

Two remarks are now in order. First,  $\psi(\tau)$  is independent of the  $q$  value. This is because the stationary density of the corresponding continuous-time Markov process in the  $\lambda$ -space is equal to  $p(\lambda)$ , irrespective of the  $q$  value. Of course, the distribution of  $\tau$  conditioned on the previous inter-event time,  $\tau'$ , is different from  $\psi(\tau)$  and depends on  $\tau'$  in general. Second, the correlated Laplace Gillespie algorithm cannot be used to generate correlated event sequences when  $\psi(\tau)$  is the exponential distribution. In this case, the event rate  $\lambda$  must be kept constant over time and therefore cannot be modulated in a temporally correlated manner.

We measure the so-called memory coefficient [24] to quantify the amount of correlation in a sequence of inter-event times generated by the correlated Laplace Gillespie algorithm. The memory coefficient for a sequence of inter-event times,  $\{\tau_1, \tau_2, \dots, \tau_n\}$ , where  $n$  is the number of inter-event times, is defined by

$$(36) \quad M = \frac{\sum_{i=1}^{n-1} (\tau_i - m_1)(\tau_{i+1} - m_2)}{\sqrt{\sum_{i=1}^{n-1} (\tau_i - m_1)^2 \sum_{i=2}^n (\tau_{i+1} - m_2)^2}},$$

where  $m_1 = \sum_{i=1}^{n-1} \tau_i / (n - 1)$  and  $m_2 = \sum_{i=2}^n \tau_i / (n - 1)$ .

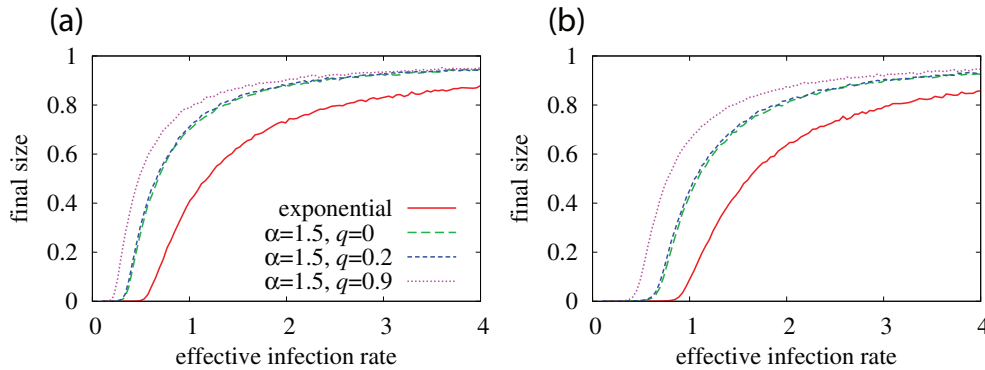


**Fig. 2** Memory coefficient,  $M$ , for the correlated Laplace Gillespie algorithm. We used the power-law distribution of inter-event times given by (12) with  $\kappa = 1$ . (a)  $\alpha = 1$ . (b)  $\alpha = 2$ . The error bar represents the mean  $\pm$  standard deviation. (c) Survival function of a single event sequence (i.e.,  $N = 1$ ) with  $10^6$  events with  $\alpha = 1$  and  $q = 0.1, 0.5, \text{ and } 0.9$ . (d) Similar to (c) with  $\alpha = 2$ .

For the power-law distribution of inter-event times given by (12) with  $\kappa = 1$ , we generate a sequence of  $n = 10^5$  inter-event times and calculate  $M$ . The mean and standard deviation of  $M$ , calculated on the basis of  $10^3$  sequences, are plotted for  $\alpha = 1$  and  $\alpha = 2$  in Figures 2(a) and 2(b), respectively. For both  $\alpha$  values,  $M$  monotonically increases with  $q$  and a range of  $M$  values between 0 and  $\approx 0.4$  is produced. In empirical data,  $M$  lies between 0 and 0.1 for human activity and between 0.1 and 0.25 for natural phenomena [24]. These ranges of  $M$  are produced using approximately  $0 \leq q \leq 0.2$  and  $0.2 \leq q \leq 0.5$ , respectively.

**7. Epidemic Processes.** Previous numerical efforts suggested that the dynamics of epidemic processes in well-mixed populations or networks were altered if contact events were generated by non-Poissonian renewal processes [34, 53, 61, 68]. The nMGA and the Laplace Gillespie algorithm can be used for implementing such models of epidemic processes. To demonstrate the use of the Laplace Gillespie algorithm, we simulate a node-centric susceptible-infected-recovered (SIR) epidemic process model, which is similar to previous models [57, 61, 62].

Consider a static network composed of  $N$  nodes. At any point in time, each node assumes one of the three states: susceptible, infected, or recovered. An infected node  $i$  transmits the disease to a susceptible node  $j$  upon the activation of link  $(i, j)$ . To activate links, we initially assign to each node  $i$  ( $1 \leq i \leq N$ ) an independent and identical point process whose probability density of inter-event times is given by  $\psi(\tau)$ .



**Fig. 3** Final outbreak size for the SIR epidemic model. (a) Well-mixed population. (b) Regular random graph with degree of each node equal to five. We set  $N = 10^4$ . For the power-law density of inter-event times, we use (12) with  $\kappa = 1$  and  $\alpha = 1.5$ .

When an event occurs at node  $i$ , we select a neighbor of  $i$ , denoted by  $j$ , with the equal probability and activate link  $(i, j)$ . If either  $i$  or  $j$  is infected and the other is susceptible, the disease is transmitted such that the susceptible node becomes infected. An infected node transits to the recovered state according to a Poisson process of rate  $\mu$ . A recovered node neither infects nor is infected by other nodes.

The mean time to node activation, which enables infection, is given by  $\langle \tau \rangle = \int_0^\infty \tau \psi(\tau) d\tau$ . The mean time for an infected node to recover is equal to  $1/\mu$ . We define the effective infection rate by  $\lambda_{\text{eff}} = (1/\mu)/\langle \tau \rangle$  [9]. We control  $\lambda_{\text{eff}}$  by changing  $\mu$  for a given  $\psi(\tau)$ . This definition is justified because multiplying  $\langle \tau \rangle$  and  $1/\mu$  by the same factor only changes the time scale of the dynamics.

We assume an equilibrium point process, i.e., we start simulations from the equilibrium state. This is equivalent to drawing the first event time for each node from the waiting-time distribution,  $\psi^w(\tau)$ , rather than from  $\psi(\tau)$ , and drawing subsequent event times from  $\psi(\tau)$ . The population structure is assumed to be either well mixed (i.e., complete graph) or the regular random graph in which all nodes have degree five and all links are randomly formed. In both cases, we set  $N = 10^4$ . Each simulation starts from the same initial condition, in which a particular node, which is the same in all simulations, is infected and all the other  $N - 1$  nodes are susceptible. We measure the number of recovered nodes at the end of the simulation normalized by  $N$ , called the final size, averaged over  $10^4$  simulations. We consider four renewal processes for node activation: the exponential distribution, corresponding to a Poisson process, and three power-law distributions given by (12) with  $\alpha = 1.5$ ,  $\kappa = 1$ , and  $q = 0, 0.2$ , and  $0.9$ .

The final size for the well-mixed population and the regular random graph is shown in Figures 3(a) and 3(b), respectively. For both population structures, and across the entire range of the effective infection rate,  $\lambda_{\text{eff}}$ , the final size is larger when  $\psi(\tau)$  is the power-law distribution than when it is the exponential distribution. Consistent with this result, the epidemic threshold, i.e., the value of  $\lambda_{\text{eff}}$  at which the final size becomes positive, is smaller for the power-law distribution  $\psi(\tau)$  than for the exponential distribution  $\psi(\tau)$ .

The final size is larger with positive correlation of inter-event times ( $q = 0.9$ ) than with no correlation ( $q = 0$ ). Results for  $q = 0.2$  are almost identical to those for  $q = 0$ . Because realistic values of the memory coefficient,  $M$ , for human activity are



produced by  $0 \leq q \leq 0.2$  (section 6), we conclude that a realistic amount of positive correlation in inter-event times does not affect the final size.

**8. Discussion.** We have provided a generalization of the Gillespie algorithm for non-Poissonian renewal processes, which we call the Laplace Gillespie algorithm. Our algorithm is exact for any number of processes running in parallel and is faster than the nMGA [9]. Although it is only applicable to renewal processes whose survival function is completely monotone, it applies to several renewal processes of interest. We have also proposed a method to simulate nonrenewal point processes with tunable positive correlation between inter-event times.

Previous studies numerically explored Poissonian explanations of long-tailed distributions of inter-event times. Examples include a nonhomogeneous Poisson process whose event rate switches between two values and is also periodically modulated [47]. Another example is a self-exciting Hawkes process with an exponential memory kernel [52]. We have shown that a power-law distribution of inter-event times (12) is generated when the rate of an event-modulated Poisson process is drawn from the gamma distribution upon the occurrence of every event. This observation provides a theoretical underpinning of the fact that nonhomogeneous Poisson processes and Hawkes processes can generate long-tailed distributions of inter-event times. In other words, switching between different rates is a general mechanism to produce long-tailed distributions of inter-event times. Although the present results indicate that, in theory, we require a mixture of an infinite number of Poisson processes of different rates to produce a power-law distribution, in practice a small number of Poisson processes may be sufficient. In fact, a mixture of a small number of exponential distributions is sometimes employed to fit empirical distributions of inter-event times [29, 35, 36, 58].

We have applied the Laplace Gillespie algorithm to an epidemic model in well-mixed and networked populations. The applicability of the Laplace Gillespie algorithm, as well as of the modified next reaction method [3] and of the nMGA, extends far beyond epidemic modeling. In fact, these algorithms can simulate systems of earthquakes, spiking neurons, financial time series, crimes, and so on (see section 1 for references). In particular, empirical data corresponding to these applications suggest long-tailed distributions of inter-event times (section 1), thus yielding a CV (coefficient of variation, i.e., the standard deviation divided by the mean) larger than unity, and therefore not excluding the use of the Laplace Gillespie algorithm. It is also straightforward to include births and deaths of nodes [6, 61] and links [32] of contact networks, or rewiring of links [28, 71], as long as these events obey renewal processes or nonrenewal point processes with positive correlation, as emulated by the correlated Laplace Gillespie algorithm.

The Laplace Gillespie algorithm can be employed if and only if the survival function of inter-event times is completely monotone. Some convenient conditions for and examples of survival functions that are not completely monotone are as follows:

(i) Nonmonotonicity. By setting  $n = 2$  in (9), we obtain  $d\psi(\tau)/d\tau \leq 0$ . Therefore,  $\psi(\tau)$  must monotonically decrease with  $\tau$  for the Laplace Gillespie algorithm to be applicable. This condition excludes the gamma and Weibull distributions with shape parameter  $\alpha > 1$ , any log-normal distribution, and any Pareto distribution, i.e.,

$$(37) \quad \psi(\tau) = \begin{cases} \frac{\alpha}{\tau_0} \left(\frac{\tau_0}{\tau}\right)^{\alpha+1} & (\tau \geq \tau_0), \\ 0 & (\tau < \tau_0), \end{cases}$$

where  $\alpha > 0$  and  $\tau_0 > 0$ .

(ii) CV is smaller than unity. Complete monotonicity implies that the CV of  $\tau$  is larger than or equal to unity [75]. This condition again excludes the gamma and Weibull distributions with  $\alpha > 1$ . In practice, a CV value less than unity indicates that events occur more regularly than in a Poisson process, which would yield  $CV = 1$ . Therefore, renewal processes producing relatively periodic event sequences are also excluded.

In epidemiology, evidence suggests that empirical recovery times are less dispersed than the exponential distribution, implying a CV value less than unity. Therefore, a gamma distribution with scale parameter  $\alpha > 1$  or even a delta distribution is often alternatively employed [45, 72]. These distributions cannot be simulated by our algorithm.

(iii) Higher-order conditions. Even when  $d\psi(\tau)/d\tau \leq 0$  and the CV is large,  $\Psi(\tau)$  may not be completely monotone. For example, the one-sided Cauchy distribution defined by  $\psi(\tau) = 1/[\pi(\tau^2 + 1)]$  yields  $d^2\psi(\tau)/d\tau^2 = 2(3\tau^2 - 1)/[\pi(\tau^2 + 1)^3]$ , whose sign depends on the value of  $\tau$ .

Empirical evidence of online correspondences of humans suggests that, except for very small  $\tau$  values,  $\tau$  obeys a power-law distribution for small  $\tau$  and an exponential distribution for large  $\tau$  [73]. Such a  $\psi(\tau)$  monotonically decreases with  $\tau$ , verifying that  $d\psi(\tau)/d\tau \leq 0$ . However, the sign of  $d^2\psi(\tau)/d\tau^2$  depends on the  $\tau$  value, such that the corresponding survival function is not completely monotone.

**Appendix A. Halving the Number of Random Variates Used in the Gillespie Algorithm.** Each step of the Gillespie algorithm usually requires two random variates,  $u_1$  and  $u_2$ , uniformly distributed on  $[0, 1]$ , one to draw the time increment via  $\Delta t = -\log u_1 / (\sum_{i=1}^N \lambda_i)$  and the other to select the process  $i$  that fires on the basis of (2). To eliminate one random variate, we first select  $i$  using random variate  $u_2$ . Independently of whether we use a binary or a linear search, we end up identifying the unique  $i$  value that satisfies  $\sum_{j=1}^{i-1} \Pi_j \leq u_2 < \sum_{j=1}^i \Pi_j$ . Once  $i$  is determined, we set  $u_1 \equiv (u_2 - \sum_{j=1}^{i-1} \Pi_j) / \Pi_i$ , which is in fact uniformly distributed on  $[0, 1]$ . Therefore, we do not have to draw  $u_1$  using a random number generator. This mathematical trick is similar to the one employed in the next reaction methods [3, 20].

**Appendix B. Derivation of the Exact Gillespie Algorithm for General Renewal Processes.** Consider  $N$  renewal processes running in parallel. If the  $i$ th process is running in isolation, the waiting time  $\tau$  until the next event is distributed according to

$$(38) \quad \psi_i^w(\tau|t_i) = \frac{\psi_i(t_i + \tau)}{\Psi_i(t_i)},$$

where  $\Psi_i(t_i)$  is the survival function of the  $i$ th process given by (3).

In fact, the  $i$ th process coexists with the other  $N - 1$  processes. We denote by  $\phi(\Delta t, i|\{t_j\})$  the probability density with which the  $i$ th process, but not the other  $N - 1$  processes, generates the next event in the set of  $N$  processes after time  $\Delta t$ , given the time since the previous event for each process,  $\{t_j\}$ , i.e.,  $t_1, \dots, t_N$ . We obtain

$$(39) \quad \phi(\Delta t, i|\{t_j\}) = \psi_i^w(\Delta t|t_i) \prod_{j=1; j \neq i}^N \Psi_j(\Delta t|t_j),$$

where

$$(40) \quad \Psi_j(\Delta t|t_j) = \int_{\Delta t}^{\infty} \psi_j^w(\tau'|t_j)d\tau' = \frac{\Psi_j(t_j + \Delta t)}{\Psi_j(t_j)}$$

is the probability that the time until the next event for the hypothetically isolated  $j$ th process is larger than  $\Delta t$ , conditioned on the assumption that the last event has occurred time  $t_j$  before. Using (38) and (40), we rewrite (39) as

$$(41) \quad \phi(\Delta t, i|\{t_j\}) = \frac{\psi_i(t_i + \Delta t)}{\Psi_i(t_i + \Delta t)}\Phi(\Delta t|\{t_j\}),$$

where  $\Phi(\Delta t|\{t_j\})$  is given by (4).

Equation (4) represents the probability that no process generates an event for time  $\Delta t$ . By equating this quantity to  $u$ , a random variate over the unit interval, we can determine  $\Delta t$ , i.e., the time until the next event in the entire population of the  $N$  renewal processes. Equation (41) implies that, once  $\Delta t$  is determined,  $\lambda_i(t_i + \Delta t) = \psi_i(t_i + \Delta t)/\Psi_i(t_i + \Delta t)$  is the instantaneous rate of the  $i$ th process and is proportional to the probability that the  $i$ th process generates this event. Therefore, the exact Gillespie algorithm for general renewal processes is produced as given in section 3.

#### REFERENCES

- [1] <http://www.know-all.net/articles/view/56> (in Russian). (Cited on p. 105)
- [2] J. ABATE AND W. WHITT, *The Fourier-series method for inverting transforms of probability distributions*, Queue. Syst., 10 (1992), pp. 5–87. (Cited on p. 104)
- [3] D. F. ANDERSON, *A modified next reaction method for simulating chemical systems with time dependent propensities and delays*, J. Chem. Phys., 127 (2007), art. 214107. (Cited on pp. 96, 110, 111)
- [4] R. BADDELEY, L. F. ABBOTT, M. C. A. BOOTH, F. SENGPIEL, T. FREEMAN, E. A. WAKEMAN, AND E. T. ROLLS, *Responses of neurons in primary and inferior temporal visual cortices to natural scenes*, Proc. R. Soc. Lond. B, 264 (1997), pp. 1775–1783. (Cited on p. 96)
- [5] P. BAK, K. CHRISTENSEN, L. DANON, AND T. SCANLON, *Unified scaling law for earthquakes*, Phys. Rev. Lett., 88 (2002), art. 178501. (Cited on p. 96)
- [6] S. BANSAL, J. READ, B. POURBOHLOUL, AND L. A. MEYERS, *The dynamic nature of contact networks in infectious disease epidemiology*, J. Biol. Dyn., 4 (2010), pp. 478–489. (Cited on p. 110)
- [7] A. L. BARABÁSI, *The origin of bursts and heavy tails in human dynamics*, Nature, 435 (2005), pp. 207–211. (Cited on p. 96)
- [8] C. BECK AND E. G. D. COHEN, *Superstatistics*, Phys. A, 322 (2003), pp. 267–275. (Cited on p. 101)
- [9] M. BOGUÑA, L. F. LAFUERZA, R. TORAL, AND M. A. SERRANO, *Simulating non-Markovian stochastic processes*, Phys. Rev. E, 90 (2014), art. 042108. (Cited on pp. 97, 98, 105, 109, 110)
- [10] C. G. BOWSHER, *Modelling security market events in continuous time: Intensity based, multivariate point process models*, J. Econom., 141 (2007), pp. 876–912. (Cited on p. 96)
- [11] T. CARLETTI AND A. FILISSETTI, *The stochastic evolution of a protocell: The Gillespie algorithm in a dynamically varying volume*, Comput. Math. Methods Med., 2012 (2012), art. 423627. (Cited on p. 96)
- [12] Á. CORRAL, *Long-term clustering, scaling, and universality in the temporal occurrence of earthquakes*, Phys. Rev. Lett., 92 (2004), art. 108501. (Cited on p. 96)
- [13] D. R. COX, *Renewal Theory*, Methuen & Co. Ltd, Frome, UK, 1962. (Cited on pp. 96, 105)
- [14] D. J. DALEY AND J. GANI, *Epidemic Modelling: An Introduction*, Cambridge University Press, Cambridge, UK, 1999. (Cited on p. 95)
- [15] W. FELLER, *An Introduction to Probability Theory and Its Applications, Volume II*, 2nd ed., John Wiley & Sons, New York, 1971. (Cited on pp. 96, 100, 102, 104, 105)

- [16] J. FERNÁNDEZ-GRACIA, V. M. EGUÍLUZ, AND M. SAN MIGUEL, *Update rules and interevent time distributions: Slow ordering versus no ordering in the voter model*, Phys. Rev. E, 84 (2011), art. 015103(R). (Cited on p. 96)
- [17] V. FILIMONOV AND D. SORNETTE, *Quantifying reflexivity in financial markets: Toward a prediction of flash crashes*, Phys. Rev. E, 85 (2012), art. 056108. (Cited on p. 96)
- [18] W. FISCHER AND K. MEIER-HELLSTERN, *The Markov-modulated Poisson process (MMPP) cookbook*, Perf. Eval., 18 (1992), pp. 149–171. (Cited on p. 99)
- [19] N. GEORGIU, I. Z. KISS, AND E. SCALAS, *Solvable non-Markovian dynamic network*, Phys. Rev. E, 92 (2015), art. 042801. (Cited on p. 103)
- [20] M. A. GIBSON AND J. BRUCK, *Efficient exact stochastic simulation of chemical systems with many species and many channels*, J. Phys. Chem. A, 104 (2000), pp. 1876–1889. (Cited on pp. 96, 105, 106, 107, 111)
- [21] D. T. GILLESPIE, *A general method for numerically simulating the stochastic time evolution of coupled chemical reactions*, J. Comput. Phys., 22 (1976), pp. 403–434. (Cited on pp. 96, 97)
- [22] D. T. GILLESPIE, *Exact stochastic simulation of coupled chemical reactions*, J. Phys. Chem., 81 (1977), pp. 2340–2361. (Cited on pp. 96, 97)
- [23] L. J. GLEESER, *The gamma distribution as a mixture of exponential distributions*, Am. Stat., 43 (1989), pp. 115–117. (Cited on p. 102)
- [24] K. I. GOH AND A. L. BARABÁSI, *Burstiness and memory in complex systems*, Europhys. Lett., 81 (2008), art. 48002. (Cited on pp. 96, 97, 107, 108)
- [25] R. GORENFLO, A. A. KILBAS, F. MAINARDI, AND S. V. ROGOSIN, *Mittag-Leffler Functions, Related Topics and Applications*, Springer-Verlag, Heidelberg, 2014. (Cited on p. 103)
- [26] R. GORENFLO AND F. MAINARDI, *Fractional calculus: Integral and differential equations of fractional order*, in Fractals and Fractional Calculus in Continuum Mechanics, A. Carpinteri and F. Mainardi, eds., Springer-Verlag, Berlin, 1997, pp. 223–276. (Cited on p. 103)
- [27] J. GRANDELL, *Mixed Poisson Processes*, Chapman & Hall, London, 1997. (Cited on pp. 99, 101, 102)
- [28] T. GROSS AND B. BLASIUS, *Adaptive coevolutionary networks: A review*, J. R. Soc. Interface, 5 (2008), pp. 259–271. (Cited on p. 110)
- [29] J. J. HECKMAN, R. ROBB, AND J. R. WALKER, *Testing the mixture of exponentials hypothesis and estimating the mixing distribution by the method of moments*, J. Amer. Statist. Assoc., 85 (1990), pp. 582–589. (Cited on pp. 104, 110)
- [30] A. HELMSTETTER AND D. SORNETTE, *Subcritical and supercritical regimes in epidemic models of earthquake aftershocks*, J. Geophys. Res., 107 (2002), art. 2237. (Cited on p. 96)
- [31] C. A. HIDALGO R., *Conditions for the emergence of scaling in the inter-event time of uncorrelated and seasonal systems*, Phys. A, 369 (2006), pp. 877–883. (Cited on pp. 101, 102)
- [32] P. HOLME AND F. LILJEROS, *Birth and death of links control disease spreading in empirical contact networks*, Sci. Rep., 4 (2014), art. 4999. (Cited on p. 110)
- [33] P. HOLME AND J. SARAMÁKI, *Temporal networks*, Phys. Rep., 519 (2012), pp. 97–125. (Cited on pp. 96, 104, 107)
- [34] D. X. HORVÁTH AND J. KERTÉSZ, *Spreading dynamics on networks: The role of burstiness, topology and non-stationarity*, New J. Phys., 16 (2014), art. 073037. (Cited on p. 108)
- [35] N. P. JEWELL, *Mixtures of exponential distributions*, Ann. Stat., 10 (1982), pp. 479–484. (Cited on pp. 102, 104, 110)
- [36] T. JIN AND L. S. GONIGUNTA, *Exponential approximation to Weibull renewal with decreasing failure rate*, J. Stat. Comput. Simul., 80 (2010), pp. 273–285. (Cited on pp. 102, 104, 110)
- [37] S. D. JOHNSON, K. J. BOWERS, AND K. PEASE, *Towards the modest predictability of daily burglary counts*, Policing, 6 (2012), pp. 167–176. (Cited on p. 96)
- [38] A. F. KARR, *Point Processes and their Statistical Inference (second edition)*, Marcel Dekker, New York, 1991. (Cited on p. 99)
- [39] D. G. KENDALL, *An artificial realization of a simple “birth-and-death” process*, J. R. Stat. Soc. Ser. B, 12 (1950), pp. 116–119. (Cited on pp. 96, 97)
- [40] J. F. C. KINGMAN, *On doubly stochastic Poisson processes*, Math. Proc. Camb. Phil. Soc., 60 (1964), pp. 923–930. (Cited on p. 99)
- [41] M. KIVELÄ AND M. A. PORTER, *Estimating interevent time distributions from finite observation periods in communication networks*, Phys. Rev. E, 92 (2015), art. 052813. (Cited on p. 104)
- [42] J. KLEINBERG, *Bursty and hierarchical structure in streams*, Data Min. Knowl. Disc., 7 (2003), pp. 373–397. (Cited on pp. 104, 107)
- [43] Z. KURTH-NELSON AND A. D. REDISH, *Temporal-difference reinforcement learning with distributed representations*, PLOS ONE, 4 (2009), art. e7362. (Cited on p. 101)

- [44] J. Q. LI AND A. R. BARRON, *Mixture density estimation*, in Advances in Neural Information Processing Systems, Vol. 12, S. A. Solla, T. K. Leen, and K. R. Müller, eds., MIT Press, Cambridge, MA, 2000, pp. 279–285. (Cited on p. 104)
- [45] A. L. LLOYD, *Destabilization of epidemic models with the inclusion of realistic distributions of infectious periods*, Proc. R. Soc. Lond. B, 268 (2001), pp. 985–993. (Cited on p. 111)
- [46] T. LU, D. VOLFSOHN, L. TSIMRING, AND J. HASTY, *Cellular growth and division in the Gillespie algorithm*, Syst. Biol., 1 (2004), pp. 121–128. (Cited on p. 96)
- [47] R. D. MALMGREN, D. B. STOUFFER, A. E. MOTTER, AND L. A. N. AMARAL, *A Poissonian explanation for heavy tails in e-mail communication*, Proc. Natl. Acad. Sci. USA, 105 (2008), pp. 18153–18158. (Cited on pp. 107, 110)
- [48] G. MARSAGLIA AND W. W. TSANG, *A simple method for generating gamma variables*, ACM Trans. Math. Software, 26 (2000), pp. 363–372. (Cited on p. 105)
- [49] J. MASOLIVER, M. MONTERO, AND G. H. WEISS, *Continuous-time random-walk model for financial distributions*, Phys. Rev. E, 67 (2003), art. 021112. (Cited on p. 96)
- [50] N. MASUDA AND P. HOLME, *Predicting and controlling infectious disease epidemics using temporal networks*, F1000Prime Rep., 5 (2013), art. 6. (Cited on pp. 96, 104)
- [51] N. MASUDA AND R. LAMBIOTTE, *A Guide to Temporal Networks*, World Scientific, Singapore, 2016. (Cited on p. 105)
- [52] N. MASUDA, T. TAKAGUCHI, N. SATO, AND K. YANO, *Self-exciting point process modeling of conversation event sequences*, in Temporal Networks, P. Holme and J. Saramäki, eds., Springer-Verlag, Berlin, 2013, pp. 245–264. (Cited on pp. 107, 110)
- [53] B. MIN, K. I. GOH, AND I. M. KIM, *Suppression of epidemic outbreaks with heavy-tailed contact dynamics*, Europhys. Lett., 103 (2013), art. 50002. (Cited on p. 108)
- [54] G. O. MOHLER, M. B. SHORT, P. J. BRANTINGHAM, F. P. SCHOENBERG, AND G. E. TITA, *Self-exciting point process modeling of crime*, J. Amer. Statist. Assoc., 106 (2011), pp. 100–108. (Cited on p. 96)
- [55] Y. OGATA, *Seismicity analysis through point-process modeling: A review*, Pure Appl. Geophys., 155 (1999), pp. 471–507. (Cited on p. 96)
- [56] V. PERNICE, B. STAUDE, S. CARDANOBILO, AND S. ROTTER, *How structure determines correlations in neuronal networks*, PLOS Comput. Biol., 7 (2011), art. e1002059. (Cited on p. 96)
- [57] N. PERRA, B. GONÇALVES, R. PASTOR-SATORRAS, AND A. VESPIGNANI, *Activity driven modeling of time varying networks*, Sci. Rep., 2 (2012), art. 469. (Cited on p. 108)
- [58] M. POLITI AND E. SCALAS, *Activity spectrum from waiting-time distribution*, Phys. A, 383 (2007), pp. 43–48. (Cited on pp. 104, 110)
- [59] M. D. PORTER AND G. WHITE, *Self-exciting hurdle models for terrorist activity*, Ann. Appl. Stat., 6 (2012), pp. 106–124. (Cited on p. 96)
- [60] M. RABERTO, E. SCALAS, AND F. MAINARDI, *Waiting-times and returns in high-frequency financial data: An empirical study*, Phys. A, 314 (2002), pp. 749–755. (Cited on p. 96)
- [61] L. E. C. ROCHA AND V. D. BLONDEL, *Bursts of vertex activation and epidemics in evolving networks*, PLOS Comput. Biol., 9 (2013), art. e1002974. (Cited on pp. 108, 110)
- [62] L. E. C. ROCHA, F. LILJEROS, AND P. HOLME, *Simulated epidemics in an empirical spatiotemporal network of 50,185 sexual contacts*, PLOS Comput. Biol., 7 (2011), art. e1001109. (Cited on p. 108)
- [63] T. SCHWALGER AND B. LINDNER, *Theory for serial correlations of interevent intervals*, Eur. Phys. J. Spec. Topics, 187 (2010), pp. 211–221. (Cited on p. 107)
- [64] W. R. SOFTKY AND C. KOCH, *The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs*, J. Neurosci., 13 (1993), pp. 334–350. (Cited on p. 96)
- [65] T. TAKAGUCHI AND N. MASUDA, *Voter model with non-Poissonian inter-event intervals*, Phys. Rev. E, 84 (2011), art. 036115. (Cited on p. 96)
- [66] W. TRUCCOLO, U. T. EDEN, M. R. FELLOWS, J. P. DONOGHUE, AND E. N. BROWN, *A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects*, J. Neurophysiol., 93 (2005), pp. 1074–1089. (Cited on p. 96)
- [67] N. G. VAN KAMPEN, *Stochastic Processes in Physics and Chemistry*, 3rd ed., Elsevier, Amsterdam, The Netherlands, 2007. (Cited on p. 95)
- [68] P. VAN MIEGHEM AND R. VAN DE BOVENKAMP, *Non-Markovian infection spread dramatically alters the susceptible-infected-susceptible epidemic threshold in networks*, Phys. Rev. Lett., 110 (2013), art. 108701. (Cited on p. 108)
- [69] A. VÁZQUEZ, J. G. OLIVEIRA, Z. DEZSÖ, K. I. GOH, I. KONDOR, AND A. L. BARABÁSI, *Modeling bursts and heavy tails in human dynamics*, Phys. Rev. E, 73 (2006), art. 036127. (Cited on p. 96)

- [70] C. L. VESTERGAARD AND M. GÉNOIS, *Temporal Gillespie algorithm: Fast simulation of contagion processes on time-varying networks*, PLOS Comput. Biol., 11 (2015), art. e1004579. (Cited on pp. 96, 97, 98, 104)
- [71] E. VOLZ AND L. A. MEYERS, *Susceptible-infected-recovered epidemics in dynamic contact networks*, Proc. R. Soc. B, 274 (2007), pp. 2925–2933. (Cited on p. 110)
- [72] H. J. WEARING, P. ROHANI, AND M. J. KEELING, *Appropriate models for the management of infectious diseases*, PLOS Med., 2 (2005), art. e174. (Cited on p. 111)
- [73] Y. WU, C. ZHOU, J. XIAO, J. KURTHS, AND H. J. SCHELLNHUBER, *Evidence for a bimodal distribution in human communication*, Proc. Natl. Acad. Sci. USA, 107 (2010), pp. 18803–18808. (Cited on p. 111)
- [74] N. YANNAROS, *On Cox processes and gamma renewal processes*, J. Appl. Prob., 25 (1988), pp. 423–427. (Cited on p. 102)
- [75] N. YANNAROS, *Weibull renewal processes*, Ann. Inst. Stat. Math., 46 (1994), pp. 641–648. (Cited on pp. 102, 111)