
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Jo, Hang Hyun; Lee, Byoung Hwa; Hiraoka, Takayuki; Jung, Woo Sung
Copula-based algorithm for generating bursty time series

Published in:
Physical Review E

DOI:
[10.1103/PhysRevE.100.022307](https://doi.org/10.1103/PhysRevE.100.022307)

Published: 14/08/2019

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Jo, H. H., Lee, B. H., Hiraoka, T., & Jung, W. S. (2019). Copula-based algorithm for generating bursty time series. *Physical Review E*, 100(2), 1-6. [022307]. <https://doi.org/10.1103/PhysRevE.100.022307>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Copula-based algorithm for generating bursty time series

Hang-Hyun Jo^{1,2,3,*}, Byoung-Hwa Lee,^{2,1} Takayuki Hiraoka,¹ and Woo-Sung Jung^{2,4,1,†}

¹Asia Pacific Center for Theoretical Physics, Pohang 37673, Republic of Korea

²Department of Physics, Pohang University of Science and Technology, Pohang 37673, Republic of Korea

³Department of Computer Science, Aalto University, Espoo FI-00076, Finland

⁴Department of Industrial and Management Engineering, Pohang University of Science and Technology, Pohang 37673, Republic of Korea



(Received 18 April 2019; published 14 August 2019)

Dynamical processes in various natural and social phenomena have been described by a series of events or event sequences showing non-Poissonian, bursty temporal patterns. Temporal correlations in such bursty time series can be understood not only by heterogeneous interevent times (IETs) but also by correlations between IETs. Modeling and simulating various dynamical processes requires us to generate event sequences with a heavy-tailed IET distribution and memory effects between IETs. For this, we propose a Farlie-Gumbel-Morgenstern copula-based algorithm for generating event sequences with correlated IETs when the IET distribution and the memory coefficient between two consecutive IETs are given. We successfully apply our algorithm to the cases with heavy-tailed IET distributions. We also compare our algorithm to the existing shuffling method to find that our algorithm outperforms the shuffling method for some cases. Our copula-based algorithm is expected to be used for more realistic modeling of various dynamical processes.

DOI: [10.1103/PhysRevE.100.022307](https://doi.org/10.1103/PhysRevE.100.022307)

I. INTRODUCTION

Dynamical processes in various natural and social phenomena have been described by a series of events or event sequences showing non-Poissonian, bursty temporal patterns. Examples include solar flares [1], earthquakes [2,3], neuronal firings [4], and human activities [5,6]. Temporal correlations in such bursty time series can be understood not only by heterogeneous interevent times (IETs) but also by correlations between IETs [7,8]. Here the IET, denoted by τ , is defined as a time interval between two consecutive events. On the one hand, heterogeneities of IETs have been characterized by heavy-tailed or power-law IET distributions $P(\tau)$ with a power-law exponent α [6]:

$$P(\tau) \sim \tau^{-\alpha}. \quad (1)$$

On the other hand, the correlations between IETs can be characterized by a memory coefficient [7] among other measures such as local variation [9] and a bursty train [10]. The memory coefficient M is defined as the Pearson correlation coefficient between two consecutive IETs, the value of which for a sequence of n IETs, i.e., $\{\tau_i\}_{i=1,\dots,n}$, can be estimated by

$$M \equiv \frac{1}{n-1} \sum_{i=1}^{n-1} \frac{(\tau_i - \mu_1)(\tau_{i+1} - \mu_2)}{\sigma_1 \sigma_2}, \quad (2)$$

where μ_1 (μ_2) and σ_1 (σ_2) are the average and the standard deviation of the first (last) $n-1$ IETs, respectively. Positive M implies that the small (large) IETs tend to be followed by small (large) IETs. Negative M implies the opposite tendency,

while $M = 0$ is for the uncorrelated IETs. We mainly focus on the case with positive M , based on the empirical observations [7,11–13].

The dynamical processes, such as spreading and diffusion, taking place in a network of individuals are known to be strongly affected by bursty interaction patterns between individuals [14–22]. This topic has been studied in a framework of temporal networks [23,24], where a link connecting two nodes is considered existent or activated only at the moment of interaction. For example, an infectious disease, information, or a random walker on a node, say A , can be transferred to the A 's neighboring node, say B , only when A interacts with B . Therefore, the dynamical behavior, such as spreading speed, would be influenced by how often these nodes interact with each other, more generally, by the temporal interaction pattern between them. This temporal interaction pattern can be described by the event sequence when each event denotes the interaction. Consequently, modeling and simulating realistic temporal networks often requires us to generate event sequences for links (or nodes) showing the empirically observed properties, such as heavy-tailed IET distribution in Eq. (1) and/or memory effects measured by the memory coefficient in Eq. (2).

It is straightforward to generate event sequences characterized only by the IET distribution $P(\tau)$, i.e., when IETs are fully uncorrelated with each other as in the renewal process [25]: n IETs are independently drawn from the given $P(\tau)$ to make a sequence of IETs, $\{\tau_i\}_{i=1,\dots,n}$. Provided that the first event occurs at time t_0 , the events are considered to occur at times $t_i = t_0 + \sum_{i'=1}^i \tau_{i'}$ for $i = 1, \dots, n$.

In contrast, the generative methods for event sequences with correlated IETs have not been fully explored, except for a few recent works. We focus on two existing methods. The first method is to shuffle a set of IETs, prepared using a given

*hang-hyun.jo@apctp.org

†wsjung@postech.ac.kr

$P(\tau)$, to implement the desired value of M in Eq. (2) [22]. By this method one can control $P(\tau)$ and M independently, while the shape of $P(\tau)$ may set bounds on the range of M [12]. Note that for the implementation of this method the number of IETs should be predetermined, which, however, is not necessarily the case. For example, let us consider the simulation of a dynamical process using the event sequence; it is often the case that how many IETs are to be needed for reaching a stationary state of the process may not be known *a priori*. The second method is to generate IETs sequentially using the conditional probability distribution $P(\tau|\tau')$ [21], by which a next IET τ is drawn given the previous IET τ' . One can generate an arbitrary number of IETs without predetermining the number of IETs. However, since the function used in Ref. [21] is $P(\tau|\tau') = \alpha(1 + \tau')^\alpha / (\tau + \tau')^{\alpha+1}$, α controls both the heterogeneity of IETs and the degree of memory effects, implying the inevitable dependency between the statistics of IETs and correlations between IETs.

To overcome disadvantages in the previous methods, we propose an alternative generative method using the conditional probability distribution $P(\tau|\tau')$ but based on the Farlie-Gumbel-Morgenstern (FGM) copula [26,27], by which one can generate an arbitrary number of IETs without predetermining the number of IETs, while the statistics of IETs and their correlations can be independently controlled. In Sec. II we introduce the copula-based algorithm for generating event sequences with correlated IETs. In Sec. III we apply our algorithm to the cases with exponential and power-law IET distributions as well as power-law IET distribution with exponential cutoff, and we also discuss the performance of our algorithm for different cases in terms of computational times, in comparison to the shuffling method used in Ref. [22]. Finally, we conclude our paper in Sec. IV.

II. COPULA-BASED ALGORITHM

We introduce the copula-based algorithm for generating event sequences with correlated IETs for a given IET distribution $P(\tau)$ and memory coefficient M between two consecutive IETs. For this, we model the joint probability distribution $P(\tau_i, \tau_{i+1})$ by adopting a FGM copula among others [26,27]. It is the fact that the FGM copula is simple and analytically tractable, despite the range of correlation being somewhat limited, which will be discussed below. The joint probability distribution based on the FGM copula is written as [28]

$$P(\tau_i, \tau_{i+1}) = P(\tau_i)P(\tau_{i+1})[1 + rf(\tau_i)f(\tau_{i+1})], \quad (3)$$

where

$$f(\tau) \equiv 2F(\tau) - 1, \quad F(\tau) \equiv \int_0^\tau d\tau' P(\tau'). \quad (4)$$

Here $F(\tau)$ denotes the cumulative distribution function (CDF) of $P(\tau)$, and $P(\tau_i)$ and $P(\tau_{i+1})$ are assumed to have the same functional form. The parameter r , controlling the correlation between two consecutive IETs, is in the range of $-1 \leq r \leq 1$ because $P(\tau_i, \tau_{i+1}) \geq 0$ and $0 \leq F(\tau) \leq 1$, hence $-1 \leq f(\tau) \leq 1$. It is straightforward to relate r with the memory coefficient M in Eq. (2) using the FGM copula in Eq. (3) as

follows:

$$M \simeq \frac{\langle \tau_i \tau_{i+1} \rangle - \mu^2}{\sigma^2} = \frac{r}{\sigma^2} \left[\int_0^\infty d\tau \tau P(\tau) f(\tau) \right]^2 \equiv ar, \quad (5)$$

where

$$\langle \tau_i \tau_{i+1} \rangle \equiv \int_0^\infty d\tau_i \int_0^\infty d\tau_{i+1} \tau_i \tau_{i+1} P(\tau_i, \tau_{i+1}), \quad (6)$$

and μ and σ are the mean and standard deviation of IETs, respectively. The positive constant a is determined only by $P(\tau)$, irrespective of M . The upper bound of a is $1/3$ for any $P(\tau)$ as proven in Ref. [29], i.e., $|M| \leq a = 1/3$. Despite such relatively weak correlations by the FGM copula, one can still study the FGM copula as the empirical values of $|M|$ tend to be relatively small in several cases [7,11,12].

Using the joint probability distribution in Eq. (3), one can get the conditional probability distribution of τ_{i+1} for a given τ_i :

$$P(\tau_{i+1}|\tau_i) = \frac{P(\tau_i, \tau_{i+1})}{P(\tau_i)} = P(\tau_{i+1})[1 + rf(\tau_i)f(\tau_{i+1})]. \quad (7)$$

In order to draw a value of τ_{i+1} from $P(\tau_{i+1}|\tau_i)$, we use the inverse transform sampling or transformation method [30]: For a random number x drawn from a uniform distribution defined in $0 \leq x < 1$, we obtain τ_{i+1} for a given τ_i by solving the equation

$$1 - x = F(\tau_{i+1}|\tau_i), \quad (8)$$

where

$$F(\tau_{i+1}|\tau_i) \equiv \int_0^{\tau_{i+1}} d\tau' P(\tau'|\tau_i). \quad (9)$$

For convenience, we define

$$c_i \equiv rf(\tau_i) = r[2F(\tau_i) - 1], \quad (10)$$

to rewrite Eq. (9) as

$$\begin{aligned} F(\tau_{i+1}|\tau_i) &= (1 - c_i) \int_0^{\tau_{i+1}} d\tau' P(\tau') + 2c_i \int_0^{\tau_{i+1}} d\tau' P(\tau') F(\tau') \\ &= (1 - c_i)F(\tau_{i+1}) + c_i F(\tau_{i+1})^2. \end{aligned} \quad (11)$$

Then from Eq. (8) with $y \equiv F(\tau_{i+1})$, one gets

$$c_i y^2 + (1 - c_i)y + x - 1 = 0, \quad (12)$$

leading to

$$\tau_{i+1} = F^{-1} \left[\frac{c_i - 1 + \sqrt{(c_i + 1)^2 - 4c_i x}}{2c_i} \right], \quad (13)$$

where F^{-1} denotes the inverse function of $F(\tau)$. The sign of the square root was chosen to be “+” for satisfying $y = 1$ for $x = 0$ and $y \rightarrow 0$ for $x \rightarrow 1$.

By our copula-based algorithm, one can generate the event sequence without predetermining the number of IETs as well as with $P(\tau)$ and M being controlled independently. The generating procedure is as depicted in Fig. 1: We randomly draw the first IET from $P(\tau)$, denoted by τ_1 ; then τ_1 is used to generate τ_{i+1} using $P(\tau_{i+1}|\tau_i)$ or Eq. (13) with a random number x independently drawn for each $i = 1, 2, \dots$. This algorithm can be called Markovian in the sense that τ_{i+1} depends only on τ_i but not on $\tau_{i'}$ with $i' < i$.

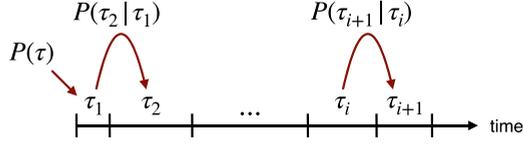


FIG. 1. Schematic diagram of the copula-based algorithm using a given interevent time (IET) distribution $P(\tau)$ and a conditional probability distribution $P(\tau_{i+1}|\tau_i)$ for two consecutive IETs in Eq. (7).

III. RESULTS

We apply the copula-based algorithm to three well-known IET distributions, i.e., exponential and power-law IET distributions as well as power-law IET distribution with exponential cutoff. We also test the performance of our algorithm for these three cases, in comparison to the shuffling method [22].

A. Exponential IET distribution

As a simple example, we consider the exponential IET distribution with the mean IET μ :

$$P(\tau) = \mu^{-1} e^{-\tau/\mu}, \quad (14)$$

$$F(\tau) = 1 - e^{-\tau/\mu}, \quad (15)$$

from which we get

$$r = 4M. \quad (16)$$

This implies that $|M| \leq 1/4$ by Eq. (5). The conditional probability distribution in Eq. (7) is written as

$$P(\tau_{i+1}|\tau_i) = \mu^{-1} e^{-\tau_{i+1}/\mu} \times [1 + 4M(1 - 2e^{-\tau_i/\mu})(1 - 2e^{-\tau_{i+1}/\mu})]. \quad (17)$$

Then from Eq. (13) we obtain τ_{i+1} for a given τ_i and a random number x as

$$\tau_{i+1} = \mu \ln \left[\frac{2c_i}{c_i + 1 - \sqrt{(c_i + 1)^2 - 4c_i x}} \right], \quad (18)$$

where

$$c_i = r f(\tau_i) = 4M(1 - 2e^{-\tau_i/\mu}). \quad (19)$$

For the demonstration, we generate 100 event sequences of size $n = 10^5$ for $\mu = 100$ and $M = 0.1$ to analyze them by measuring the conditional probability distributions and the resultant IET distribution, which are comparable to the corresponding analytical curves, as shown in Figs. 2(a) and 2(b). We also measure the memory coefficient $M = 0.100(4)$ from the same generated event sequences, which is in good agreement with the input value of 0.1.

B. Power-law IET distribution

Next, we study the case with a power-law IET distribution with power-law exponent α :

$$P(\tau) = (\alpha - 1)\tau^{-\alpha}\theta(\tau - 1), \quad (20)$$

$$F(\tau) = (1 - \tau^{1-\alpha})\theta(\tau - 1), \quad (21)$$

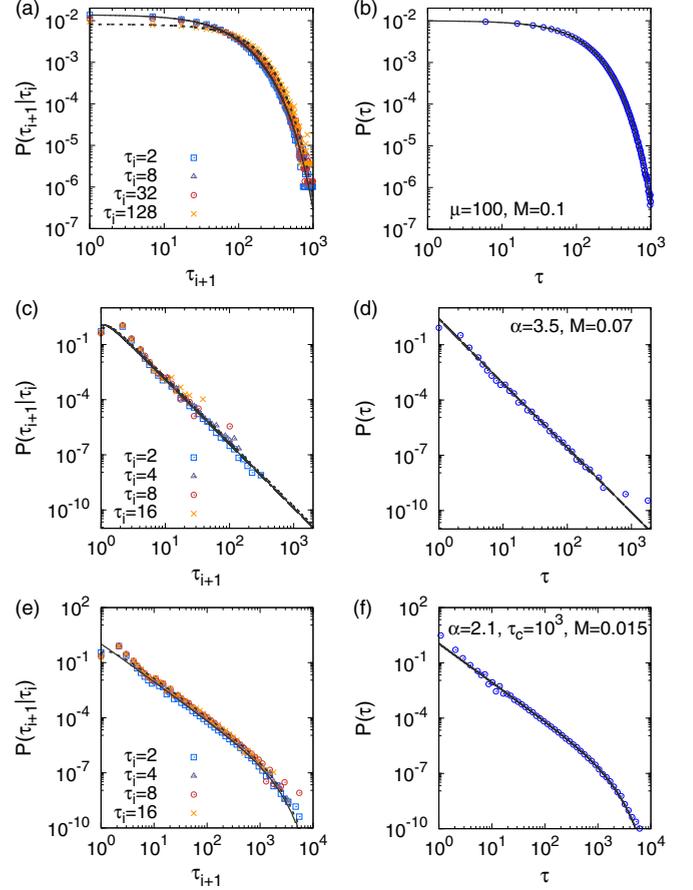


FIG. 2. Numerical results of the copula-based algorithm: Conditional probability distributions $P(\tau_{i+1}|\tau_i)$ for several values of τ_i (left) and the resultant IET distribution (right) for the exponential case in Eq. (14) with $\mu = 100$ and $M = 0.1$ (top), for the power-law case in Eq. (20) with $\alpha = 3.5$ and $M = 0.07$ (middle), and for the case of power law with exponential cutoff in Eq. (26) with $\alpha = 2.1$, $\tau_c = 10^3$, and $M = 0.015$ (bottom) are compared to analytical curves in Eqs. (17), (14), (23), (20), (28), and (26) in panels (a–f), respectively. In panel (a), the solid and dashed curves correspond to $\tau_i = 2$ and 128, respectively. In panels (c, e) the solid and dashed curves correspond to $\tau_i = 2$ and 16, respectively. In panels (a, b) the curves are linearly binned, while the curves in panels (c–f) are all log-binned. Numerical results are averaged over 100 event sequences of size $n = 10^5$.

where $\theta(\cdot)$ denotes the Heaviside step function and the lower bound of IET has been set to 1. We assume that $\alpha > 3$ for the finite variance of IETs. The relation between r and M is obtained as

$$r = \frac{(2\alpha - 3)^2}{(\alpha - 1)(\alpha - 3)} M. \quad (22)$$

The conditional probability distribution in Eq. (7) for $\tau_i \geq 1$ and $\tau_{i+1} \geq 1$ is written as

$$P(\tau_{i+1}|\tau_i) = (\alpha - 1)\tau_{i+1}^{-\alpha} \times \left[1 + \frac{(2\alpha - 3)^2}{(\alpha - 1)(\alpha - 3)} M (1 - 2\tau_i^{1-\alpha})(1 - 2\tau_{i+1}^{1-\alpha}) \right]. \quad (23)$$

Then from Eq. (13) we obtain τ_{i+1} for a given τ_i and a random number x as

$$\tau_{i+1} = \left[\frac{2c_i}{c_i + 1 - \sqrt{(c_i + 1)^2 - 4c_i x}} \right]^{1/(\alpha-1)}, \quad (24)$$

where

$$c_i = rf(\tau_i) = \frac{(2\alpha - 3)^2}{(\alpha - 1)(\alpha - 3)} M(1 - 2\tau_i^{1-\alpha}). \quad (25)$$

Using our copula-based algorithm we generate 100 event sequences of size $n = 10^5$ for $\alpha = 3.5$ and $M = 0.07$. Note that $\alpha = 3.5$ leads to $|M| \leq a = 5/64 \approx 0.078$. From the generated event sequences the conditional probability distributions and the resultant IET distribution are measured and compared to the corresponding analytical curves, as shown in Figs. 2(c) and 2(d). We also find the measured memory coefficient $M = 0.08(1)$ comparable to the input value of 0.07.

C. Power-law IET distribution with exponential cutoff

As a more realistic case evidenced by empirical results [6], we consider a power-law IET distribution with exponential cutoff τ_c as

$$P(\tau) = \frac{\tau_c^{\alpha-1}}{\Gamma(1-\alpha, 1/\tau_c)} \tau^{-\alpha} e^{-\tau/\tau_c} \theta(\tau - 1), \quad (26)$$

$$F(\tau) = \left[1 - \frac{\Gamma(1-\alpha, \tau/\tau_c)}{\Gamma(1-\alpha, 1/\tau_c)} \right] \theta(\tau - 1), \quad (27)$$

where $\Gamma(\cdot, \cdot)$ is the upper incomplete Gamma function and the lower bound of IET has been set to 1. Note that the IET distribution in Eq. (26) reduces to that in Eq. (20) in the limit of $\tau_c \rightarrow \infty$, or to that in Eq. (14) if $\alpha = 0$ and the lower bound of IET is set to zero. In contrast to the exponential and power-law cases, in the case of power law with exponential cutoff the analytic calculation of the integration in Eq. (5) is not straightforward, hence a in Eq. (5) will be numerically evaluated. With this numerical value of a , the conditional probability distribution in Eq. (7) is written as

$$P(\tau_{i+1}|\tau_i) = \frac{\tau_c^{\alpha-1}}{\Gamma(1-\alpha, 1/\tau_c)} \tau_{i+1}^{-\alpha} e^{-\tau_{i+1}/\tau_c} \times \left\{ 1 + \frac{M}{a} \left[1 - 2 \frac{\Gamma(1-\alpha, \tau_i/\tau_c)}{\Gamma(1-\alpha, 1/\tau_c)} \right] \left[1 - 2 \frac{\Gamma(1-\alpha, \tau_{i+1}/\tau_c)}{\Gamma(1-\alpha, 1/\tau_c)} \right] \right\}. \quad (28)$$

Then the value of τ_{i+1} can be obtained for a given $\tau_i \geq 1$ and a random number x by numerically solving Eq. (13), namely,

$$\frac{\Gamma(1-\alpha, \tau_{i+1}/\tau_c)}{\Gamma(1-\alpha, 1/\tau_c)} = \frac{c_i + 1 - \sqrt{(c_i + 1)^2 - 4c_i x}}{2c_i}, \quad (29)$$

where

$$c_i = rf(\tau_i) = \frac{M}{a} \left[1 - 2 \frac{\Gamma(1-\alpha, \tau_i/\tau_c)}{\Gamma(1-\alpha, 1/\tau_c)} \right]. \quad (30)$$

We generate 100 event sequences of size $n = 10^5$ for $\alpha = 2.1$, $\tau_c = 10^3$, and $M = 0.015$. Note that for $\alpha = 2.1$ and $\tau_c = 10^3$ we have $|M| \leq a \approx 0.020$. The generated event sequences are analyzed to result in the conditional probability

distributions and the resultant IET distribution that are comparable to the corresponding analytical curves, as shown in Figs. 2(e) and 2(f). The measured memory coefficient $M = 0.015(5)$ is found to be close to the input value of 0.015. All these results indicate that the correlations between two consecutive IETs are successfully implemented.

D. Computation times

We discuss the performance of our algorithm in terms of computation times for generating event sequences with correlated IETs. For the exponential and power-law IET distributions, the generation of event sequences is easy and fast, while for the power-law IET distribution with exponential cutoff (“power+cutoff” in short) one needs to numerically solve Eq. (29) for the generation of each IET, implying longer computation times than other cases. To study this issue, we measure the average computation times in seconds for generating event sequences. We use codes written in C on a Linux system with 3.5-GHz Intel Core i5-7600 CPU and 16-GB RAM. We also use GNU Scientific Library for calculating the incomplete Gamma function, e.g., in Eqs. (29) and (30).

We test various combinations of parameter values for the estimation of computation times when $n = 10^5$ is fixed. For the exponential case, we use $\mu = 10, 10^2, 10^3$ and $M = 0.01, 0.03, 0.1, 0.2$. For the power-law case, we use $\alpha = 3.3, 3.5, 4$ and $M = 0.001, 0.01, 0.03, 0.07$. For the power+cutoff case, $\alpha = 2.1, 2.5, 2.9$ and $M = 0.005, 0.01, 0.015$ are used for a fixed $\tau_c = 10^3$. Figures 3(a)–3(c) show that the average computation times in the power+cutoff case are larger than those for other two cases, as expected. We observe that the computation times are linearly increasing with n in Figs. 3(d)–3(f), which is trivial for the copula-based algorithm.

We now compare the computation times of the copula-based algorithm to those of the shuffling method in Ref. [22]. For implementing the shuffling method, we first draw n random values from $P(\tau)$ to make an IET sequence $T \equiv \{\tau_i\}_{i=1, \dots, n}$. Using the definition of Eq. (2), we measure the memory coefficient from T , denoted by \tilde{M} . Two IETs are randomly chosen in T and swapped only when this swapping makes \tilde{M} closer to M , i.e., the target value. By repeating the swapping, we can obtain the IET sequence the \tilde{M} of which is close enough to M . Precisely, the swapping stops when $|M - \tilde{M}| < \epsilon$ with a small number ϵ .

The computation times for generating event sequences by the shuffling method are estimated for the same combinations of parameter values as in the copula-based algorithm, together with several values of $\epsilon = 0.001, \dots, 0.02$. Note that the computation time for the shuffling method is the sum of the time for generating the initial set of IETs for each given IET distribution and the time for the shuffling procedure. In Figs. 3(a)–3(c) we find that the copula-based algorithm outperforms the shuffling method only for the power+cutoff case by a factor of 10, while for the exponential and power-law cases the shuffling method is around twice faster than the copula-based algorithm. Therefore, the shuffling procedure itself seems relatively fast. Then much slower generation by the shuffling method in the power+cutoff case could be due

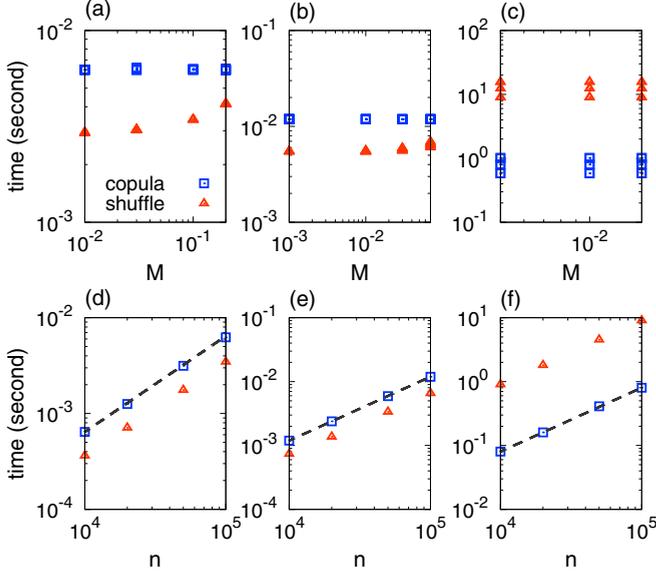


FIG. 3. Average computation times in seconds for generating an event sequence using the copula-based algorithm (square) and the shuffling method (triangle) for the exponential case (a, d), the power-law case (b, e), and the case of power law with exponential cutoff (“power+cutoff” in short) (c, f). In panels (a–c), the computation times for various values of parameters (see the text) are plotted as a function of M for $n = 10^5$. In panels (d–f), we show the n dependence of computation times when the event sequences are generated using the same parameter values as in Fig. 2, where the dashed line of $y = x$ is for guiding the eye. Each point is averaged over 100 event sequences for exponential and power-law cases and ten event sequences for the power+cutoff case.

to the longer time for generating the initial set of IETs using Eq. (26) [31]. We also find that the computation times for the shuffling method are mostly insensitive to the variation of parameter values, except for the effect of M in the exponential and power-law cases. This can be explained by the fact that the initial sequence of IETs drawn from $P(\tau)$ is uncorrelated, i.e., $\tilde{M} \approx 0$, implying that it takes longer time to reach the larger target value of M . Such M dependence is not apparent in the power+cutoff case probably due to the dominant effects of the time for generating the initial set of IETs. In addition, the computation times turn out to be linearly increasing with n in Figs. 3(d)–3(f) as in the copula-based algorithm.

Finally the quality of event sequences generated by the shuffling method is tested by measuring the conditional probability distribution $P(\tau_{i+1}|\tau_i)$ and the IET distribution $P(\tau)$ from the generated event sequences using the same parameter values as in Fig. 2. Here we choose $\epsilon = 10^{-3}$, which is of the order of the standard deviations of estimated M in the case with the copula-based algorithm. In Fig. 4 we find almost the same behaviors as in the results from the copula-based algorithm, implying that the copula-based algorithm and the shuffling method can be used interchangeably. It might be due to the fact that both methods are designed to implement only the correlations between two consecutive IETs for a given IET distribution, while randomizing or ignoring all other higher-order correlations between IETs. In this sense, another generative method called the Laplace Gillespie algorithm (LGA)

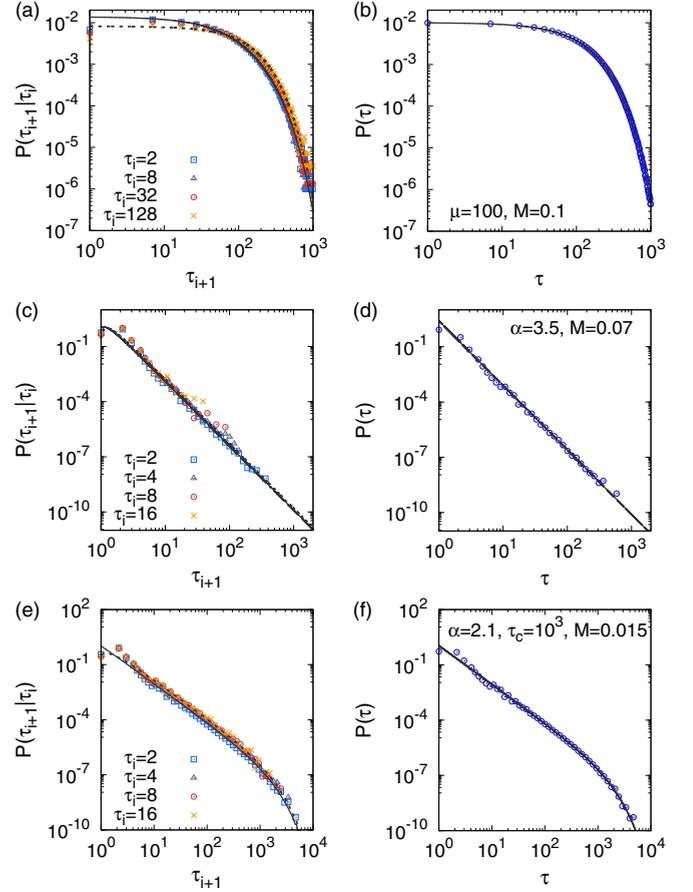


FIG. 4. Numerical results of the shuffling method with the same parameter values as in Fig. 2, with $\epsilon = 10^{-3}$.

[32] can also be compared to our algorithm and shuffling method as it can generate event sequences with correlated IETs. However, we leave this comparison as a future work mainly because fine-tuning the value of M by the LGA seems to be difficult.

IV. CONCLUSION

We have proposed the FGM copula-based algorithm for generating event sequences with correlated IETs for a given IET distribution and a given memory coefficient between two consecutive IETs. This is to overcome the disadvantages in the previous generative methods, i.e., the shuffling method in Ref. [22] and the method using conditional probability distribution in Ref. [21]: By adopting the conditional probability distribution $P(\tau_{i+1}|\tau_i)$ based on the FGM copula [26,27], one can generate an arbitrary number of IETs without pre-determining the number of IETs, while the statistics of IETs and their correlations can be independently controlled. After deriving the analytical forms of the next IET τ_{i+1} for a given previous IET τ_i , we show that our algorithm successfully generates the event sequences with desired statistical properties.

We also compare the performance of our copula-based algorithm to the shuffling method in terms of computation times: It turns out that the copula-based algorithm outperforms the shuffling method for generating event sequences

with power-law IET distributions with exponential cutoff, while for the exponential and power-law IET distributions the shuffling method is around twice faster than the copula-based algorithm. We note that since both methods generate event sequences of the same statistical properties, any of them can be used appropriately. Considering the advantages of our algorithm, we expect our algorithm to be used for modeling and simulating more realistic event sequences, eventually for more realistic temporal networks.

Finally we remark on the limited range of M . Apart from the bounds of M set by the shape of $P(\tau)$ [12], the range of M is also bounded due to the form of the FGM copula, while it is more flexible in the shuffling method. Therefore, other members of the FGM family, e.g., the iterated FGM copula

[33], can be investigated to explore a wider range of M for the IET distributions of our interest as a future work. In addition, our copula-based algorithm can be used to generate any other sequence of correlated variables.

ACKNOWLEDGMENTS

H.-H.J. was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF Grant No. 2018R1D1A1A09081919). W.-S.J. was supported by Basic Science Research Program through the NRF funded by the Ministry of Education (NRF Grant No. 2016R1D1A1B03932590).

-
- [1] M. S. Wheatland, P. A. Sturrock, and J. M. McTiernan, *Astrophys. J.* **509**, 448 (1998).
- [2] Á. Corral, *Phys. Rev. Lett.* **92**, 108501 (2004).
- [3] L. de Arcangelis, C. Godano, E. Lippiello, and M. Nicodemi, *Phys. Rev. Lett.* **96**, 051102 (2006).
- [4] T. Kemuriyama, H. Ohta, Y. Sato, S. Maruyama, M. Tandai-Hiruma, K. Kato, and Y. Nishida, *BioSystems* **101**, 144 (2010).
- [5] A.-L. Barabási, *Nature (London)* **435**, 207 (2005).
- [6] M. Karsai, H.-H. Jo, and K. Kaski, *Bursty Human Dynamics* (Springer International, Cham, 2018).
- [7] K.-I. Goh and A.-L. Barabási, *Europhys. Lett.* **81**, 48002 (2008).
- [8] H.-H. Jo, *Phys. Rev. E* **96**, 062131 (2017).
- [9] S. Shinomoto, K. Shima, and J. Tanji, *Neural Comput.* **15**, 2823 (2003).
- [10] M. Karsai, K. Kaski, A.-L. Barabási, and J. Kertész, *Sci. Rep.* **2**, 397 (2012).
- [11] W. Wang, N. Yuan, L. Pan, P. Jiao, W. Dai, G. Xue, and D. Liu, *Physica A (Amsterdam)* **436**, 846 (2015).
- [12] F. Guo, D. Yang, Z. Yang, Z.-D. Zhao, and T. Zhou, *Phys. Rev. E* **95**, 052314 (2017).
- [13] L. Böttcher, O. Woolley-Meza, and D. Brockmann, *PLoS ONE* **12**, e0178062 (2017).
- [14] A. Vazquez, *Physica A (Amsterdam)* **373**, 747 (2007).
- [15] M. Karsai, M. Kivelä, R. K. Pan, K. Kaski, J. Kertész, A.-L. Barabási, and J. Saramäki, *Phys. Rev. E* **83**, 025102(R) (2011).
- [16] G. Miritello, E. Moro, and R. Lara, *Phys. Rev. E* **83**, 045102(R) (2011).
- [17] L. E. C. Rocha, F. Liljeros, and P. Holme, *PLOS Comput. Biol.* **7**, e1001109 (2011).
- [18] H.-H. Jo, J. I. Perotti, K. Kaski, and J. Kertész, *Phys. Rev. X* **4**, 011041 (2014).
- [19] J. I. Perotti, H.-H. Jo, P. Holme, and J. Saramäki, Temporal network sparsity and the slowing down of spreading, [arXiv:1411.5553](https://arxiv.org/abs/1411.5553) (2014).
- [20] J.-C. Delvenne, R. Lambiotte, and L. E. C. Rocha, *Nat. Commun.* **6**, 7366 (2015).
- [21] O. Artime, J. J. Ramasco, and M. San Miguel, *Sci. Rep.* **7**, 41627 (2017).
- [22] T. Hiraoka and H.-H. Jo, *Sci. Rep.* **8**, 15321 (2018).
- [23] P. Holme and J. Saramäki, *Phys. Rep.* **519**, 97 (2012).
- [24] N. Masuda and R. Lambiotte, *A Guide to Temporal Networks*, Series on Complexity Science (World Scientific, Singapore, 2016).
- [25] F. Mainardi, R. Gorenflo, and A. Vivoli, *J. Comput. Appl. Math.* **205**, 725 (2007).
- [26] R. B. Nelsen, *An Introduction to Copulas*, Springer Series in Statistics (Springer, New York, 2006).
- [27] T. T. Takeuchi, *Mon. Not. R. Astron. Soc.* **406**, 1830 (2010).
- [28] The FGM copula is a function C joining a bivariate cumulative distribution function (CDF) to their one-dimensional marginal CDFs such that $G(x, y) = C[u(x), v(y)] = uv[1 + r(1 - u)(1 - v)]$, where u and v are CDFs of variables x and y , respectively [27, 26]. Then the bivariate probability distribution function (PDF) of x and y is obtained by $\frac{\partial^2 G}{\partial x \partial y} = P_1(x)P_2(y)[1 + r(2u - 1)(2v - 1)]$, where $P_1(x)$ and $P_2(y)$ denote PDFs of x and y , respectively. The FGM copula for IETs has recently been found to be useful for the analytical approach to autocorrelation functions [34].
- [29] W. R. Schucany, W. C. Parr, and J. E. Boyer, *Biometrika* **65**, 650 (1978).
- [30] A. Clauset, C. R. Shalizi, and M. E. J. Newman, *SIAM Rev.* **51**, 661 (2009).
- [31] To generate the IETs from the power-law distribution with exponential cutoff, we have used the method described in Ref. [30], which indeed requires one more random number per each IET than the cases with exponential and power-law IET distributions.
- [32] N. Masuda and L. E. C. Rocha, *SIAM Rev.* **60**, 95 (2018).
- [33] J. S. Huang and S. Kotz, *Biometrika* **71**, 633 (1984).
- [34] H.-H. Jo, *Phys. Rev. E* **100**, 012306 (2019).