A Gillespie algorithm for non-Markovian stochastic processes: Laplace transform approach

Naoki Masuda[1,*] and Luis E. C. Rocha[2,3]

[1] Department of Engineering Mathematics, University of Bristol, Bristol, UK.
[2] Department of Mathematics and naXys, Université de Namur, Namur, Belgium.
[3] Department of Public Health Sciences, Karolinska Institutet, Stockholm, Sweden.

[*] Corresponding author (naoki.masuda@bristol.ac.uk)

January 8, 2016

Abstract

The Gillespie algorithm provides statistically exact methods to simulate stochastic dynamics modelled as interacting sequences of discrete events including systems of biochemical reactions or earthquakes, networks of queuing processes or spiking neurons, and epidemic and opinion formation processes on social networks. Empirically, inter-event times of various human activities, in particular human communication, and some natural phenomena are often distributed according to long-tailed distributions. The Gillespie algorithm and its extant variants either assume the Poisson process, which produces exponentially distributed inter-event times, not long-tailed distributions, assume particular functional forms for time courses of the event rate, or works 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. It

uses the fact that a class of point processes is represented as a mixture of Poisson processes with different event rates. The method allows renewal processes whose survival function of inter-event times is completely monotone functions and works faster than a recently proposed Gillespie algorithm for general renewal processes. We also propose a method to generate sequences of event times with a given distribution of inter-event times and a tunable amount of positive correlation between inter-event times. We demonstrate our algorithm with exact simulations of epidemic processes on networks. We find that positive correlation in inter-event times modulates dynamics but in a quantitatively minor way with the amount of positive correlation comparable with empirical data.

1 Introduction

Social, biological, chemical, neural, seismological, and financial dynamics, among others, are often driven by time-stamped discrete events. For example, an individual human or animal transmits an infectious disease to another only when a contact event between the two individuals happens. The state transition in such a system, e.g., whether an individual is infected or not, can be modelled as being event-driven. Another example is chemical substances in which a chemical reaction event changes the number of reagents in a discrete manner in both time and state. Stochastic point processes are central tools for emulating these phenomena [1–3] and also find applications in operations research domains such as queuing systems and reliability analysis [4]. The most central point process model is the Poisson process, which assumes that events independently occur at a constant rate throughout time.

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 a different rate is attached to one reaction. In epidemic processes taking place on human or animal contact networks, each Poisson process is assigned to an individual or a link, which potentially transmits

the infection. The event rate of some of the Poisson processes may change upon the occurrence of a reaction or infection in the entire system. The simplest simulation method is to discretize time and judge whether an event occurs or not in each time window for individual processes. This widely used method, called the rejection method, is sub-optimal because the size of the time window must be sufficiently small for high accuracy, which is computationally costly [5]. The Gillespie algorithm is an efficient and statistically exact algorithm for such an interacting population of Poisson processes [6–8]. The Gillespie algorithm, or particularly the direct method of Gillespie [7,8], exploits the fact that superposition of independent Poisson processes is a single Poisson process whose event rate is the summation of those of the constituent Poisson processes. Using this mathematical property, only a single Poisson process needs to be emulated in the Gillespie algorithm (section 2).

Empirical data obtained from various domains suggest that real-life event sequences are far from those generated by the Poisson process. In particular, several empirical distributions of inter-event times obey long-tailed distributions [9–13], whereas the Poisson process generates the exponential distribution of inter-event times. This is a prominent example of non-Markovian event sequences, that is, one needs the history of events to know the statistics of the time of the next event. This is not the case for the Poisson process, which is memoryless. When inter-event times are independently generated from a given distribution, the point process is called the renewal process [14].

Efficient and accurate simulations of interacting renewal processes contribute to the understanding of effects of long-tailed behaviour of inter-event times on various dynamics in well-mixed and networked populations [13]. One may draw the next event time for all processes from the predetermined distributions, select the process that has generated the minimum waiting time to the next event, execute the event, draw the next event times for the affected processes, and repeat. This is so-called the Gillespie's first reaction method [7], whose improved versions

are the next reaction method [15] and the modified next reaction method [16]. In fact, for non-Markovian renewal processes, it is generally difficult to numerically solve equations for determining the next event time although these methods call for generation of fewer random numbers than the Gillespie algorithm [5]. In the following, we restrict ourselves to the Gillespie algorithm and its variants.

Motivated by situations of chemical reactions, many extensions of the Gillespie's algorithm to the case of non-Markovian processes assume that the dynamical change in the event rate is exogenously driven in particular functional forms [17, 18]. They are not applicable to non-Markovian renewal processes characterised by long-tailed distributions of inter-event times because such an exogeneous drive would be different across the renewal processes running in parallel and would not have a desirable functional form. Boguñá and colleagues extended the Gillespie algorithm to be applicable to general renewal processes [19] (section 3; also see [5] for further developments). However, the algorithm has practical limitations. First, it is not accurate when the number of ongoing renewal processes is small [19], thus affecting the beginning or ending of the dynamics of epidemic and opinion formation models, in which only a small number of processes is active even in large well-mixed or networked populations [5]. Second, it is necessary to recalculate the instantaneous event rate of each process upon every event in the entire population, a procedure that can be computationally costly.

In the present study, we propose an innovative Gillespie algorithm, the Laplace Gillespie algorithm, applicable to general 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 [19]. To demostrate the generalizability of the algorithm, we introduce a method to generate event sequences with positive correlation in temporally close inter-event times, as typically observed in human behaviour and natural phenomena [12], for a given distribution of inter-event times. We also demostrate our methods by performing exact

simulations of an epidemic process in which inter-event times follow a power-law distribution.

2 Gillespie algorithm

The original Gillespie algorithm [6–8] assumes N independent Poisson processes with rate λ_i $(1 \leq i \leq N)$ running in parallel. Because of the independence of different Poisson processes, superposition of the N processes is also a Poisson process with rate $N\langle\lambda\rangle \equiv \sum_{i=1}^{N} \lambda_i$. Therefore, we first draw Δt, an increment in time to the next event of the superposed Poisson process, from the exponential distribution given by

$$\phi(\Delta t) = N\langle\lambda\rangle e^{-N\langle\lambda\rangle\Delta t}. \tag{1}$$

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

$$\Pi_i = \frac{\lambda_i}{N\langle\lambda\rangle}. \tag{2}$$

Third, we advance the time by Δt and repeat this procedure. Upon an event, any λ_i is allowed 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 time for the ith process $(1 \leq i \leq N)$. If the process is Poisson, 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 (nMPA) [19]. The nMGA works as follows.

To determine the time of the next event, we track the time since the last event for each process, denoted by t_i. If the ith process were running in isolation, the waiting time τ until the next event would be distributed according to

$$\psi_i^{\mathrm{w}}(\tau|t_i) = \frac{\psi_i(t_i+\tau)}{\Psi_i(t_i)}, \tag{3}$$

where

$$\Psi_i(t_i) = \int_{t_i}^{\infty} \psi_i(\tau')\mathrm{d}\tau' \tag{4}$$

is the survival function, i.e., probability that the inter-event time is larger than t_i.

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

$$\phi(\Delta t, i|\{t_j\}) = \psi_i^{\mathrm{w}}(\Delta t|t_i) \prod_{j=1;j\neq i}^{N} \Psi_j(\Delta t|t_j), \tag{5}$$

where

$$\Psi_j(\Delta t|t_j) = \int_{\Delta t}^{\infty} \psi_j^{\mathrm{w}}(\tau'|t_j)\mathrm{d}\tau' = \frac{\Psi_j(t_j+\Delta t)}{\Psi_j(t_j)} \tag{6}$$

is the probability that the time to the next event for the hypothetically isolated jth process is larger than τ conditioned that the last event occurred at time t_j before. Using Eqs. (3) and (6), we rewrite Eq. (5) as

$$\phi(\Delta t, i|\{t_j\}) = \frac{\psi_i(t_i+\Delta t)}{\Psi_i(t_i+\Delta t)}\Phi(\Delta t|\{t_j\}), \tag{7}$$

where

$$\Phi(\Delta t|\{t_j\}) = \prod_{j=1}^{N} \frac{\Psi_j(t_j+\Delta t)}{\Psi_j(t_j)}. \tag{8}$$

Equation (8) represents the probability that no process generates an event for another time Δt. By equating this quantity to u, a random variate on the unit interval, we can determine Δt,

i.e., the time to the next event in the entire population of the N renewal processes. Equation (7) implies that, once Δt is determined, $\lambda_i(t_i+\Delta t) \equiv \psi_i(t_i+\Delta t)/\Psi_i(t_i+\Delta t)$ is the instantaneous rate of the ith process and proportional to the probability that the ith process generates this event. Therefore, the exact Gillespie algorithm for general renewal processes is given as follows:

1. Initialise t_j $(1 \le j \le N)$ for all j (for example, $t_j = 0$).

2. Draw the time to the next event, Δt, by solving $\Phi(\Delta t|\{t_j\}) = u$, where u is a random variate uniformly distributed on $[0,1]$.

3. Select the process i that has generated the event with probability

$$\Pi_i \equiv \frac{\lambda_i(t_i+\Delta t)}{\sum_{j=1}^{N}\lambda_j(t_j+\Delta t)}. \tag{9}$$

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

5. Repeat steps 2–4.

Although this algorithm is statistically exact, step 2 is time-consuming [5, 19]. To improve performance, Boguñá and colleagues introduced the nMGA. The nMGA is an approximation to the aforementioned algorithm and exact in the limit of $N \to \infty$. When Δt is small, which would be the case when N is large, Eq. (8) is approximated as

$$\begin{aligned}
\Phi(\Delta t|\{t_j\}) &= \exp\left[-\sum_{j=1}^{N}\ln\frac{\Psi_j(t_j)}{\Psi_j(t_j+\Delta t)}\right] \\
&= \exp\left[-\sum_{j=1}^{N}\ln\frac{\Psi_j(t_j)}{\Psi_j(t_j)-\psi_j(t_j)\Delta t+O(\Delta t^2)}\right] \\
&\approx \exp\left[-\Delta t N\overline{\lambda}(\{t_j\})\right],
\end{aligned} \tag{10}$$

where

$$\overline{\lambda}(\{t_j\}) = \frac{\sum_{j=1}^{N}\lambda_j(t_j)}{N} = \frac{1}{N}\sum_{j=1}^{N}\frac{\psi_j(t_j)}{\Psi_j(t_j)}. \tag{11}$$

With this approximation, the time to the next event is determined from $\Phi(\Delta t|\{t_j\}) \approx \exp\left[-\Delta t N\overline{\lambda}(\{t_j\})\right]$ u, i.e., $\Delta t = -\ln u/N\overline{\lambda}(\{t_j\})$. The process that generates the event is determined by setting $\Delta t = 0$ in Eq. (9). For the Poisson process, we set $\lambda_i(t_i) = \lambda_i$ to recover the original Gillespie algorithm (Eqs. (1) and (2)).

4 Laplace Gillespie algorithm

4.1 Algorithm

In the nMGA, we update the instantaneous event rates for allthe processes and its summation $\overline{\lambda}(\{t_j\})$ in Eq. (11) upon the occurrence of each event. This is because t_j $(1 \leq j \leq N)$ is updated upon an event. This procedure is time consuming when N is large; we have to update individual instantaneous rates even if the probability density of the inter-event time for a 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, which we call the event-modulated Poisson process. When an event occurs, we first draw the rate of the Poisson process, denoted by λ, according to a fixed probability density function $p(\lambda)$. Then, we draw the time of the next event according to the Poisson process with rate λ. Upon the next event, we renew the rate λ by redrawing it from $p(\lambda)$. We repeat these procedures.

This model, the event-modulated Poisson process, is a mixture of Poisson processes with different rates. It is a non-Poissonian renewal process in general. It is slightly different from the mixed Poisson process, in which a single rate is drawn from a random ensemble in the beginning and used throughout a realisation [20, 21]. It is also different from doubly stochastic Poisson process (also called Cox process), in which the rate of the Poisson process is a stochastic process [20–22], or its subclass called the Markov-modulated Poisson process, in which the event

rate switches in time according to a Markov process [23]. In these processes, the dynamics of the event rate are independent of the occurrence of events. In contrast, the event rate changes upon the occurrence of events in the event-modulated Poisson process.

The event-modulated Poisson process is a Poisson process when conditioned on the current value of λ. 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 ith process, denoted by λ_i, for all i ($1 \leq i \leq N$). This property enables us to construct a Gillespie algorithm, similar to the original one. 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 term the Laplace Gillespie algorithm (the reason for Laplace will be clear in section 4.2; it has a 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 ith process by $p_i(\lambda_i)$. The Laplace Gillespie algorithm proceeds as follows:

1. Initialise each of the N processes by drawing the rate λ_i ($1 \leq i \leq N$) according to the respective density function $p_i(\lambda_i)$.

2. Draw the time to the next event $\Delta t = -\ln u / \sum_{j=1}^{N} \lambda_j$, where u is the random variate uniformly distributed on $[0, 1]$.

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 λ_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 upon 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 neighbour), modify $p_j(\lambda_j)$ accordingly and draw a new rate λ_j from the modified $p_j(\lambda_j)$. The rate of the remaining processes is unchanged.

5. Repeat steps 2–4.

4.2 Theory

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

$$\psi(\tau) = \int_0^\infty p(\lambda)\lambda e^{-\lambda\tau}\mathrm{d}\lambda. \tag{12}$$

Integration of both sides of Eq. (12) gives the survival function of the inter-event time as follows:

$$\Psi(\tau) = \int_\tau^\infty \psi(\tau')\mathrm{d}\tau' = \int_0^\infty p(\lambda)e^{-\lambda\tau}\mathrm{d}\lambda. \tag{13}$$

Equation (13) 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 non-negative values. Although this statement can be made more rigorous if we replace $p(\lambda)\mathrm{d}\lambda$ by the probability distribution function, we use the probability density representation for simplicity.

A necessary and sufficient condition for the existence of $p(\lambda)$ is that $\Psi(\tau)$ is a completely monotone and $\Psi(0) = 1$ [24]. The complete monotonicity is defined by

$$(-1)^n \frac{\mathrm{d}\Psi(\tau)}{\mathrm{d}\tau} \geq 0 \quad (\lambda \geq 0, n = 0, 1, \ldots). \tag{14}$$

Condition $\Psi(0) = 1$ is satisfied by any survival function. With $n = 0$, Eq. (14) reads $\Psi(\tau) \geq 0$, which all survival functions satisfy. With $n = 1$, Eq. (14) reads $\psi(\tau) \geq 0$, which is also always satisfied. Equation (14) with $n \geq 2$ imposes nontrivial conditions.

4.3 Examples

In this section we show examples of distribution of inter-event times $\psi(\tau)$ for which the Laplace Gillespie algorithm can be used. These examples are summarised in Table 1.

- Exponential distribution

  The Poisson process with rate λ_0, i.e., $\psi(\tau) = \lambda_0 e^{-\lambda_0 \tau}$, is trivially produced by $p(\lambda) = \delta(\lambda - \lambda_0)$, where δ is the delta function.

- Power-law distribution

  Consider the case in which $p(\lambda)$ is the gamma distribution given by

  $$p(\lambda) = \frac{\lambda^{\alpha-1} e^{-\lambda/\kappa}}{\Gamma(\alpha)\kappa^{\alpha}}, \tag{15}$$

  where $\Gamma(\alpha)$ is the gamma function, α is the shape parameter of the gamma distribution, and κ is the scale parameter of the gamma distribution. With $\alpha = 1$, $p(\lambda)$ is reduced to the exponential distribution. By combining Eqs. (13) and (15), we obtain

  $$\Psi(\tau) = \frac{1}{(1+\kappa\tau)^{\alpha}}. \tag{16}$$

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

  $$\psi(\tau) = \frac{\kappa}{(1+\kappa\tau)^{\alpha+1}}. \tag{17}$$

  The same mathematical procedure has been used in a reinforcement learning model for generating discount rates that decay as a power law in time [25].

- Power-law distribution with an exponential tail

  Consider a shifted gamma distribution [21] given by

  $$p(\lambda) = \begin{cases} \frac{(\lambda-\lambda_0)^{\alpha-1} e^{-(\lambda-\lambda_0)/\kappa}}{\Gamma(\alpha)} & (\lambda \geq \lambda_0), \\ 0 & (0 < \lambda < \lambda_0), \end{cases} \tag{18}$$

where λ_0 is a constant. By combining Eqs. (13) and (18), we obtain

$$\Psi(\tau) = \frac{e^{-\lambda_0 \tau}}{(1+\kappa\tau)^{\alpha}}. \tag{19}$$

By differentiating Eq. (19), we obtain a power-law distribution with an exponential tail given by

$$\psi(\tau) = \frac{e^{-\lambda_0 \tau}}{(1+\kappa\tau)^{\alpha}} \left(\lambda_0 + \frac{\kappa\alpha}{1+\kappa\tau} \right). \tag{20}$$

- Weibull distribution

  The Weibull distribution is defined by

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

  which yields

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

  The Weibull distribution with $\alpha = 1$ is the 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 is expressed as the Laplace transform of a $p(\lambda)$ if and only if $0 < \alpha \leq 1$ [26, 27]. The distribution when $\alpha = 1/2$ is the so-called stable distribution of order $1/2$, for which we obtain [21, 24, 26]

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

  For other values of α ($0 < \alpha < 1/2$, $1/2 < \alpha < 1$), the explicit form of $p(\lambda)$ is complicated [26] such that it is impractical to use the Laplace Gillespie algorithm. For these α values, a mixture of a small number of exponential distributions may resemble the Weibull distribution [28] such that we may be able to use $p(\lambda)$ with point masses at some discrete values of λ to approximate the Weibull distribution of inter-event times.

- Gamma distribution

  When the inter-event time obeys the gamma distribution, i.e.,

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

  $\Psi(\tau)$ is expressed as the Laplace transform of a probability density function $p(\lambda)$ if and only if $0 < \alpha \leq 1$ [29,30]. We obtain [30]

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

- Mittag-Leffler distribution

  Consider the distribution of inter-event times defined in terms of the survival function given by

$$\Psi(\tau) = E_{\beta}(-\tau^{\beta}), \tag{26}$$

  where

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

  is the so-called Mittag-Leffler function. When $0 < \beta < 1$, $\Psi(\tau)$ is completely monotone, and we obtain [31]

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

  When $\beta = 1$, Eqs. (26) and (27) imply $\Psi(\tau) = e^{-\lambda\tau}$, yielding the Poisson process. When $0 < \beta < 1$, $\Psi(\tau)$ is long-tailed with the following asymptotics [31,32]:

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

  or equivalently,

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

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

- Integral of a valid survival function

For a survival function $\Psi(\tau)$ corresponding to the probability density of inter-event times $\psi(\tau)$, consider

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

Function $\Psi^{\mathrm{w}}(\tau)$ 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 the use of the Laplace Gillespie algorithm. Because $\Psi^{\mathrm{w}}(\tau) \geq 0$ $(\tau \geq 0)$, $\mathrm{d}^n\Psi^{\mathrm{w}}(\tau)/\mathrm{d}\tau^n = -\left[\mathrm{d}^{n-1}\Psi(\tau)/\mathrm{d}\tau^{n-1}\right] / \int_0^{\infty} \Psi(\tau')\mathrm{d}\tau'$ $(n = 1, 2, \ldots)$, and $\Psi(\tau)$ is completely monotone, $\Psi^{\mathrm{w}}(\tau)$ is completely monotone. In addition, Eq. (31) implies $\Psi^{\mathrm{w}}(0) = 1$. Therefore, the renewal process with survival function $\Psi^{\mathrm{w}}(\tau)$ can be also simulated by the Laplace Gillespie algorithm.

The corresponding probability density of the inter-event time is given by

$$\psi^{\mathrm{w}}(\tau) = -\frac{\mathrm{d}\Psi^{\mathrm{w}}(\tau)}{\mathrm{d}\tau} = \frac{\Psi(\tau)}{\langle\tau\rangle_{\psi}}, \tag{32}$$

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

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

and

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

Therefore, in each update in the Laplace Gillespie algorithm with the density of inter-event times $\psi^{\mathrm{w}}(\tau)$, the rate of the Poisson process λ should be sampled according to

density $p^{\mathrm{w}}(\lambda)$, where

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

For example, if $\psi(\tau)$ is the exponential distribution, $\psi^{\mathrm{w}}(\tau)$ is the exponential distribution with the same rate. If $\psi(\tau)$ is the power-law distribution given by Eq. (17), $\psi^{\mathrm{w}}(\tau)$ is the same form of the power-law distribution with α replaced by $\alpha - 1$.

- Product of valid survival functions

  The product of two completely monotone functions $\Psi_1(\tau)$ and $\Psi_2(\tau)$ are completely monotone [24]. In addition, $\Psi_1(0)\Psi_2(0) = 1$ if $\Psi_1(0) = \Psi_2(0) = 1$. Therefore, survival function $\Psi(\tau) \equiv \Psi_1(\tau)\Psi_2(\tau)$ admits the Laplace Gillespie algorithm if $\Psi_1(\tau)$ and $\Psi_2(\tau)$ do. The probability density of the rate will be the convolution of $p_1(\lambda)$ and $p_2(\lambda)$, where $\Psi_i(\tau) = \int_0^\infty p_i(\lambda) e^{-\lambda\tau} \mathrm{d}\lambda$ $(i = 1, 2)$.

4.4 Limitations

The Laplace Gillespie algorithm cannot be used if and only if the survival function of the inter-event time is not completely monotone. In this section, we present some convenient conditions and examples that fit in this class of survival functions.

- Non-monotone

  By setting $n = 2$ in Eq. (14), we obtain $\mathrm{d}\psi(\tau)/\mathrm{d}\tau \leq 0$. Therefore, $\psi(\tau)$ must monotonically decrease with τ 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.,

$$\psi(\tau) = \begin{cases} \frac{\alpha}{\tau_0} \left(\frac{\tau_0}{\tau}\right)^{\alpha+1} & (\tau \geq \tau_0), \\ 0 & (\tau < \tau_0), \end{cases} \tag{36}$$

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

- Coefficient of variation smaller than unity

  The complete monotonicity implies that the coefficient of variation (CV), i.e., standard deviation divided by the mean, of τ is larger than or equal to unity [27]. This condition excludes the gamma and Weibull distributions with $\alpha > 1$. In practice, a CV value smaller than unity indicates that events occur more regularly than in the case of the Poisson process, which yields CV $= 1$. Therefore, renewal processes producing relatively periodic event sequences are also excluded.

- Higher-order conditions

  Even if $\mathrm{d}\psi(\tau)/\mathrm{d}\tau \leq 0$ and the CV is large, the survival function of a common distribution may not be completely monotone. For example, the one-sided Cauchy distribution defined by $\psi(\tau) = 1/\left[\pi(\tau^2+1)^2\right]$ yields $\mathrm{d}^2\psi(\tau)/\mathrm{d}\tau^2 = 4(3\tau^2-1)/(\tau^2+1)^3$, whose sign depends on the value of τ.

  Empirical evidence of online correspondences of humans suggests bimodal $\psi(\tau)$ in the sense that, excluding very small τ, it obeys a power-law distribution for small τ and an exponential distribution for large τ [33]. Such a $\psi(\tau)$ monotonically decreases with τ, verifying $\mathrm{d}\psi(\tau)/\mathrm{d}\tau \leq 0$. However, the sign of $\mathrm{d}^2\psi(\tau)/\mathrm{d}\tau^2$ depends on the τ value such that the corresponding survival function is not completely monotone.

4.5 Empirical distributions of inter-event times

We are often interested in informing population dynamics described as interacting stochastic point processes, such as epidemic processes, by empirical data of event sequences. The standard numerical method emulates the dynamics on top of empirical event sequences, i.e., using a given list of events with the event type and time stamp [13, 34, 35]. Another approach would be to estimate $\psi(\tau)$ from empirical data and use a variant of the Gillespie algorithm to simulate

stochastic processes.

The nMGA [19] and the modified next reaction method [16] allow arbitrary $\psi(\tau)$ including the empirical distribution. In contrast, 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 cost [36,37]. Because an empirical distribution is likely not completely monotone, we propose two alternative methods for empirical data. The first method is to fit a completely monotone survival function of inter-event times, such as Eq. (16), to given data. The second method is to estimate a mixture of exponential distributions with different rates to approximate the empirical $\psi(\tau)$ or $\Psi(\tau)$. Likelihood or other cost-function methods are available for the estimation [26,28,38–40]. The approximation error is guaranteed to decay inversely proportionally to the number of constituent distributions [41].

4.6 Initial conditions

When we start running N processes in parallel, we may initially draw the inter-event time for each process from $\psi(\tau)$. This initial condition defines the so-called ordinary renewal process [14]. An alternative model, called the equilibrium renewal process, assumes that the process has started at time $t = -\infty$ such that the first inter-event time for each process, drawn at $t = 0$, uses the distribution of waiting times to the next event in the equilibrium, not $\psi(\tau)$ (i.e., distribution of inter-event times) [14]. The probability density of the waiting time to the next event is given by Eq. (32). To simulate the equilibrium renewal process, we start by drawing the rate of the Poisson process according to $p^{\mathrm{w}}(\lambda)$ given by Eq. (35). Afterwards, we draw the rate according to $p(\lambda)$.

5 Numerical performance

In this section, we compare the performance of the nMGA [19] and the Laplace Gillespie algorithm. We use the power-law distribution of inter-event times given by Eq. (17). Because κ only controls the scale of the inter-event time, we set $\kappa = 1$ without loss of generality. To generate gamma-distributed random variates, we use a popular algorithm based on the rejection method [42]. In fact, we adapt an open source code [43] to our purposes. We generate $N/3$ processes by Eq. (17) with $\alpha = 1$, another $N/3$ processes with $\alpha = 1.5$, and the remaining processes with $\alpha = 2$. We continue the simulation until any of the N processes generates 10^6 inter-event times for the first time. We assume the ordinary renewal process such that the initial inter-event time for each process is drawn from $\psi(\tau)$.

The survival function for one process with $\alpha = 1$, one with $\alpha = 1.5$, and another with $\alpha = 2$ is shown by the solid curves for the nMGA and the Laplace Gillespie algorithm in Figs. 1(a) and 1(b), respectively, with $N = 10$. The theoretical survival function, Eq. (16), is plotted by the dashed curves. The results obtained from the Laplace Gillespie algorithm (Fig. 1(b)) are more accurate than those obtained from the nMGA. This is because the nMGA is exact in the limit of $N \to \infty$, whereas the Laplace Gillespie algorithm is exact for any N. When $N = 10^3$, the nMGA is sufficiently accurate (Fig. 1(c)), as is the Laplace Gillespie algorithm (Fig. 1(d)). The results shown in Figs. 1(a) and 1(c) are consistent with the numerical results obtained in Ref. [19].

The nMGA may require a lot of time in calculating the instantaneous event rate ($\lambda_j(0)$ in Eq. (11)) for all processes every time an event occurs in one of the N processes. The Laplace Gillespie algorithm avoid the rate recalculation whereas it may be costly to calculate the gamma variate each time an event occurs. We numerically compare the computation time for the two algorithms by varying N. The other parameters are the same as those used in Fig. 1. We do

not optimise the method to select the i value with probability Π_i at the occurrence of each event; we use the simple linear search for both the nMGA and Laplace Gillespie algorithm. In the Laplace Gillespie algorithm, we use a binary tree data structure to store and update λ_i $(1 \leq i \leq N)$ to accelerate the algorithm. This structure is useful when only a small number of λ_i is changed upon each event. This is not the case for the nMGA. We use codes written in C++, compiled with a standard g++ compiler without an optimisation option on a Mac Book Air with 1.7 GHz Intel Core i7 and 8Gz 1600 MHz DDR3. The computation time in secs plotted against N in Fig. 1(e) indicates that the Laplace Gillespie algorithm is substantially faster than the nMGA as N increases.

The Laplace Gillespie algorithm outperforms the nMGA in that the Laplace Gillespie algorithm is exact for any N and generally runs faster than the nMGA. The price paid is that the form of $\psi(\tau)$ is limited, whereas the nMGA allows any $\psi(\tau)$.

6 Applications

6.1 Positively correlated sequences of inter-event times

We have considered the renewal processes, i.e., stationary point processes without correlation between different inter-event times. However, inter-event times are positively correlated, albeit weakly, in a majority of data recorded from human activity [12, 44]. This and other types of correlation in sequences of event times is known to affect upshots of epidemic processes [34, 45–47].

The Laplace Gillespie algorithm provides a succinct 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, λ_i, with probability $1 - q$ $(0 \leq q < 1)$, when the ith process has generated an event. With probability q, we continue to use the same value of

λ_i until the ith process generates the next event. The standard Laplace Gillespie algorithm is recovered with $q = 0$. The correlation between inter-event times is expected to grow as q increases. Although the same λ_i value may be used for generating consecutive inter-event times, the corresponding inter-event times are different because they are generated from the Poisson process. The computation time decreases as q increases because the number of times one has to redraw λ_i is proportional to $1 - q$.

In the 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 [48]. The present algorithm can be interpreted as a special case of this general framework such that the state is continuous, the process hops to the current state with probability $1 - q$, and it hops to any other state with the probability proportional to $q \times p(\lambda)$. The correlated Laplace Gillespie algorithm can be alternatively built on top of a finite-state [48] or infinite-state [39] Markov process with a general transition probability between states. This variant of the model will be similar to a two-state cascading Poisson process that assumes transitions between a normal and excited states, with the excited state having a higher event rate than the normal state [49].

Two remarks are in order. First, $\psi(\tau)$ is independent of the p value. This is because the stationary density of the corresponding continuous-time Markov process in the λ-space is equal to $p(\lambda)$ irrespectively of the q value. Second, this algorithm cannot be used to generate correlated event sequences when $\psi(\tau)$ is the exponential distribution. In this case, the event rate λ must be kept constant throughout the time and therefore cannot be modulated in a temporally correlated manner.

We measure the so-called memory coefficient [12] to quantify the amount of correlation in a sequence of inter-event times. The memory coefficient for a sequence of inter-event times,

$\{\tau_1, \tau_2, \ldots, \tau_n\}$, where n is the number of inter-event times, is defined by

$$M = \frac{\sum_{i=1}^{n-1}(\tau_i - m_1)(\tau_{i+1} - m_2)}{\sqrt{\sum_{i=1}^{n-1}(\tau_i - m_1)^2 \sum_{i=2}^{n}(\tau_{i+1} - m_2)^2}}, \tag{37}$$

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

For the power-law distribution of inter-event times given by Eq. (17) 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 Figs. 2(a) and 2(b), respectively. For both α values, M monotonically increases with p, and a range of M between 0 and ≈ 0.4 is produced. In empirical data, M is between 0 and 0.1 for human activities and 0.1 and 0.25 for natural phenomena [12]. These ranges of M are produced with approximately $0 \leq q \leq 0.2$ and $0.2 \leq q \leq 0.5$, respectively.

6.2 Epidemic processes

Previous numerical efforts suggested that dynamics of epidemic processes in well-mixed populations or networks were altered if contact events were generated by renewal processes different from the Poisson process [50–53]. The nMGA and Laplace Gillespie algorithm can be used for efficiently implementing such models of epidemic processes. To demonstrate the use of the Laplace Gillespie algorithm, we model node-centric susceptible-infected-recovered (SIR) epidemic spreading, which is similar to previous models [47, 50, 54].

Consider a population of N nodes. At any point of time, each node assumes one of the three states, susceptible, infected, or recovered. An infected node is assumed to infect a susceptible node upon a contact. An infected node transits to the recovered state according to the Poisson process with rate μ. A recovered node neither infects nor is infected by other nodes. Regarding infection events, each node i ($1 \leq i \leq N$) is driven by an independent and identical point process whose probability density of inter-event times is given by $\psi(\tau)$. It should be noted that

a similar approach could be link-centric, that is, the activation of each link follws independent and idential point processes. When an event occurs, node i is instantaneously activated and contacts a randomly selected node j. In the case of a well-mixed population, each j is selected with probability $1/(N-1)$. In the case of a network, which is fixed over time, j is selected from the neighbours of node i with the equal probability. If either i or j is infected and the other is susceptible, the infection is transmitted such that the susceptible node becomes infected. Then, node i waits for another time τ drawn from $\psi(\tau)$ before the next activation.

The mean time to the node activation, which enables infection, is given by $\langle\tau\rangle = \int_0^\infty \tau\psi(\tau)\mathrm{d}\tau$. The mean time for an infected node to recover is equal to $1/\mu$. We define the effective infection rate by $\lambda_{\text{eff}} = (1/\mu)/\langle\tau\rangle$ [19]. We control λ_{eff} by changing μ 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 the equilibrium point process, i.e., start the simulation from the equilibrium state. This is equivalent to drawing the first event time for each node from the waiting-time distribution, $\psi^{\text{w}}(\tau)$, not from $\psi(\tau)$, and subsequent event times from $\psi(\tau)$. The population structure is assumed to be either well-mixed 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, called the final size, normalised by N and averaged over 10^4 simulations. We consider four point processes for node activation, i.e., the exponential distribution, corresponding to the Poisson process, and three power-law distributions given by Eq. (17) 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 Figs. 3(a) and 3(b), respectively. For both population structures and across the entire range of

the effective infection rate, λ_{eff}, the final size is larger when $\psi(\tau)$ is the power-law than when $\psi(\tau)$ is the exponential distribution. Consistent with this result, the epidemic threshold, i.e., the value of λ_{eff} at which the final size becomes positive, is smaller for the power-law $\psi(\tau)$ than the exponential $\psi(\tau)$.

The final size is larger with positive correlation of inter-event times ($q = 0.9$) than with no correlation ($q = 0$). The results for $q = 0.2$ are almost the same as those for $q = 0$. Because realistic values of the memory coefficient, M, are produced by $0 \leq q \leq 0.2$ (section 6.1), we conclude that a realistic amount of positive correlation in inter-event times does not affect the final size.

7 Discussion

We provided a generalisation of the Gillespie algorithm for non-Poissonian renewal processes, called the Laplacian Gillespie algorithm. Our algorithm is exact for any number of processes running in parallel and faster than the nMGA [19]. Although it is only applicable to the renewal processes whose survival function is completely monotone, it covers several renewal processes of interest. We also proposed a method to simulate non-renewal point processes with an arbitrary distribution of inter-event times and positive correlation between inter-event times on the same event sequence.

We 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 that of the modified next reaction method [16] and the nMGA, is much wider. These algorithms can simulate systems of spiking neurons, earthquakes, financial time series, and so on. Inter-event times of these systems are often distributed according to distributions whose CV (i.e., coefficient of variation, i.e., standard deviation divided by the mean) is larger than unity [13], therefore not excluding the use of the Laplace Gillespie algorithm. It is also straightforward to allow births

and deaths of nodes [50,55] and links [56] of contact networks, or rewiring of links [57,58], if the occurrence of these events obeys renewal processes or our non-renewal processes with positive correlation. In more realistic scenarios, the recovery events do not have to obey the Poisson process, as assumed in our numerical simulations. The Laplace Gillespie algorithm has some limitations. Evidence suggests that empirical recovery times are less dispersed than the exponential distribution, implying the CV value less than unity. Therefore, the gamma distribution with scale parameter $\alpha > 1$ or even the delta distribution are often employed in numerical and theoretical analysis [59–61]. In contrast, the Laplace Gillespie algorithm is only applicable to distributions of inter-event times whose CV is larger than unity.

Previous studies aimed at Poissonian explanation of long-tailed distributions of inter-event times. Examples include a non-homogeneous Poisson process with switching between two event rates plus an additional periodic modulation of the rate [49] and the self-exciting Hawkes process with an exponential memory kernel [44]. We showed that the power-law distribution of inter-event times, Eq. (17), was produced when the rate of the Poisson process is drawn from the gamma distribution. This observation provides a theoretical underpinning of the fact that modulated Poisson processes [44,49] may generate power-law distributed inter-event times. Although we theoretically need an infinite number of Poisson processes to fully produce a power-law distribution, a mixture of a small number of Poisson processes may be practically sufficient. In fact, a mixture of a small number of exponential distributions is sometimes employed to fit empirical distributions of inter-event times [26, 28, 38, 40].

Acknowledgments

N.M. acknowledges the support provided through 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.

References

[1] Y Ogata. Seismicity analysis through point-process modeling: A review. *Pure Appl. Geophys.*, 155:471–507, 1999.

[2] N. G. Van Kampen. *Stochastic Processes in Physics and Chemistry*. Elsevier, Netherlands, Amsterdam, 3rd edition, 2007.

[3] F. Gabbiani and S. J. Cox. *Mathematics for Neuroscientists*. Academic Press, Amsterdam, 2010.

[4] M. Jacobsen. *Point Processes Theory and Applications (second edition)*. Birkhäuser, Boston, MA, 2006.

[5] C. L. Vestergaard and M. Génois. Temporal Gillespie algorithm: Fast simulation of contagion processes on time-varying networks. *PLOS Comput. Biol.*, 11:e1004579, 2015.

[6] D. G. Kendall. An artificial realization of a simple "birth-and-death" process. *J. R. Stat. Soc. Ser. B*, 12:116–119, 1950.

[7] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Comput. Phys.*, 22:403–434, 1976.

[8] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, 81:2340–2361, 1977.

[9] J. P. Eckmann, E. Moses, and D. Sergi. Entropy of dialogues creates coherent structures in e-mail traffic. *Proc. Natl. Acad. Sci. USA*, 101:14333–14337, 2004.

[10] A. L. Barabási. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207–211, 2005.

[11] 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*, 73:036127, 2006.

[12] K. I. Goh and A. L. Barabási. Burstiness and memory in complex systems. *Europhys. Lett.*, 81:48002, 2008.

[13] P. Holme and J. Saramäki. Temporal networks. *Phys. Rep.*, 519:97–125, 2012.

[14] D. R. Cox. *Renewal Theory.* Methuen & Co. Ltd, Frome, UK, 1962.

[15] M. A. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A*, 104:1876–1889, 2000.

[16] D. F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127:214107, 2007.

[17] T. Lu, D. Volfson, L. Tsimring, and J. Hasty. Cellular growth and division in the Gillespie algorithm. *Syst. Biol.*, 1:121–128, 2004.

[18] T. Carletti and A. Filisetti. The stochastic evolution of a protocell: The Gillespie algorithm in a dynamically varying volume. *Comput. Math. Methods Med.*, 2012:423627, 2012.

[19] M. Boguñá, L. F. Lafuerza, R. Toral, and M. Á. Serrano. Simulating non-Markovian stochastic processes. *Phys. Rev. E*, 90:042108, 2014.

[20] A. F. Karr. *Point Processes and their Statistical Inference (second edition).* CRC Press, New York, NY, 1991.

[21] J. Grandell. *Mixed Poisson Processes.* Chapman & Hall, London, 1997.

[22] J. F. C. Kingman. On doubly stochastic Poisson processes. *Math. Proc. Camb. Phil. Soc.*, 60:923–930, 1964.

[23] W. Fischer and K. Meier-Hellstern. The Markov-modulated Poisson process (MMPP) cookbook. *Perf. Eval.*, 18:149–171, 1992.

[24] W. Feller. *An Introduction to Probability Theory and its Applications, Volume II, Second Edition.* John Wiley & Sons, 1971.

[25] Z. Kurth-Nelson and A. D. Redish. Temporal-difference reinforcement learning with distributed representations. *PLOS ONE*, 4:e7362, 2009.

[26] N. P. Jewell. Mixtures of exponential distributions. *Ann. Stat.*, 10:479–484, 1982.

[27] N. Yannaros. Weibull renewal processes. *Ann. Inst. Stat. Math.*, 46:641–648, 1994.

[28] T. Jin and L. S. Gonigunta. Exponential approximation to Weibull renewal with decreasing failure rate. *J. Stat. Comput. Simul.*, 80:273–285, 2010.

[29] N. Yannaros. On Cox processes and gamma renewal processes. *J. Appl. Prob.*, 25:423–427, 1988.

[30] L. J. Gleser. The gamma distribution as a mixture of exponential distributions. *Am. Stat.*, 43:115–117, 1989.

[31] R. Gorenflo and F. Mainardi. Fractional calculus: Integral and differential equations of fractional order. In A. Carpinteri and F. Mainardi, editors, *Fractals and Fractional Calculus in Continuum Mechanics*, pages 223–276. Springer, Berlin, 1997.

[32] N. Georgiou, I. Z. Kiss, and E. Scalas. Solvable non-markovian dynamic network. *Phys. Rev. E*, 92:042801, 2015.

[33] 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*, 107:18803–18808, 2010.

[34] N. Masuda and P. Holme. Predicting and controlling infectious disease epidemics using temporal networks. *F1000Prime Reports*, 5:6, 2013.

[35] P. Holme. Modern temporal network theory: A colloquium. *Eur. Phys. J. B*, 88:234, 2015.

[36] J. Abate and W. Whitt. The Fourier-series method for inverting transforms of probability distributions. *Queue. Syst.*, 10:5–87, 1992.

[37] J. Abate and W. Whitt. Numerical inversion of Laplace transforms of probability distributions. *ORSA J. Comput.*, 7:36–43, 1995.

[38] 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. Am. Stat. Assoc.*, 85:582–589, 1990.

[39] J. Kleinberg. Bursty and hierarchical structure in streams. *Data Min. Knowl. Disc.*, 7:373–397, 2003.

[40] M. Politi and E. Scalas. Activity spectrum from waiting-time distribution. *Physica A*, 383:43–48, 2007.

[41] J. Q. Li and A. R. Barron. Mixture density estimation. In S. A. Solla, T. K. Leen, and K. R Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 279–285, Cambridge, MA, 2000. MIT Press.

[42] G. Marsaglia and W. W. Tsang. A simple method for generating gamma variables. *ACM Trans. Math. Software*, 26:363–372, 2000.

[43] http://www.know-all.net/articles/view/56.

[44] N. Masuda, T. Takaguchi, N. Sato, and K. Yano. Self-exciting point process modeling of conversation event sequences. In P. Holme and J. Saramäki, editors, *Temporal Networks*, pages 245–264. Springer-Verlag, Berlin, 2013.

[45] M. Karsai, M. Kivelä, R. K. Pan, K. Kaski, J. Kertész, A. L. Barabási, and J. Saramäki. Small but slow world: How network topology and burstiness slow down spreading. *Phys. Rev. E*, 83:025102(R), 2011.

[46] G. Miritello, E. Moro, and R. Lara. Dynamical strength of social ties in information spreading. *Phys. Rev. E*, 83:045102(R), 2011.

[47] 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:e1001109, 2011.

[48] T. Schwalger and B. Lindner. Theory for serial correlations of interevent intervals. *Eur. Phys. J. Spec. Topics*, 187:211–221, 2010.

[49] 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*, 105:18153–18158, 2008.

[50] L. E. C. Rocha and V. D. Blondel. Bursts of vertex activation and epidemics in evolving networks. *PLOS Comput. Biol.*, 9:e1002974, 2013.

[51] 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.*, 110:108701, 2013.

[52] B. Min, K. I. Goh, and I. M. Kim. Suppression of epidemic outbreaks with heavy-tailed contact dynamics. *EPL*, 103:50002, 2013.

[53] D. X. Horváth and J. Kertész. Spreading dynamics on networks: The role of burstiness, topology and non-stationarity. *New J. Phys.*, 16:073037, 2014.

[54] N. Perra, B. Gonçalves, R. Pastor-Satorras, and A. Vespignani. Activity driven modeling of time varying networks. *Sci. Rep.*, 2:469, 2012.

[55] S. Bansal, J. Read, B. Pourbohloul, and L. A. Meyers. The dynamic nature of contact networks in infectious disease epidemiology. *J. Biol. Dyn.*, 4:478–489, 2010.

[56] P. Holme and F. Liljeros. Birth and death of links control disease spreading in empirical contact networks. *Sci. Rep.*, 4:4999, 2014.

[57] E. Volz and L. A. Meyers. Susceptible-infected-recovered epidemics in dynamic contact networks. *Proc. R. Soc. B*, 274:2925–2933, 2007.

[58] T. Gross and B. Blasius. Adaptive coevolutionary networks: A review. *J. R. Soc. Interface*, 5:259–271, 2008.

[59] A. L. Lloyd. Destabilization of epidemic models with the inclusion of realistic distributions of infectious periods. *Proc. R. Soc. Lond. B*, 268:985–993, 2001.

[60] H. J. Wearing, P. Rohani, and M. J. Keeling. Appropriate models for the management of infectious diseases. *PLOS Med.*, 2:e174, 2005.

[61] J. M. Heffernan and L. M. Wahl. Improving estimates of the basic reproductive ratio: Using both the mean and the dispersal of transition times. *Theor. Popul. Biol.*, 70:135–145, 2006.

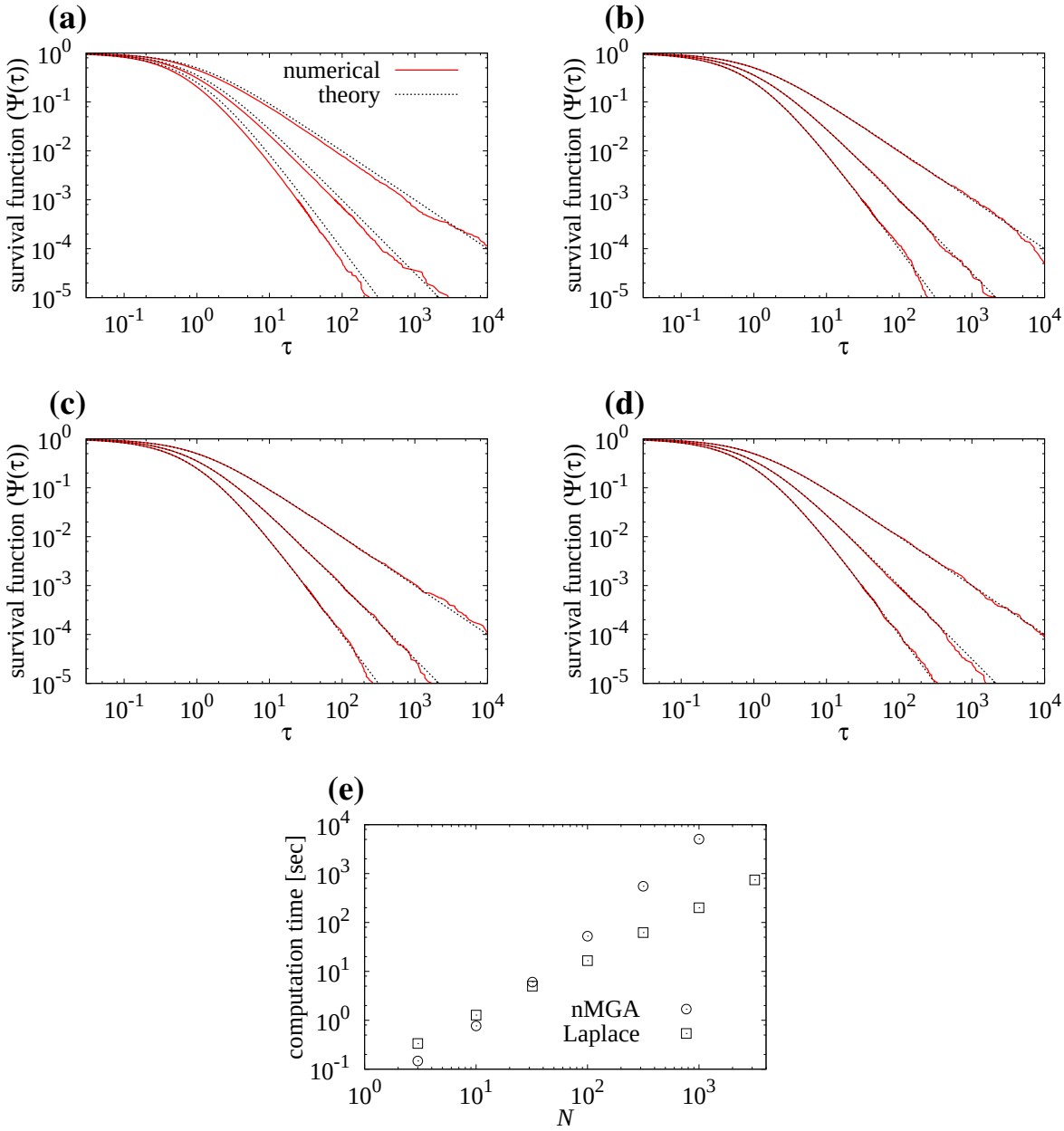

Figure 1: Comparison between the nMGA and the Laplace Gillespie algorithm. The distribution of inter-event times is power law (Eq. (17)) with $\kappa = 1$. Among the N processes, $N/3$ processes age simulated with $\alpha = 1$, another $N/3$ processes with $\alpha = 1.5$, and the other $N/3$ processes 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 with $N = 10$. (b) Laplace Gillespie algorithm with $N = 10$. (c) nMGA with $N = 10^3$. (d) Laplace Gillespie algorithm with $N = 10^3$. (e) Computation time as a function of the number of processes, N.

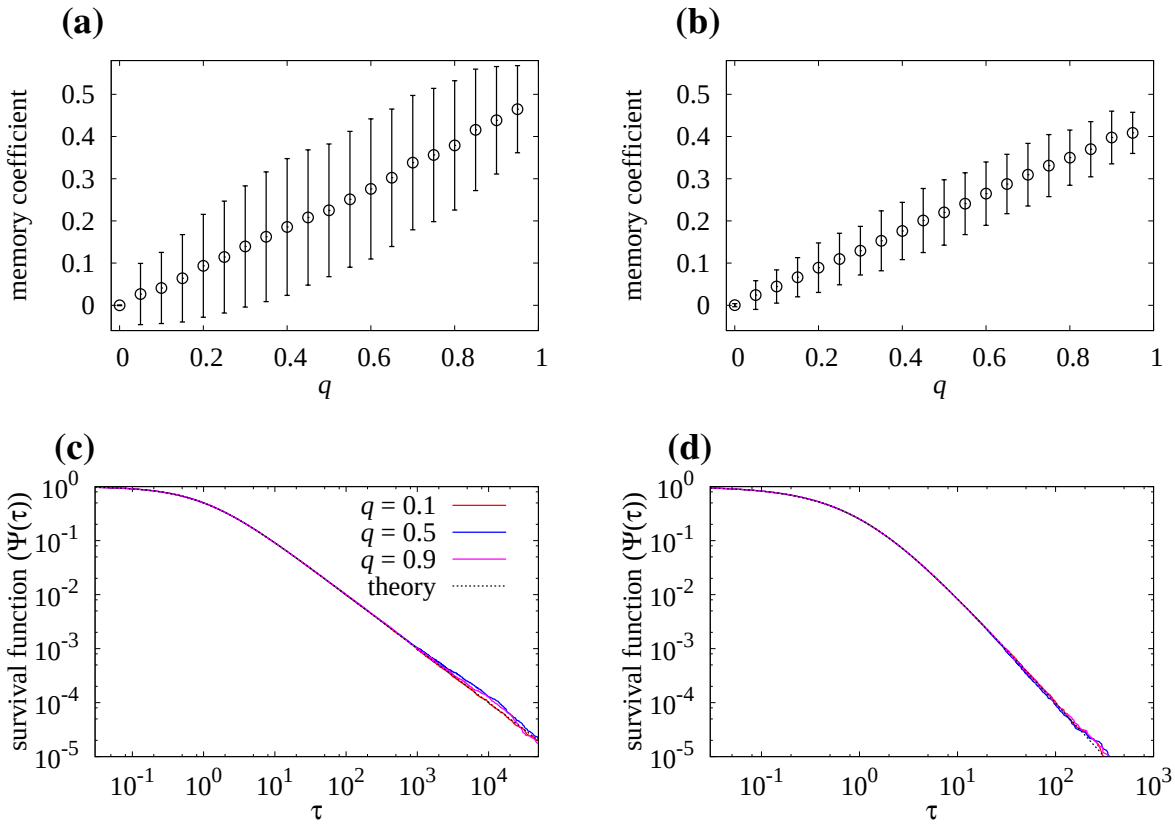

Figure 2: Memory coefficient, M, as a function of q, for the Laplace Gillespie algorithm. We used the power-law distribution of inter-event times given by Eq. (17) with $\kappa = 1$. (a) $\alpha = 1$. (b) $\alpha = 2$. The error bar represents the mean $\pm$ standard deviation. (c) Survival function of a single event sequence (i.e., $N = 1$) with 10^6 events with $\alpha = 1$ and $q = 0.1$, 0.5, and 0.9. (d) Same for $\alpha = 2$.

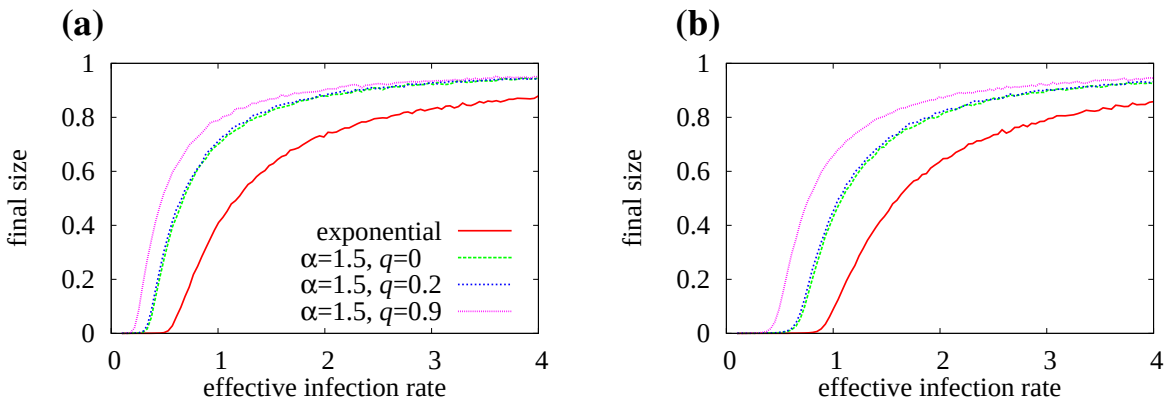

Figure 3: Final outbreak size for the SIR epidemic model. (a) Well-mixed population. (b) Regular random graph with degree of each node equal to five. We set $N = 10^4$ nodes. For the power-law density of inter-event times, we use Eq. (17) with $\kappa = 1$ and $\alpha = 1.5$.

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	$p(\lambda)$
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	$\frac{e^{-\lambda_0\tau}}{(1+\kappa\tau)^\alpha}\left(\lambda_0 + \frac{\kappa\alpha}{1+\kappa\tau}\right)$	$\frac{e^{-\lambda_0\tau}}{(1+\kappa\tau)^\alpha}$		$\frac{(\lambda-\lambda_0)^{\alpha-1} e^{-(\lambda-\lambda_0)/\kappa} H(\lambda-\lambda_0)}{\Gamma(\alpha)}$
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	$\approx \frac{\beta \sin(\beta\pi)\Gamma(\beta)}{\pi\tau^{\beta+1}}$	$E_\beta(-\tau^\beta)$	$0 < \beta < 1$	$\frac{1}{\pi}\frac{\lambda^{\beta-1}\sin(\beta\pi)}{\lambda^{2\beta}+2\lambda^\beta\cos(\beta\pi)+1}$