A generalized linear threshold model for an improved description of the spreading dynamics

Yijun Ran, Xiaomin Deng, Xiaomeng Wang, and Tao Jia
College of Computer and Information Science, Southwest University, Beibei, Chongqing, 400715 P. R. China

(Dated: 18 December 2020)

Many spreading processes in our real-life can be considered as a complex contagion, and the linear threshold (LT) model is often applied as a very representative model for this mechanism. Despite its intensive usage, the LT model suffers several limitations in describing the time evolution of the spreading. First, the discrete-time step that captures the speed of the spreading is vaguely defined. Second, the synchronous updating rule makes the nodes infected in batches, which can not take individual differences into account. Finally, the LT model is incompatible with existing models for the simple contagion. Here we consider a generalized linear threshold (GLT) model for the continuous-time stochastic complex contagion process that can be efficiently implemented by the Gillespie algorithm. The time in this model has a clear mathematical definition and the updating order is rigidly defined. We find that the traditional LT model systematically underestimates the spreading speed and the randomness in the spreading sequence order. We also show that the GLT model works seamlessly with the susceptible-infected (SI) or susceptible-infected-recovered (SIR) model. One can easily combine them to model a hybrid spreading process in which simple contagion accumulates the critical mass for the complex contagion that leads to the global cascades. Overall, the GLT model we proposed can be a useful tool to study complex contagion, especially when studying the time evolution of the spreading.

The linear threshold (LT) model is a typical model for the complex contagion process. However, it systematically underestimates the spreading speed and the randomness in the spreading sequence order. To cope with this issue, we propose a generalized linear threshold (GLT) model, where the time evolution is controlled by the continuous-time stochastic process. The GLT model can be efficiently implemented by the Gillespie algorithm, providing a useful tool to investigate and simulate more complicated spreading processes, especially when the time evolution is the focus.

I. INTRODUCTION

The process of adoption such as the adoption of innovations, commercial products, and social behavior can be considered as a kind of contagion process in social systems. It assumes that the infection probability also increases monotonically with the number of infected neighbors. The complex contagion is inspired by collective behaviors in social systems. It assumes that the infection can be important information such as when the global cascading could occur or how to select effective seed nodes to maximize the spreading. When it comes to the spreading dynamics, the LT model suffers three limitations. First, the evolution in LT model is controlled by discrete-time steps that lack a proper definition, which gives rise to issues when the speed of the spreading needs to be investigated. Second, the status of a node is updated in a synchronous manner. At each time step, all nodes currently satisfying the spreading threshold will turn into the infected state. This can be an issue in application such as machine learning where the order of infection can be important information. Finally, the LT model is not very flexible. It is both theoretically and practically challenging if one plans to combine the LT model and other simple contagion model to model some complicated hybrid spreading processes.

To overcome these limitations, we consider a generalized linear threshold (GLT) model for the continuous-time stochastic spreading process that can be efficiently implemented by the Gillespie algorithm. The evolutionary time in the new model has a physical meaning, which is associated with the...
A generalized linear threshold model

rate of the underlying stochastic process. We find that compared with the GLT model, the traditional LT model tends to underestimate the spreading speed. The order of nodes being infected is properly defined in the GLT model, allowing us to better generate synthetic spreading node sequence to model the spreading in real systems. Finally, the GLT model is compatible with the susceptible-infected (SI) or susceptible-infected-recovered (SIR) model, because they are defined under the same mathematical framework. One can easily build a hybrid spreading by combining both simple and complex contagion, or adding the recovery process into the complex contagion. The remainder of the paper is structured as follows. We first give a brief description of the classical LT model with both the synchronous and asynchronous updating rules. We then propose the GLT model and show how to model it efficiently with the Gillespie algorithm. To further shed light on this model, we compare the spreading results from the GLT and LT model. Finally, we show how the GLT model can be combined with other spreading models.

II. RESULTS

A. The linear threshold (LT) model

The LT model was first introduced in the field of social science to analyze the effects of social reinforcement by assuming that each adoption requires a certain fraction of exposures. The community of network science may be more familiar with the work by Duncan Watts where the LT model is used to study the condition for global spreading. The same model was also applied in the community of computer science to find the optimal initiator set that maximizes the spreading outcome. In the LT model, each node is in one of the two possible states: 0 (inactive, susceptible, etc.) or 1 (active, infected, etc.). A node in the network can switch only from state 0 to state 1. The transition probability depends on the fraction of its neighbors that are on state 1, denoted by , as

\[ p(\phi) = \begin{cases} 
0 & \text{if } \phi < \phi^*, \\
1 & \text{if } \phi \geq \phi^*,
\end{cases} \tag{1} \]

where \( \phi^* \) is the threshold value of node \( i \), which can be chosen from a probability distribution or stay fixed for all nodes. Note that there are other variations for the choice of threshold, such as the number of contacts or the number of infected neighbors. In this paper, we adopt the model by Watts that uses the fraction of infected neighbors.

The evolution of the system is characterized by discrete-time steps in the LT model. At each time step, we go through the network and calculate the transition probability of each node according to Eq. (1). All nodes that can change the state are updated synchronously in that time step. The process is repeated until no more nodes can change the state. The time step that characterizes the system evolution, however, is never explicitly defined. This may be because that initial studies that proposed the model mainly focused on the outcome of the spreading, which does not depend on the choice of time step or how the system actually evolves with time. Nevertheless, the time step needs a proper definition when the spreading dynamics are concerned. Indeed, while the discrete-time step is used in both the LT and SI model, they are inherently different with distinct physical meanings. In the SI model, the time step is associated with the probability that the disease is transferred from one node to another, or the chance that a node gets infected from an infected neighbor. If the time step is equivalent to a longer period of real time, the infection probability would be tuned larger, which eventually gives the same spreading dynamics. As an example, the infection probability in disease spreading would be different if the time step refers to an hour or a day. In the LT model, however, the time step is associated with a node’s status updating, which is independent of the transmission probability. Its physical meaning, related to why every node updates its status within one time step, is not clearly interpreted.

Another issue is the order of the infection. Under the synchronous updating rule, all nodes satisfying the threshold condition change the state together in one time step. Considering the case that node \( i \) changes the state from 0 to 1, which makes its neighboring node \( j \) reach the threshold. While the transition condition is satisfied, node \( j \) can not change the state in that time step. In other words, node \( j \)’s state is frozen till all nodes in the same batch of node \( i \) complete the transition. In terms of the infection order, node \( j \) always ranks behind them. Note that the infection order is important in tasks such as tracking the spreading source or learning the embedding of the underlying network. The simplification of LT model may limit its application in generating the synthetic spreading node sequence in real systems. A simple fix of this issue is to use the asynchronous updating rule. One option is that at each time step, we randomly pick only one node from those whose threshold is reached and update the node’s state. In this way, the spreading order would be more realistic. However, the spreading dynamics would become unrealistic as the number of infected nodes increases linearly with time steps. An alternative option is to randomly pick an arbitrary node at each time step regardless of its threshold condition and update its states according to its \( p(\phi) \). This actually becomes a Monte Carlo simulation. But the computational complexity raised to \( O(N^2) \) where \( N \) is the number of nodes in a network. More importantly, even though the asynchronous updating rule can fix the order, it is very difficult to model a system with individual differences. For example, if we assume that some nodes are more active and would change the states faster than others, it would be very difficult to implement this feature in the model.

Finally, the LT model is not very flexible. This is partially related with the vague definition of the discrete-time step. If we want to model a system with both simple and complex contagion, we need to define two types of time steps. One type of time step is for the deterministic infection in the LT model and the other for the probabilistic infection in the SI model. The conversion between the two types of time step can be an interesting interplay, which, however, lacks a proper definition and brings challenges for theoretical interpretation. Because the node status is updated synchronous at the end
of LT time step, the spreading curve will not be smooth but containing multiple bursts separated by fixed time intervals. We also need to propose a rule to decide which action should occur first when the two types of time step coincides. All these difficulties increase when more dynamics are involved, such as adding a recovery process to have the susceptible\textendash infected\textendash susceptible (SIS) or SIR model in the system. Therefore, it is challenging to apply the traditional LT model for complex spreading process.

B. A generalized linear threshold model and its stochastic simulation

To cope with the issues mentioned, we consider a simple variation of the original LT model and generalize it to continuous-time stochastic process. In the generalized linear threshold (GLT) model, a node $i$ has a certain rate to transfer from state 0 to state 1, which is given by

$$
\beta_i(\phi_i) = \begin{cases} 
0 & \text{if } \phi_i < \phi_i^* \\
k_i & \text{if } \phi_i \geq \phi_i^*.
\end{cases}
$$

(2)

Eq. [2] is similar to Eq. [1], both capturing a threshold dynamic. When $\phi_i$ is below the threshold, the transition (or infection) cannot occur. When $\phi_i$ is above the threshold, the transition will occur in certain. The extra information given by Eq. [2] is the rate $k_i$, which controls the speed of the transition and to what extent node $i$ would be infected ahead of other nodes. By assigning different $k_i$ value to different nodes, the individual differences on the transition are well characterized.

To efficiently simulate the GLT model, we apply the Gillespie algorithm. It is an efficient simulation method for the stochastic process and was heavily used to investigate the interactions of molecules in chemical systems or the cellular growth and division in biological systems. It can also be used to simulate the epidemic spreading such as SI and SIR model. Indeed, though it is not explicitly specified, when we use a rate $k$ to quantify a dynamic process, we imply that it is a Poisson process with a rate $k$. The inter-event time or waiting time $\tau$ is random and follows an exponential distribution with a rate $k$. This property can be generalized to cases when multiple Poisson processes coexist. Assume that there are $N$ nodes in the network, each has a transition rate $\beta_i$. The inter-event time $\tau$ for the occurrence of next transition follows an exponential distribution with rate $\tilde{\beta} = \sum_{i=1}^{N} \beta_i$. In practice, $\tau$ can be efficiently calculated from a random number $r$ uniformed picked from the interval $(0,1)$ as

$$
\tau = -\frac{\ln r}{\tilde{\beta}}.
$$

(3)

The probability that the transition takes place on node $j$ linearly depends on its transition rate as

$$
p = \frac{\beta_j}{\tilde{\beta}}.
$$

(4)

Algorithm 1 The generalized linear threshold model based on the Gillespie algorithm

**Input:** Network $G$, infection rate $\beta$, threshold $\phi^*$, initial seeds $\rho$

**Output:** Time series list $T$, susceptible number list $S$, infected number list $I$

1. function GLT($G$, $\beta$, $\phi^*$, $\rho$)
2. $T, S, I \leftarrow [0], |G| - |\rho|, |\rho|$ where the $|G|$ is the number of nodes
3. $\text{nodes} \leftarrow \text{nodes in the } G$
4. $\text{infected_nodes} \leftarrow \text{random.sample(nodes, } \rho)$
5. $\text{risk_nodes} \leftarrow \text{the susceptible neighbors of the infected_nodes}$
6. $\text{susceptible_nodes} \leftarrow []$
7. for $u$ in nodes do
8. $\text{infected_rate}[u] \leftarrow \beta$
9. end for
10. $\tau \leftarrow 0$
11. for $n$ in risk_nodes do
12. $\text{num}[n] \leftarrow$ the number of infected neighbors of $n$
13. $\text{degree}[n] \leftarrow$ the number of neighbors of $n$
14. if $\frac{\text{num}[n]}{\text{degree}[n]} \geq \phi^*$ then
15. add $n$ into susceptible_nodes
16. remove $n$ from risk_nodes
17. end if
18. end for
19. total_rate $\leftarrow \sum_{n \in \text{susceptible_nodes}} \text{infected_rate}[n]$
20. while total_rate $> 0$ do
21. $n = \text{random.choice(susceptible_nodes)}$
22. remove $n$ from susceptible_nodes
23. add $n$ into infected_nodes
24. $\tau \leftarrow \tau - \text{ln (random.uniform(0.0, 1.0))}$
25. Update $T, S, I$ total_rate
26. susceptible_neighbors $\leftarrow$ the susceptible neighbors of the $n$
27. for $u$ in susceptible_neighbors do
28. if $u$ not in susceptible_nodes then
29. risk_nodes $\leftarrow u$
30. end if
31. end for
32. for $n$ in risk_nodes do
33. $\text{num}[n] \leftarrow$ the number of infected neighbors of $n$
34. $\text{degree}[n] \leftarrow$ the number of neighbors of $n$
35. if $\frac{\text{num}[n]}{\text{degree}[n]} \geq \phi^*$ then
36. add $n$ into susceptible_nodes
37. remove $n$ from risk_nodes
38. end if
39. end for
40. total_rate $\leftarrow \sum_{n \in \text{susceptible_nodes}} \text{infected_rate}[n]$
41. end while
42. return $T, S, I$
43. end function

The Gillespie algorithm takes this property of the stochastic process. At each simulation step, it decides, in a random manner, which event would occur and when it would occur. The procedure can be summarized as follows:

1. At the time $t$, find all events that may occur (with a positive rate) and get the sum of the rate $\tilde{\beta}$.
2. Generate a random variable $\tau$ from an exponential distri-
function with rate \( \hat{\beta} \).
3. Randomly draw an event according to the probability of \( p \) in Eq. (4).
4. Update the system according to the event drawn. Update the time from \( t \) to \( t + \tau \).
5. Repeat from step 1.

To illustrate the simulation of the GLT model, we provide the pseudocode in Algorithm 1. At each simulation step, we need to determine which action would occur from the rates of all actions. When the \( k_i \) is the same for all nodes, or there are only a few choice of \( k_i \) values, we can do a random selection of actions to simplify this process, which takes only \( O(1) \) complexity. In comparison with the Monte Carlo version of the LT model with complexity \( O(N^2) \), the Gillespie algorithm significantly reduces the computation cost. When all nodes have different \( k_i \) values, the Monte Carlo version of the LT model would fail because it assumes that all nodes are picked to update the states with equal probability. The Gillespie algorithm can handle this situation by deciding the process that happens according the rate \( k_i \). The selection is a typical fitness proportionate selection, also known as roulette-wheel selection. The complexity is usually \( O(N) \) because we need to calculate \( \hat{\beta}_i / \hat{\beta} \) for every node at each simulation step. However, using a recently proposed optimization, the complexity can be reduced to \( O(1) \) type. Taken together with the \( N \) nodes in the system, the complexity to simulate the whole evolution is roughly \( O(N) \).

C. The application of the GLT model

To show features of the GLT model, we compare its time evolution with that of the LT model. Because the transition rate \( k_i \) can be any value, we have to first adjust the continuous rate and the discrete-time step to make the continuous-time and discrete-time model comparable. Unlike SI model where the relationship among the rate, the infection probability and the discrete-time step is known, there is no method yet to handle the parameter conversion in the threshold model. To cope with this issue, we consider re-scaling the spreading time window. We choose average cascade size \( S = 0.98 \) as our reference point and record the time (either discrete-time steps or continuous time) takes from the beginning of the spreading to \( S = 0.98 \) as the time window. The discrete-time steps and the continuous time are then re-scaled such that the spreading time window is the same. We consider \( S = 0.98 \) instead of \( S = 1 \) to avoid possible tails in the infection of the last node. To test the validity of our approach, we first apply it to SI model. After re-scaling the time window, the continuous-time and discrete-time spreading curve overlap each other (Fig. 1(a)), giving rise to the same results by parameter conversion. Hence the approach of time window re-scaling is effective. For the threshold model, we consider its Monte Carlo version as the baseline, where an arbitrary node is picked at random at each time step regardless of its threshold condition. The state of the node is then updated according to Eq. (4). This baseline is compared with the spreading generated by the GLT model and the traditional LT model with synchronous updating rule. The dynamics by the GLT model matches with the baseline, but the dynamics of the LT model is different, where the infected size grows slower than both the GLT model and the baseline (Fig. 1(b)). This indicates that the LT model underestimates the speed of spreading, supporting our initial statement of the LT model’s limitations.

The LT model underestimates the spreading dynamics due to its coarse-grained description of the time evolution. The spreading speed is not instantaneously updated according to the number of nodes satisfying the threshold condition. As an example, let us assume there are 10 nodes satisfying the threshold in the LT model. Naturally, these 10 nodes will be infected in the next time step. If we assume one time step corresponds to a continuous time \( T \), the infection of each of the 10 nodes takes \( T/10 \) time on average. In the LT model, if there are 10 nodes satisfying the threshold, the infection rate of the first node will be \( 10 \times k \) (assuming \( k_i = k \) for all nodes). The average waiting time to infect the first node is \( T/10 \) if \( k \) is set as \( k = 1/T \), which is the same as that in the LT model. However, the infection of one node will activate more nodes during the spreading. Therefore, after the infection of this node, there will be more than 10 nodes in the system satisfying the threshold. The rate for the next infection to occur is greater than \( 10 \times k \) and the average waiting time takes less than \( T/10 \) time. As the spreading proceeds, the nodes available for state change accumulates, which presents a higher spreading speed than what LT model suggests.

We also compare the spreading sequence generated by the two models. We fix all other parameters, pick a node, and focus on the rank of the node in the full spreading sequence. Due to the randomness in the spreading, the rank is not the same in every spreading. Hence we measure the distribution of the rank. Note that in the LT model, multiple nodes change the state in a batch. So we randomize the order of infection in the same batch in the LT model. We find that the rank of a node’s infection can be very segregated in the LT model, and can be very random in the GLT model (Fig. 2). Indeed, even when we increase the \( k_i \) of the node selected, giving it a much higher priority to change the state, its rank is still very
FIG. 2: The distribution of the rank of an infected node in spreading sequences on an Erdös-Rényi network with $N = 1000$ nodes and the average degree $\langle k \rangle = 4$. The threshold value is $\phi^* = 0.16$. We fix the node in all models and check when it will be infected. (a) The LT model, (b) the GLT model in which every node has the same infection rate $\beta = 1$, (c) the GLT model in which the node selected has the infection rate $\beta = 10$ and other nodes have infection rate $\beta = 1$. The results are based on $10^4$ runs of simulation. We select 1 node as the initiator and fix it in each run.

FIG. 3: Time evolution of the average cascade size $S$ for among the continuous-time SI model (the blue diamond), the hybrid model (the black square) and the GLT model (the red circle) on the Erdös-Rényi network with $N = 1000$ nodes and the average degree $\langle k \rangle = 4$. (a) The threshold value $\phi^* = 0.25$ and every node has the same infection rate $\beta = 2$ in the GLT and hybrid model. (b) The threshold value $\phi^* = 0.16$ and every node has the same infection rate $\beta = 4$. The fraction of nodes that follow the GLT model and the SI model is 1:1 in the hybrid model. We select 1 node as the initiator and fix it in each run. The curve is based on $10^3$ runs of simulation.

randomly distributed (Fig. 2(c)). Therefore, when using the LT model to generate synthetic spreading data, we may underestimate the complexity of spreading brought by the inherent randomness.

Finally, because the GLT model is based on the continuous-time stochastic process, it is compatible with other continuous-time stochastic processes. We only need to add more reactions in the queue when multiple processes coexist. As an example, we apply the GLT model as a tool to simulate the hybrid spreading process. There are works in epidemics assuming that the disease infection rate can be different under different conditions.\textsuperscript{[51][52]} Hence there will be two infection rates in the system. This feature is hard to implement in the LT model but can be easily added by the GLT model. Here we consider a more interesting situation. In the threshold spreading, to have a global cascade triggered by a single initiator, the threshold value $\phi^*$ needs to be small ($\phi^* \leq 1/\langle k \rangle$).\textsuperscript{[53][54]} This brings questions on how a social spreading, which is usually believed to be the complex contagion, could occur since the threshold of a real social system may not be that small. One explanation is that there can be multiple initiators.\textsuperscript{[12][33]} Alternatively, we may also assume that both simple and complex contagion are active.\textsuperscript{[53]} Here we analyze co-evolutionary contagion which is a hybrid model combining the SI and GLT model. In the model, we assume that there are two types of nodes, one evolves according to the SI model and the other to the GLT model. The result shows that complex contagion alone can not take place, but the global cascade could occur when simple contagion co-exist (Fig. 2(a)). The simulation result demonstrates the model’s capability in combining other spreading mechanisms.

We further find that the dynamics of the hybrid model always locates between the SI model and the GLT model whatever the infected rate $\beta$ is when the threshold $\phi^*$ is smaller than the $1/\langle k \rangle$ (Fig. 2(b)). This shows that the simple contagion accumulates the critical mass for the complex contagion, which allows other nodes to be infected earlier than when the complex contagion alone takes place. It also implies that the simple and complex contagion may demonstrate identical spreading dynamics under certain parameters.

III. CONCLUSION AND DISCUSSION

To summarize, we propose a GLT model for the continuous-time complex contagion process. It overcomes the limitations of the LT model in studying the system evolution. The GLT model can be efficiently implemented by the Gillespie Algorithm. We find that the traditional LT model tends to underestimate the speed of spreading and the randomness of the spreading sequence, compared with cases when the dynamics are more properly defined. We show that the GLT model can be very efficient to simulate more complicated spreading. Taken together, the GLT model we proposed can be a useful tool to study complex contagion, especially when the time evolution of the spreading is the focus. Our result not only sheds light on a series of important questions that were not emphasized previously, but also brings insight into the modeling process of real spreading data. Previous research shows that real spreading process is usually more complex. There are examples of combining multiple spreading mechanisms.\textsuperscript{[55][56]} More importantly, the recovery process is included in real spreading.\textsuperscript{[57]} This urges us to combine the linear threshold model with SIR model, which is readily doable with the GLT model proposed in this paper. These more sophisticated models together with real spreading data would
A generalized linear threshold model
definitely help us understand the underlying patterns in informa-
tion spreading.

Our model also has some shortcomings, the events based
on the spreading dynamics are described as a Poisson ran-
dom process for the GLT model, which may not deal with the
real information spreading well. In the GLT model, whether
a node becomes active depends only on the number of cur-
rent exposures from its neighbors, without memory effects.
The previous records, however, could impact the information
rent exposures from its neighbors, without memory effects.
real information spreading well. In the GLT model, whether
on the spreading dynamics are described as a Poisson ran-
definitely help us understand the underlying patterns in infor-
REFERENCES

DATA AVAILABILITY STATEMENT

This research is supported by the Chongqing Graduate Re-
search and Innovation Project (Grant No. CYB18080), and
the S-Tech Internet Communication Academic Support Plan.

ACKNOWLEDGMENTS

This article is not applicable to this article as data were
generated by the theoretical model.

REFERENCES

1E. M. Rogers, Diffusion of innovations (Simon and Schuster, 2010).
2C. H. Weiss, J. Poncela-Casasnovas, J. I. Glaser, A. R. Pań, S. D. Persell,
D. W. Baker, R. G. Wunderink, and L. A. N. Amaral, “Adoption of a
high-impact innovation in a homogeneous population,” Physical review x
4, 041008 (2014).
3Z.-K. Zhang, C. Liu, X.-X. Zhan, X. Lu, C.-X. Zhang, and Y.-C. Zhang,
“Dynamics of information diffusion and its applications on complex net-
works,” Physics Reports 651, 1–34 (2016).
4J. Borge-Holthoefer, R. A. Baños, S. González-Bailón, and Y. Moreno,
“Cascading behaviour in complex socio-technical networks,” Journal of
Networks 1, 3–24 (2013).
5D. J. Watts, “A simple model of global cascades on random networks,”
Proceedings of the National Academy of Sciences 99, 5766–5771 (2002).
6S. Vosoughi, D. Roy, and S. Aral, “The spread of true and false news
through a social network,” in Proceedings of the ninth ACM SIGKDD
international conference on Knowledge discovery and data mining (ACM,
2003) pp. 137–146.
7P. Singh, S. Sreenivasan, B. K. Szymanski, and G. Korniss, “Threshold-
limited spreading in social networks with multiple initiators,” Scientific rep-
s2, 2330 (2013).
8Q.-H. Liu, F.-M. Lu, Q. Zhang, M. Tang, and T. Zhou, “Impacts of opin-
opinion leaders on social contagions,” Chaos: An Interdisciplinary Journal
of Nonlinear Science 28, 053103 (2018).
9D.-B. Chen, H.-L. Sun, Q. Tang, S.-Z. Tian, and M. Xie, “Identifying in-
fluential spreaders in complex networks by propagation probability dynam-
ics,” Chaos: An Interdisciplinary Journal of Nonlinear Science 29, 033120
(2019).
10Q. Cao, H. Shen, K. Chen, W. Ouyang, and X. Cheng, “Deephawkes: Bridg-
ing the gap between prediction and understanding of information cascades,”
in Proceedings of the 2017 ACM on Conference on Information and Knowl-
gedge Management (2017) pp. 1149–1158.
11T. D. Gillespie, “A general method for numerically simulating the stochastic
time evolution of coupled chemical reactions,” Journal of computational
physics 22, 403–434 (1976).
D. T. Gillespie, “Exact stochastic simulation of coupled chemical reactions,” The journal of physical chemistry 81, 2340–2361 (1977).

J. P. Gleeson, “High-accuracy approximation of binary-state dynamics on networks,” Physical Review Letters 107, 068701 (2011).

J. P. Gleeson, “Binary-state dynamics on complex networks: Pair approximation and beyond,” Physical Review X 3, 021004 (2013).

Z. Shen, S. Cao, W.-X. Wang, Z. Di, and H. E. Stanley, “Locating the source of diffusion in complex networks by time-reversal backward spreading,” Physical Review E 93, 032301 (2016).

S. Bourigault, S. Lamprier, and P. Gallinari, “Representation learning for information diffusion through social networks: an embedded cascade model,” in Proceedings of the Ninth ACM international conference on Web Search and Data Mining (2016) pp. 573–582.

C. Gou, H. Shen, P. Du, D. Wu, Y. Liu, and X. Cheng, “Learning sequential features for cascade outbreak prediction,” Knowledge and Information Systems 57, 721–739 (2018).

M. A. Porter and J. P. Gleeson, “Dynamical systems on networks,” Frontiers in Applied Dynamical Systems: Reviews and Tutorials 4 (2016).

N. Sinitsyn, N. Hengartner, and I. Nemenman, “Adiabatic coarse-graining and simulations of stochastic biochemical networks,” Proceedings of the National Academy of Sciences 106, 10546–10551 (2009).

R. Ramaswamy and I. F. Shalzarin, “A partial-propensity formulation of the stochastic simulation algorithm for chemical reaction networks with delays,” The Journal of chemical physics 134, 014106 (2011).

T. Jia and R. V. Kulkarni, “Intrinsic noise in stochastic models of gene expression with molecular memory and bursting,” Physical review letters 106, 058102 (2011).

S. Qiu, T. Jia, et al., “Quantifying the noise in bursty gene expression under regulation by small ras,” International Journal of Modern Physics C (IJMPC) 30, 1–14 (2019).

N. Kumar, T. Jia, K. Zarringhalam, and R. V. Kulkarni, “Frequency modulation of stochastic gene expression bursts by strongly interacting small ras,” Physical Review E 94, 042419 (2016).