

Fixed-Size Dynamic Scale-Free Networks: Modeling, Stationarity, and Resilience

Yichao Yao¹, Student Member, IEEE, Minyu Feng², Senior Member, IEEE, Matjaž Perc³, Member, IEEE, and Jürgen Kurths⁴

Abstract—Many real-world scale-free networks, such as neural networks and online communication networks, consist of a fixed number of nodes but exhibit dynamic edge fluctuations. However, traditional models frequently overlook scenarios where the node count remains constant, instead prioritizing node growth. In this work, we depart from the assumptions of node number variation and preferential attachment to present an innovative model that conceptualizes node degree fluctuations as a state-dependent random walk process with stasis and variable diffusion coefficient. We show that this model yields stochastic dynamic networks with stable scale-free properties. Through comprehensive theoretical and numerical analyses, we demonstrate that the degree distribution converges to a power-law distribution, provided that the lowest degree state within the network is not an absorbing state. Furthermore, we investigate the resilience of the fraction of the largest component and the average shortest path length following deliberate attacks on the network. By using three real-world networks, we confirm that the proposed model accurately replicates actual data. The proposed model thus elucidates mechanisms by which networks, devoid of growth and preferential attachment features, can still exhibit power-law distributions and be used to simulate and study the resilience of attacked fixed-size scale-free networks.

Index Terms—Degree distribution, evolving networks, Markov processes, scale-free networks.

I. INTRODUCTION

MANY complex physical, biological, and social systems can be abstracted as networks and network science

Received 10 October 2025; accepted 27 December 2025. Date of publication 12 January 2026; date of current version 19 March 2026. The work of Minyu Feng and Yichao Yao was supported in part by the Natural Science Foundation of Chongqing under Grant CSTB2025YITP-QCRCX0007 and in part by the National Natural Science Foundation of China (NSFC) under Grant 62206230. The work of Matjaž Perc was supported by The Slovenian Research and Innovation Agency (Javna agencija za znanstvenoraziskovalno in inovacijsko dejavnost Republike Slovenije) under Grant P1-0403. This article was recommended by Associate Editor L. Sheremetov. (Corresponding author: Minyu Feng.)

Yichao Yao and Minyu Feng are with the College of Artificial Intelligence, Southwest University, Chongqing 400715, China (e-mail: myfeng@swu.edu.cn).

Matjaž Perc is with the Faculty of Natural Sciences and Mathematics, University of Maribor, 2000 Maribor, Slovenia, also with the Community Healthcare Center Dr. Adolf Drolc Maribor, 2000 Maribor, Slovenia, also with the University College, Korea University, Seoul 02841, Republic of Korea, and also with the Department of Physics, Kyung Hee University, Seoul 02447, Republic of Korea.

Jürgen Kurths is with Potsdam Institute for Climate Impact Research, 14437 Potsdam, Germany, and also with the Department of Physics, Humboldt University of Berlin, 10117 Berlin, Germany.

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TSMC.2025.3649867>.

Digital Object Identifier 10.1109/TSMC.2025.3649867

plays a crucial role in studying various complex systems [1], [2]. The modeling of networks that accurately reflect reality has been extensively studied for a long time. Watts and Strogatz [3] introduced the small-world network model, imbuing network models with six degrees of separation characteristic of real-world social networks through random rewiring. Barabási and Albert [4] later discovered the degree accumulation and preferential attachment mechanisms present in real networks, constructing network models endowed with scale-free (SF) properties. Subsequently, an extended SF model was proposed, allowing for edge rewiring within the network [5]. Building upon these foundational studies, various types of network models emerged. Bianconi and Barabási [6] noted that networks could adhere to Bose statistics and undergo Bose–Einstein condensation. Li et al. [7] proposed the local-world evolutionary network model. Later, evolutionary network models with weighted connections and community structures were established [8], along with extensions of local-world network models incorporating node deletion considerations [9]. With the research deepened into real-world networks and the diverse statistical data proliferated, the fitting of network models to the dynamic and stochastic inherent in real networks garnered increasing attention [10], [11]. The structural evolution of the network is exhibited in various important systems [12], [13], such as neural networks [14], Worldwide-Web networks [15], etc. [16], [17], [18]. In response, evolutionary network models based on probabilistic vertex addition and deletion were constructed [19], queueing systems were employed to describe vertex growth and removal in evolutionary networks [20], and various network evolution mechanisms were further investigated [21]. Furthermore, Li et al. [22] conceptualized nodes as queueing systems, proposing a network model with a degree birth–death mechanism. Zeng et al. introduced a network model considering online and hidden vertices based on the birth and death process [23]. In summary, there exists a multitude of dynamic network models such as temporal networks [24], adaptive networks [25], and simplicial activity-driven networks (SADs) [26] and the corresponding extents. All these models provide suitable frameworks for describing dynamic networks. However, many of the aforementioned studies, especially those aimed at modeling SF networks, primarily focus on the modeling and evolution of networks through node growth and preferential attachment mechanisms, while paying less attention to scenarios where the number of nodes remains

constant and only edges exhibit dynamics. Therefore, it is challenging to propose a model to explain the formation of SF properties in some fixed-size networks [27], and developing a model capable of evolving SF properties within such networks is still of significant value. In many real-world systems, such as neural networks, online communication networks, and social media platforms, nodes with high connectivity often experience frequent fluctuations in connections due to their central role. For example, in a neural network, brain regions with numerous connections are more likely to form or sever connections due to fluctuating neural activity. Similarly, in email networks (ENs), central individuals frequently gain or lose contacts based on shifts in communication needs. In social media networks, popular pages tend to have highly dynamic follower bases, with significant fluctuations as users interact differently over time. These cases suggest that in fixed-size networks, the larger a node's degree, the more frequently its connections change, underscoring the need to model degree variation as a dynamic process. To capture this characteristic, we assume that nodes with higher degrees in the network exhibit greater instability in their degree values and model the fluctuations of node degrees using a state-dependent random walk process. In our model, the probabilities of degree increase and decrease are assumed to be equal, resulting in a random but degree-dependent fluctuation process. Through the application of stochastic process theory, we analyze the stationary degree distribution under specific degree fluctuation rates. Different from the traditional SF networks, the model presented in this article does not rely on the "rich get richer" principle. Rather, it is based on the principle that "rich leads to instability." This model achieves a dynamically stable scale-fixed network with a power-law degree distribution in scenarios where growth and preferential attachment mechanisms are not relied upon. Our proposed network model provides a framework for studying the formation and evolution of complex network systems with fixed sizes in real-world scenarios. It offers a perspective for explaining the origin of their power-law characteristics. Through this model, we can better understand, analyze, and simulate the properties and evolutionary patterns of complex networks in practice.

The organization of this article is as follows: in Section II, we provide a detailed description of the evolutionary network model, the modeling algorithm, and a theoretical analysis of its degree distribution. The simulations regarding the degree distribution, reliability, and fidelity of the network are presented in Section III. Conclusions and future work are given in Section IV.

II. MODELING IN DISCRETE AND CONTINUOUS TIME

In our model, we consider a set containing all the degree values that a node can have as the state space, then regard the process of node degree variation as a state-dependent random walk within this state space. The choice of this state-dependent random walk process is motivated by its fundamental properties: unbiased increments and compatibility with Poisson-driven events. These features align well with the dynamics observed in real-world networks. Moreover, the

simplicity and adaptability of the random walk framework make analytical exploration of network degree distributions feasible, and its flexible and tunable diffusion coefficient allows the dynamic properties of the network to be adjusted. To address this random walk process, we need to consider the changes of a node's degree as an independent stochastic process. However, due to the handshake theorem, fluctuations in the degree of a node will affect not only the node itself but also its neighbors. To enable us to treat degree changes as an independent stochastic process, we do not directly alter degrees by adding or removing edges. Instead, we introduce a matching queue as an intermediate state for degree adjustment. When a node's degree increases or decreases, the node itself or its influenced neighbor will be added to the matching queue. Upon the existence of two nodes in the matching queue that can be connected, an immediate connection will be established between them, and they will get out of the queue.

The matching queue adopts a first-in, first-out strategy to ensure that the number of edges connected to a node aligns as closely as possible with its connection demands. This design weakens the impact of a node's edge addition or removal on other nodes in the network, thereby enhancing the independence of each node's degree evolution as a stochastic process. Under this mechanism, a node's degree is influenced by others only in one specific situation: when a node has an edge demand but cannot immediately establish a connection with any other node. In such cases, its degree cannot change instantly. However, as the network evolution progresses over an extended period, this node will eventually meet a compatible node in the matching queue. Due to the queue's priority matching principle, it will connect to this node as soon as possible. Therefore, while some nodes' degree change may experience a delay, this delay is unlikely to significantly affect the steady-state degree distribution of the network when the evolution time becomes sufficiently long. The precise extent of this impact can be assessed through the simulations.

This mechanism can effectively mitigate the influence of the handshake theorem, allowing the degree evolution process of each node to remain largely independent, and it is reasonable because relationships in real networks do not only disappear in pairs. For example, in communication networks, when a high-traffic router loses connection to a neighboring router due to congestion or failure, it will still seek to establish new connections with other routers to maintain traffic flow. Similarly, in power grids, a substation may disconnect from a load to balance the grid during a fluctuation, but it will continuously seek new connections with other parts of the grid to restore balance. In cloud computing, a heavily loaded server may drop some connections to optimize performance, but it will seek new tasks or clients as the load fluctuates. These systems require constant dynamic adjustment, and the introduction of a matching queue for degree adjustments helps reflect this real-world behavior of continuously forming and dissolving connections to maintain optimal network performance.

In summary, the network model proposed in this article is built upon the following four fundamental assumptions below.

1) The degree fluctuation process of nodes in the network can be viewed as a state-dependent random walk process with a variable diffusion coefficient, where the drift coefficient increases with the node's degree.

2) When a node's degree increases, or its neighbor's degree decreases, new edges are not immediately established. Instead, the node is added to a matching queue, and a new edge is formed only when the node is matched within the queue.

3) The network does not allow isolated nodes, multiple edges, or self-loops. Therefore, nodes with a degree of 1 do not lose their remaining edge, and nodes already connected in the matching queue will not be reconnected.

4) The time a node spends in the matching queue waiting for a new connection to be established does not significantly affect the overall degree distribution of the network.

In Sections II-A and II-B, we divide the modeling process into two cases: discrete time and continuous time. For each case, we present the specific modeling algorithms and analyze the degree distributions accordingly.

A. Discrete Time Network Modeling

For the discrete-time case, we assign a stable probability $S(k)$ to each point in the state space, which means the probability that a node with k degrees remaining at the state of having k degrees after one time step is $S(k)$. That is to say, let P be the one-step state transition matrix for degree changes, and let P_{ij} denote the probability that a node with degree i transitions to degree j in one time step, then $S(k) = P_{kk}$. After that, we consider the velocity (the rate of change of the degree) of the particle's (node's) movement dependent on its position (current degree value), and fix the step size at 1. Meaning that a node's degree can only increase or decrease by 1 at each time if it changes. Furthermore, for the boundary conditions of state transition, considering that in many real-world networks, nodes with a low degree (e.g., degree $k = 1$) are often subject to functional constraints. For instance, nodes cannot typically become completely isolated (degree $k = 0$) due to the need for minimal connectivity in maintaining network participation or basic functionality. In addition, in the event of a failed disconnect or connection attempt, such nodes are more likely to remain in their current state rather than actively seek new connections, reflecting a conservative behavioral tendency or resource limitations. These considerations motivate us to design the transition probabilities of boundary nodes as $P_{1,1} = S(1) + (1 - S(1))/2$, and $P_{n-1,n-1} = S(n-1) + (1 - S(n-1))/2$. According to the previously discussed settings of the matching queue, the process of degree variation can be regarded as a random walk process with a diffusion coefficient that varies spatially, which possesses the Markov property. Furthermore, based on the previously mentioned assumption of "the rich get richer," the probability that a node chooses to change its degree in a given time step needs to be increased with its degree. Fig. 1 presents an example of the network modeling process. As shown in the figure, nodes exhibit three basic behaviors: losing half an edge, gaining half an edge, and matching to establish a new complete edge.

For nodes that decide to change, since we treat the degree fluctuation process as a random walk process with the step

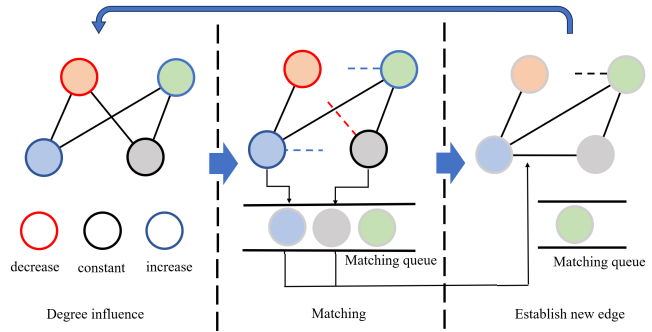


Fig. 1. Fundamental steps of network evolution. The circles of different colors represent distinct nodes, with red, black, and blue circles indicating nodes that are about to lose a degree, maintain their degree, or increase their degree, respectively. The dashed lines represent unconnected half-edges, and nodes with half-edges will be added to the matching queue.

size restricted to 1, another random selection is made to either increase or decrease the node's degree with equal probability. Furthermore, for nodes whose degree increases by 1, they are directly added to the matching queue. For nodes whose degree decreases by 1, one of their edges is randomly selected and removed, and the neighbor connected by that edge is added to the matching queue. During the evolution process, each iteration of the network begins with each node determining changes to its degree and ends once all connectable nodes in the matching queue have established connections. The complete pseudocode for the process, with the additional constraints that nodes are prohibited from forming multiple edges (duplicate edges) or self-loops, and are prevented from becoming isolated (i.e., the degree is bounded between 1 and $n-1$), is as the Algorithm 1.

Further, we use a Markov method to analyze the evolution of the degree distribution in the discrete-time case. The maximum degree is $n-1$, and the minimum degree is 1, thus the state space includes $n-1$ discrete positions for nodes to reside. To keep the degree values between 1 and $n-1$, we need to accumulate the probability of transitioning a node's degree from 1 to 0, and from $n-1$ to n , to the probabilities of remaining at 1 and $n-1$, respectively.

Theorem 1: If $S(k) < 1$ for all k , the limiting degree distribution Π of the network exists and converges to

$$\begin{cases} \pi_k = \pi_1 \frac{1 - S(1)}{1 - S(k)}, & 1 < k < n \\ \pi_1 = 1 / \left(1 + \frac{1 - S(1)}{1 - S(2)} + \dots + \frac{1 - S(1)}{1 - S(n-1)} \right). \end{cases} \quad (1)$$

Proof: With these constraints, we can derive a transition matrix for a single node between different degrees within one time step as follows:

$$P = \begin{pmatrix} \frac{S(1)+1}{2} & \frac{1-S(1)}{2} & 0 & \dots & 0 \\ \frac{1-S(2)}{2} & S(2) & \frac{1-S(2)}{2} & \dots & 0 \\ 0 & \frac{1-S(3)}{2} & S(3) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \frac{S(n-1)+1}{2} \end{pmatrix}. \quad (2)$$

Algorithm 1 Algorithm of Discrete Time Network Model**Input:** T, N, G**Output:** G

```

1: MatchingQueue ← []
2: for each  $t \in (1, T)$  do
3:   for each  $n \in N$  do
4:     RandomNumber ← GenerateRandomNumber(0, 1)
5:     if  $RandomNumber > \frac{1+S(k(n))}{2}$  and  $k(n) < |N| - 1$ 
6:       then
7:         Append(MatchingQueue,  $n$ )
8:       else if  $RandomNumber < \frac{1-S(k(n))}{2}$  and  $k(n) > 1$ 
9:         then
10:           $m \leftarrow \text{ChoiceRandomNeighbor}(n)$ 
11:          RemoveEdge( $m, n$ )
12:          Append(MatchingQueue,  $m$ )
13:        else
14:          CONTINUE
15:        end if
16:      end for
17:    for each  $n$  in MatchingQueue do
18:      for each  $m$  in MatchingQueue do
19:        if  $n \neq m$  and NotHaveEdge( $n, m$ ) then
20:          Connect( $n, m$ )
21:          Remove(MatchingQueue,  $m$ )
22:          Remove(MatchingQueue,  $n$ )
23:        end if
24:      end for
25:    end for
26:  end for
27: return G

```

According to the properties of tridiagonal matrices, when all the elements on the upper and lower diagonals of the matrix are positive, i.e., when $S(k) < 1$ for every k , the corresponding Markov chain will be ergodic, and its absolute distribution will be equal to the stationary distribution. In that case, define π_i as the proportion of nodes with degree i and $\Pi = (\pi_1, \pi_2, \dots, \pi_{n-1})$ as the stationary distribution, Π will satisfy the following equation:

$$\Pi P = \Pi. \quad (3)$$

Calculating the stationary distribution means to solve the following system of equations:

$$\begin{cases} (\pi_1, \pi_2, \dots, \pi_{n-1}) P = (\pi_1, \pi_2, \dots, \pi_{n-1}) \\ \pi_1 + \pi_2 + \dots + \pi_{n-1} = 1. \end{cases} \quad (4)$$

When we expand the state-transition equation formula for the degree distribution evolution, we can express it as follows:

$$\begin{cases} \pi_1 = (\pi_1 (S(1) + 1) + \pi_2 (1 - S(2))) / 2 \\ \pi_2 = (\pi_1 (1 - S(1)) + 2\pi_2 S(2) + \pi_3 (1 - S(3))) / 2 \\ \vdots \\ \pi_{n-1} = (\pi_{n-2} (1 - S(n-2)) + \pi_{n-1} (S(n-1) + 1)) / 2 \\ \pi_1 + \pi_2 + \dots + \pi_{n-1} = 1. \end{cases} \quad (5)$$

From the first and second lines of the above system of equations, we can derive

$$\begin{cases} \pi_2 = \pi_1 \frac{1 - S(1)}{1 - S(2)} \\ \pi_3 = \pi_2 \frac{1 - S(2)}{1 - S(3)}. \end{cases} \quad (6)$$

By mathematical induction, we can derive the following system of equations:

$$\begin{cases} \pi_k = \pi_1 \frac{1 - S(1)}{1 - S(k)}, \quad 1 < k < n \\ \pi_1 = 1 / \left(1 + \frac{1 - S(1)}{1 - S(2)} + \dots + \frac{1 - S(1)}{1 - S(n-1)} \right). \end{cases} \quad (7)$$

Therefore, combining (1)–(7), the results follow. ■

It is evident that the characteristic function of the stationary distribution will primarily depend on the spatial variation of the stay probability $S(k)$.

Specifically, based on the concept of “rich leads to instability,” we can assume the stay probability $S(k)$ as

$$S(k) = 1 - \left(\frac{k+c}{n} \right)^a. \quad (8)$$

In this scenario, nodes with larger degrees have a lower probability of remaining at that degree within a time step, ensuring that high-degree nodes are more likely to experience changes. Here, the constant c serves as a smoothing parameter. When $c = -1$ or minor, one or a set of the minimum degree in the state space becomes an absorbing state. When $c = 1$ or greater, one or a set of the maximum degrees in the state space becomes a reflective state. The parameter a is used to adjust the level of nonlinearity for the variation of the stay probability against the degree. For $0 < k < n$, the stationary degree distribution is given by

$$\pi_k = \pi_1 \left(\frac{c+1}{c+k} \right)^a. \quad (9)$$

It is obvious that in this case, as c approaches 0, the stationary degree distribution converges to a power-law distribution.

B. Continuous Time Network Modeling

To extend our approach to the case of continuous time, we now consider the increase or decrease in degree as a single event occurring on a node. Therefore, the arrival of events on a single node can be seen as a nonhomogeneous Poisson process, where the interarrival times between two events follow an exponential distribution [28], and the parameter of this exponential distribution is dependent on the current degree of this node.

Similar to the discrete time intervals, for continuous time, we also define a matching queue as an intermediate process for node degree fluctuations. Specifically, the network evolution process can be divided into the following four steps.

- 1) At the initial time, each node generates the arrival time of the next degree fluctuation event based on the exponential distribution, in which the exponential parameter is determined by its degree.

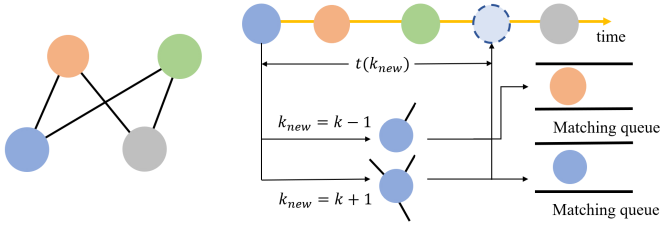


Fig. 2. Degree fluctuation process of nodes in the continuous-time model. Nodes of different colors along the time axis represent the chronological occurrence of their degree fluctuation events. When a node's degree fluctuation event occurs, based on the status of edge disconnections and connections, either the node itself or one of its neighbors will be added to the matching queue. A round of matching will then be performed on the nodes in the queue, and based on their new degree values, the arrival times of their subsequent degree fluctuation events will be generated along the time axis.

- 2) Select the node with the smallest arrival time, and with equal probability, either increase or decrease its degree by 1. When the degree increases, add the node to the matching queue. When the degree decreases, disconnect the node from a random neighbor and add the neighbor to the matching queue.
- 3) Connect all nodes in the matching queue that can have an edge and remove them from the matching queue.
- 4) Subtract the arrival time of the just-updated nodes from the arrival time of all the nodes, and generate a new arrival time of the next degree fluctuation event for the just-updated node based on its new degree. Return to step 2).

Fig. 2 illustrates the main steps of the algorithm, and Algorithm 2 is the pseudocode for the complete modeling algorithm.

In particular, we do not allow the emergence of multiple edges and self-loops in this work, thus, the same node and the node pairs that already have an edge in the matching queue will not be connected. Additionally, due to the boundary setting of the state space, any node whose degree reaches 0 or n will be reverted to its previous state. The continuous-time degree variation process also exhibits Markovian properties. Let π_i be the proportion of nodes with degree i , $\Pi = (\pi_1, \pi_2, \dots, \pi_{n-1})$ as the stationary distribution, and $\lambda(k)$ represent the parameter of the Poisson process associated with the degree k .

Theorem 2: If $\lambda(k) > 0$ for all k , the limiting degree distribution of the network exists and converges to a stable distribution

$$\begin{cases} \pi_k = \frac{\pi_1 \lambda(1)}{\lambda(k)}, & 1 < k < n \\ \pi_1 = 1 / \left(1 + \frac{\lambda(1)}{\lambda(2)} + \dots + \frac{\lambda(1)}{\lambda(n-1)} \right). \end{cases} \quad (10)$$

Proof: For state k in the state space, there exists a transition rate

$$\begin{cases} \pi_k \lambda(k), & 1 < k < n \\ \frac{\pi_k \lambda(k)}{2}, & k = 1 \text{ or } k = n - 1 \end{cases} \quad (11)$$

Algorithm 2 Algorithm of Continuous Time Network Model

Input: T, N, G

Output: G

```

1: MatchingQueue  $\leftarrow []$ 
2:  $t \leftarrow []$ 
3: for each  $n$  in  $N$  do
4:    $t_n \leftarrow \text{GenerateExponential}(\lambda(k(n)))$ 
5:   Append(TimeList,  $t_n$ )
6: end for
7: for each  $t \in (1, T)$  do
8:    $n \leftarrow \text{GetMinIndex}(\text{TimeList})$ 
9:   RandomNumber  $\leftarrow \text{GenerateRandomNumber}(0, 1)$ 
10:  if RandomNumber  $> \frac{1}{2}$  and  $k(n) < |N| - 1$  then
11:    Append(MatchingQueue,  $n$ )
12:  else if  $k(n) > 1$  then
13:     $m \leftarrow \text{ChoiceRandomNeighbor}(n)$ 
14:    RemoveEdge( $m, n$ )
15:    Append(MatchingQueue,  $m$ )
16:  end if
17:  for each  $n$  in MatchingQueue do
18:    for each  $m$  in MatchingQueue do
19:      if  $n \neq m$  and NotHaveEdge( $n, m$ ) then
20:        Connect( $n, m$ )
21:        Remove(MatchingQueue,  $m$ )
22:        Remove(MatchingQueue,  $n$ )
23:      end if
24:    end for
25:  end for
26:  for each  $m$  in  $N$  do
27:    TimeList( $m$ )  $\leftarrow \text{TimeList}(m) - \text{TimeList}(n)$ 
28:  end for
29:   $t_n \leftarrow \text{GenerateExponential}(\lambda(k(n)))$ 
30:  TimeList( $n$ )  $\leftarrow t_n$ 
31: end for
32: return  $G$ 

```

When the system is in a steady state, the rate of transitions out of each state is equal to the rate of transitions into the state. On the condition that a node can only change its degree by 1 at a time, we have the following balance condition:

$$\begin{cases} \pi_1 \lambda(1) = \pi_2 \lambda(2) \\ \pi_k \lambda(k) = \frac{\pi_{k-1} \lambda(k-1)}{2} + \frac{\pi_{k+1} \lambda(k+1)}{2}, & 1 < k < n - 1 \\ \pi_{n-1} \lambda(n-1) = \pi_{n-2} \lambda(n-2). \end{cases} \quad (12)$$

By mathematical induction, we get

$$\begin{cases} \pi_k = \frac{\pi_1 \lambda(1)}{\lambda(k)}, & 1 < k < n \\ \pi_1 = 1 / \left(1 + \frac{\lambda(1)}{\lambda(2)} + \dots + \frac{\lambda(1)}{\lambda(n-1)} \right). \end{cases} \quad (13)$$

Therefore, combining (11)–(13), the results follow. ■

When the formula holds, the transition rate for each degree in the state space is equal to the inflow rate. As we design the

function about the Poisson parameter based on the concept of “rich leads to instability,” λ will be increased with the degree. In this article, we assume the $\lambda(k)$ as

$$\lambda(k) = \left(\frac{k+c}{n}\right)^a. \quad (14)$$

$\lambda(k)$ is a monotonically increasing function of a node’s degree k , ensuring that high-degree nodes are more likely to experience changes.

Furthermore, let $\pi_k(t)$ be the proportion of nodes with degree k at time t we can perform the analysis using the Fokker–Planck equation.

Corollary 1: If $\lambda(k) = ((k+c)/n)^a$, the steady-state distribution is a power law distribution, and the multiple steady-state solutions do not exist.

Proof: Based on the assumptions of Poisson process, it is straightforward to derive that the variance of Δk

$$\text{Var}[\Delta k] = \lambda(k) \Delta t. \quad (15)$$

Thus, the diffusion term $g(k)$ in the Fokker–Planck equation is given by

$$g(k)^2 = \lambda(k) = \left(\frac{k+c}{n}\right)^a. \quad (16)$$

Since the drift coefficient $f(x) = 0$, the Fokker–Planck equation is

$$\begin{aligned} \frac{\partial \pi_k(t)}{\partial t} &= -\frac{\partial}{\partial k} [f(k) \pi_k(t)] + \frac{\partial^2}{\partial k^2} \left[\frac{g(k)^2}{2} \pi_k(t) \right] \\ &= \frac{\partial^2}{\partial k^2} [\lambda(k) \pi_k(t)]. \end{aligned} \quad (17)$$

Substituting in the concrete $\lambda(k)$, would be

$$\frac{\partial \pi_k(t)}{\partial t} = \frac{\partial^2}{\partial k^2} \left[\left(\frac{k+c}{n}\right)^a \pi_k(t) \right]. \quad (18)$$

In the steady-state condition, $(\partial \pi_k(t))/\partial t = 0$, the equation is reduced to

$$\frac{\partial}{\partial k} \left[\left(\frac{k+c}{n}\right)^a \frac{\partial \pi_k}{\partial k} \right] = 0. \quad (19)$$

Performing the integration, we obtain

$$\pi_k = \frac{C_1}{\left(\frac{k+c}{n}\right)^a} + C_2. \quad (20)$$

This indicates that the steady-state distribution follows a power-law distribution with an exponent of $-a$. The distribution exhibits a single peak, thus, no multiple steady-state solutions exist. The results follow. ■

According to Theorem 2, for $0 < k < n$, the degree distribution is

$$\pi_k = \pi_1 \left(\frac{c+1}{k+c}\right)^a. \quad (21)$$

The inclusion of c ensures that when $c > 0$, the 1-degree state does not become an absorbing state, and the parameter a is used to adjust the level of nonlinearity for the variation of the λ against degree. The degree distribution will approach an exponential when c approaches 0.

III. SIMULATION

In this section, we conduct simulation experiments to validate the previously proposed theories and models. We perform attack tests on the networks generated by the model to observe their stationarity and resilience, and use the model to fit real-world data.

In this work, all the simulations are implemented using Python 3.10. For the simulation of degree distributions and network attacks, we employed the `dense_gnm_random_graph()` function from the `networkx` package to generate an initial random network with a size of 1000 and an average degree of 4. During the network’s evolution, the `random()` function from the `random` package was employed to generate random numbers to determine the probabilistic behavior of nodes at each step, while the `choice()` function from the same package was used to select nodes for removal during random attacks. For fitting experiments with real networks, we used the `optimize.curve_fit()` function from the `scipy` package to obtain the corresponding values of parameters a and c for the real networks. Subsequently, we generated initial networks of identical size to the real-world networks and an average degree of 4 for further evolution.

In Section III-A, we simulate the network’s evolution, obtain its degree distribution, and compare it with the theoretical degree distribution. In Section III-B, we simulate the effects of attacks on the network and its resilience. Finally, in Section III-C, we apply the proposed network model to simulate real-world networks and demonstrate the fitting results for various topological properties.

A. Degree Distribution and Fitting Results

The degree distributions of the network after evolving to a steady state in discrete time are shown in Fig. 3. Fig. 3(a)–(c) correspond to the results when a is set to 1, 2, and 0.5, respectively, with c fixed at 0.1. The initial network is an undirected graph with 1000 nodes and an average degree of 4. After 10^5 iterations, the degree distributions were averaged over ten independent runs of the evolution process to obtain the results displayed in Fig. 3. It can be observed from Fig. 3 that the numerical simulations align well with the theoretical predictions. The lower-degree regions adhere closer to the theoretical values than the higher-degree regions. These results validate the proposed theoretical framework.

Fig. 4 displays the degree distribution of networks at steady state for different values of a in continuous time. Fig. 4(a)–(c) corresponds to the results when a is set to 1, 2, and 0.5, respectively, with c fixed at 0.1. The initial network is an undirected graph with 1000 nodes and an average degree of 4. After 8×10^7 iterations, the degree distributions were averaged over ten independent runs of the evolution process to obtain the results displayed in Fig. 4. From Fig. 4, it can be observed that the numerical simulation results closely match theoretical results, this suggests that the dynamic network model with continuous time can also achieve a good fit with theoretical values, and generate networks with stable power-law degree distributions.

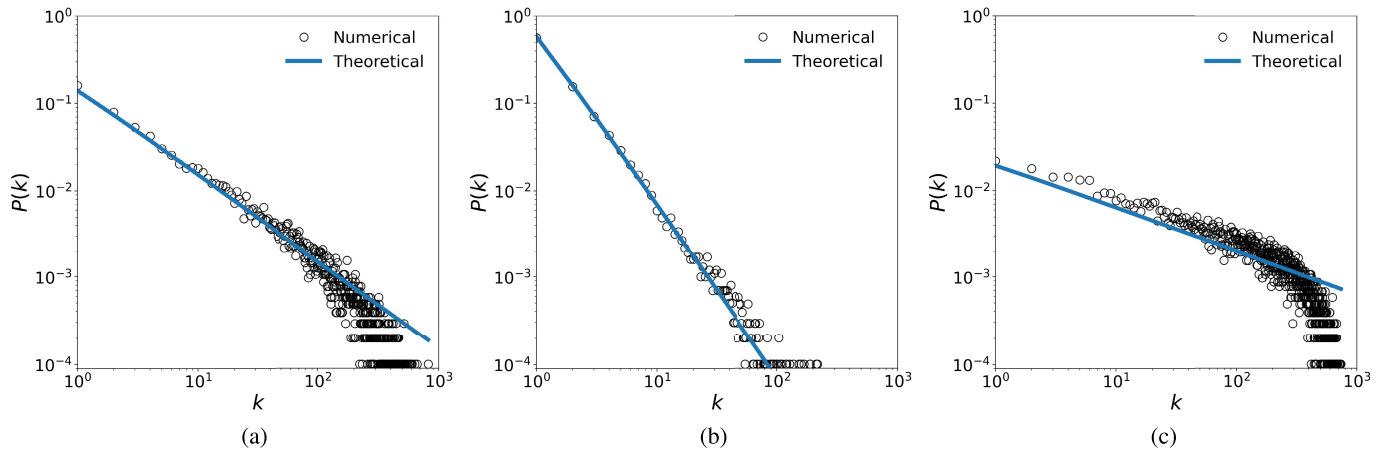


Fig. 3. Degree distribution of the network as it evolves to a stable state in discrete time. The parameter a is depicted as follows: (a) $a = 1$, (b) $a = 2$, (c) $a = 0.5$. In the figure, black circles represent the results from numerical simulations, while the blue lines represent the theoretical values.

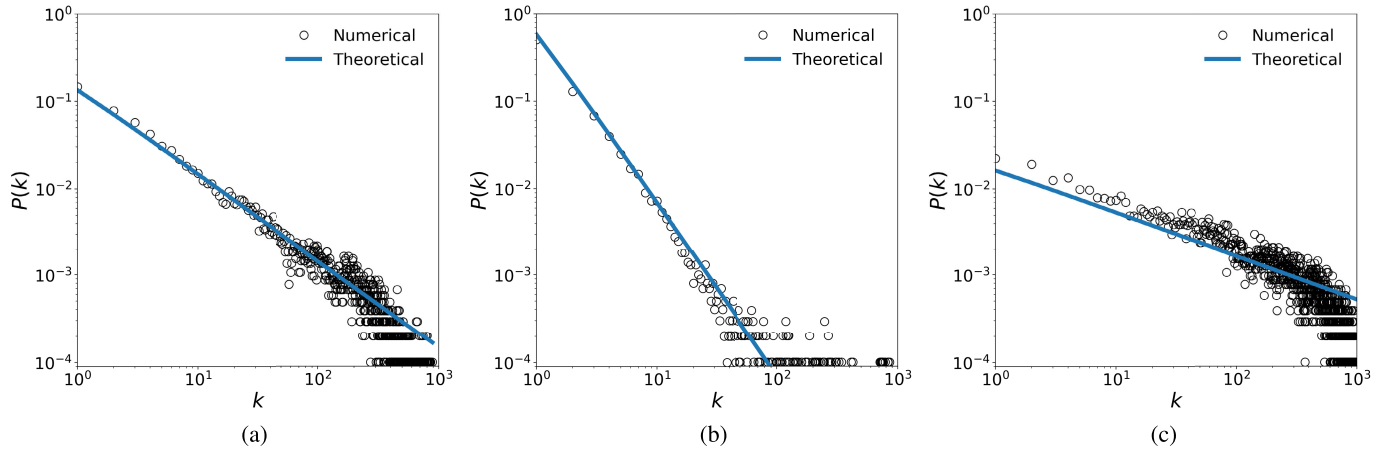


Fig. 4. Degree distribution of the network as it evolves to a stable state in continuous time. The parameter a is depicted as follows: (a) $a = 1$, (b) $a = 2$, (c) $a = 0.5$. In the figure, black circles represent the results from numerical simulations, while the blue lines represent the theoretical values.

Although the theoretical results in (9) and (21) are obtained without approximation, the network evolution involves inherent stochasticity. In the discrete-time model, nodes probabilistically increase or decrease their degrees, while in the continuous-time model, the interevent times follow a Poisson process. Therefore, Figs. 3 and 4 not only verify the agreement between theory and simulation, but also illustrate how the stochastic fluctuations influence the degree distribution, confirming the robustness of the model.

Fig. 5 illustrates the variation in the number of stranded nodes in the matching queue under different parameter settings. Each point in the figure is separated by 5000 iterations. From the figure, it can be observed that in the discrete-time scenario, only when $a = 1$ does the number of stranded nodes become relatively high, peaking at nearly 20. For $a = 0.5$ and $a = 2$, the number of stranded nodes remains consistently below 10. In the continuous-time scenario, the number of stranded nodes is higher for $a = 2$, remaining under 15, whereas for $a = 1$ and $a = 0.5$, there are at most three nodes in the queue. Notably, throughout the whole evolution process,

the number of stranded nodes in the matching queue never exceeds 2% of the total number of nodes in the network, more than 98% of the nodes can be considered to have independent degree evolution processes. This indicates that the theoretical analysis is largely valid, as the impact of node stranding in the matching queue on the overall network degree distribution is negligible.

To measure the similarity between the theoretical and empirical degree distributions, we utilized Kullback–Leibler (KL) divergence and Jensen–Shannon (JS) divergence [29]. The KL divergence is used to quantify how one probability distribution diverges from a second, reference distribution. It is expressed as

$$D_{\text{KL}}(\Pi || \Pi^*) = \sum_x \pi_x \log \left(\frac{\pi_x}{\pi_x^*} \right) \quad (22)$$

where Π and Π_x^* represent the empirical and theoretical stationary degree distributions, respectively. The KL divergence is nonnegative and ranges from 0 (indicating identical distributions) to $+\infty$.

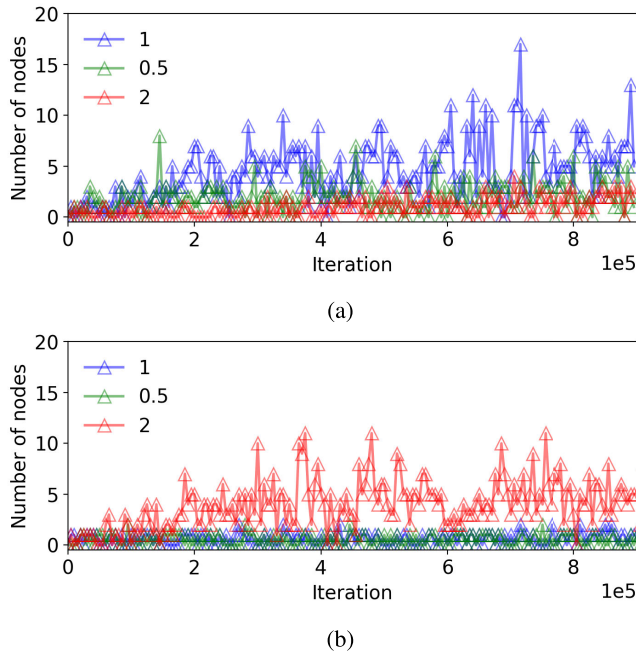


Fig. 5. Number of nodes in the matching queue over iterations. (a) and (b) Results for discrete and continuous time, respectively. In the figures, the blue, green, and red lines correspond to the results obtained for $a = 1$, $a = 2$, and $a = 0.5$, respectively.

TABLE I
SIMILARITY OF THEORETICAL DISTRIBUTIONS AND
SIMULATION BY KL AND JS

a	continuous		discrete	
	KL	JS	KL	JS
1	0.053	0.187	0.141	0.170
2	0.022	0.200	0.043	0.215
0.5	0.070	0.236	0.602	0.196

The JS divergence, which is the symmetric average of the KL divergence in both directions, is expressed as

$$D_{JS}(\Pi || \Pi^*) = \frac{1}{2} D_{KL} \left(\Pi || \frac{\Pi + \Pi^*}{2} \right) + \frac{1}{2} D_{KL} \left(\Pi^* || \frac{\Pi + \Pi^*}{2} \right). \quad (23)$$

Unlike KL divergence, the JS divergence is always finite and bounded within the range $[0, 1]$, where 0 indicates perfect similarity and 1 indicates maximal divergence.

These two metrics provide an effective framework for measuring the differences between the theoretical degree distribution and the simulation results, as well as the discrepancies between the simulation results and real-world data. The KL divergence and JS divergence values, calculated for the theoretical degree distributions and the actual values obtained from simulations under different values of a , are shown in Table I.

From the results presented in Table I, we can observe that the continuous-time model consistently produces results that align more closely with the theoretical degree distribution compared to the discrete-time model. Specifically, when $a = 2$, both the KL divergence and JS divergence reach

their minimum values, indicating the closest match between the experimental results and theoretical values. In contrast, when $a = 0.5$, both divergences reach their maximum values.

Despite these variations, the maximum KL and JS divergence values observed across all results are only 0.602 and 0.236, respectively. This indicates that the degree distribution of the network model fits the theoretical distribution well. Furthermore, as the value of a increases, the simulated results exhibit an even closer fit to the theoretical values, and the continuous-time model demonstrates a stronger correspondence and better modeling performance.

B. Resilience of Network Under Attacks

Fig. 6(a) illustrates the variation of the largest connected component P_{LCC} with respect to the parameter a , where c is fixed at 0.1, for both the continuous-time and discrete-time models. Fig. 6(b) and (c) presents the changes in P_{LCC} of the continuous-time and discrete-time models, respectively, under random and deliberate attacks of different intensities. All data points in Fig. 6 represent the average results obtained from experiments conducted on ten independently generated networks with identical parameter settings. The attack ratio refers to the proportion of nodes removed from the network. Deliberate attacks sequentially remove nodes with the highest degrees, whereas random attacks remove nodes randomly. According to the results in Fig. 6(a), the network is fully connected when $a < 1.5$. When $a > 1.5$, P_{LCC} decreases as a increases, showing similar trends in continuous and discrete time scenarios. Fig. 6(b) and (c) shows that, like other SF networks, the network is robust against random attacks but vulnerable to targeted attacks, with vulnerability increasing as parameter a grows. Additionally, it is observed that for the same attack frequency, the network model in discrete time demonstrates relatively stronger robustness compared to the continuous-time model.

Table II presents more comprehensive results. In addition to targeting nodes based on degree, it also includes results for attacks targeting nodes with the highest betweenness centrality. The table further illustrates changes in the relative shortest path length L , the size of the largest k-shell $k_{\text{core-max}}$, and the modularity of the network following the attacks. The L is pressed as

$$L = \frac{l}{L_0} \quad (24)$$

where l is the average shortest path length of the current largest connected component, and L_0 is the average shortest path length of the network at its initial state. All data in Table II were obtained by conducting deliberate attacks on a network with $a = 1$, 1000 nodes, and evolved to a stable state. From the data, it can be observed that the effects of attacks based on degree are nearly identical to those based on centrality. Networks generated by the discrete-time model retain a higher P_{LCC} under attack, while those generated by the continuous-time model exhibit slightly higher modularity.

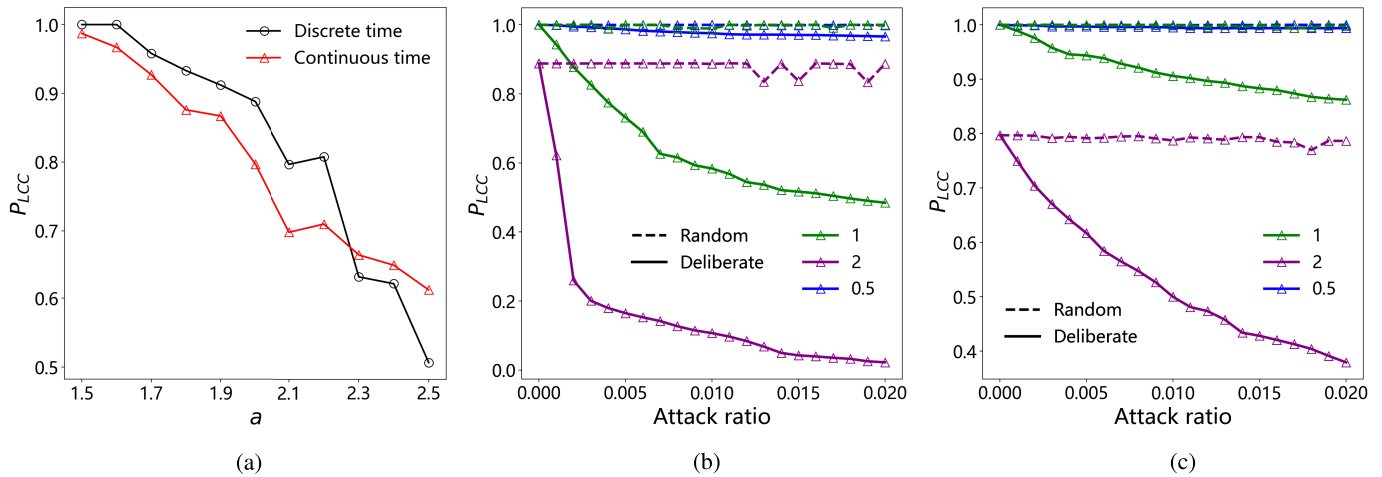


Fig. 6. Impact of power-law exponent a and attack ratio on (a) P_{LCC} , (b) and (c) is for the model of continuous time and discrete time, respectively. (a) Black and red represent results from discrete and continuous time, respectively. (b) and (c) Green, purple, and blue represent networks generated by continuous time models with a values of 1, 2, and 0.5, respectively, while solid and dashed lines represent deliberate attacks and random attacks, respectively.

TABLE II
NETWORK PROPERTIES UNDER DELIBERATE ATTACKS

	attack ratio	continuous				discrete			
		P_{LCC}	L	$k_{core-max}$	modularity	P_{LCC}	L	$k_{core-max}$	modularity
degree	0.01	0.583	0.926	144	0.163	0.906	1.041	73	0.075
	0.02	0.484	0.909	134	0.147	0.862	1.063	63	0.082
	0.03	0.400	0.879	124	0.132	0.829	1.090	55	0.089
centrality	0.01	0.582	0.925	144	0.163	0.907	1.040	73	0.077
	0.02	0.484	0.909	134	0.147	0.861	1.064	63	0.082
	0.03	0.400	0.87	124	0.136	0.824	1.089	55	0.090

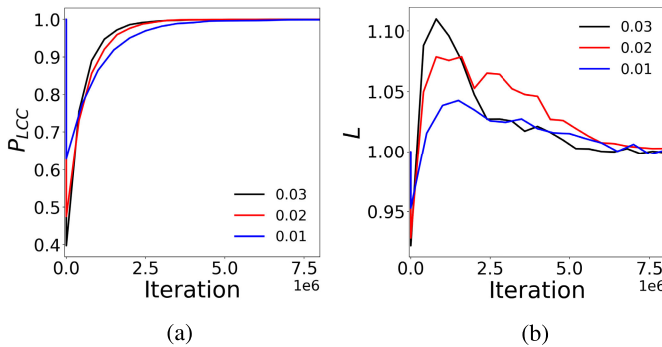


Fig. 7. Recovery of (a) P_{LCC} and (b) L over iterations after different ratios of deliberate attack in the continuous-time network model. L represents the ratio of the current average path length to the initial average path length. The network is generated by the continuous time model with $a = 1$. Blue, red, and black lines represent the results for the attack ratios of 0.01, 0.02, and 0.03, respectively.

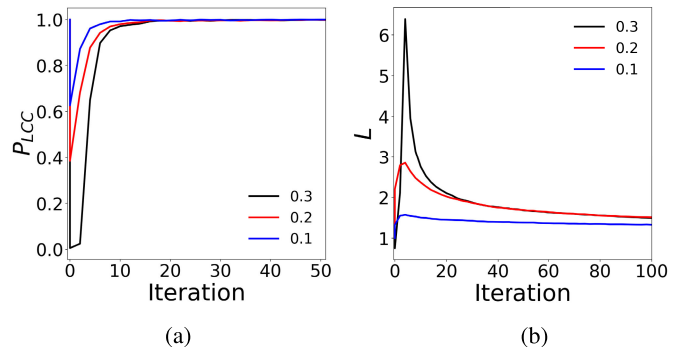


Fig. 8. Recovery of (a) P_{LCC} and (b) L over iterations after different ratios of deliberate attack in the discrete-time network model. L represents the ratio of the current average path length to the initial average path length. The network is generated by the discrete-time model with $a = 1$. Blue, red, and black lines represent the results for the attack ratios of 0.1, 0.2, and 0.3, respectively.

Furthermore, since the network continuously converges during its evolution, there is a recovery after an attack. Figs. 7 and 8 display the resilience of the network following an attack. Figs. 7 and 8(a) and (b), respectively, depict the changes in the largest connected component P_{LCC} and the relative shortest path length L under different deliberate attack frequencies for the continuous and discrete time models, the data points in Figs. 7 and 8 represent the average results obtained from attacks performed on ten independently generated networks with identical parameter settings. Although the recovery of P_{LCC} is a natural result of the rewiring mechanism, Figs. 7–9

are presented to illustrate the detailed recovery trajectories of P_{LCC} under different parameter configurations. By comparing the recovery processes and timescales associated with various values of attack ratio, we can assess how it affects the network's resilience. In addition, comparing the recovery patterns of P_{LCC} and the relative average shortest path length L provides further insight into the distinct temporal responses of global connectivity and path efficiency during the restoration process. The results shown in Figs. 7 and 8 indicate that, even though the size of our network is fixed, it can naturally recover its topological properties after being subjected to attacks, similar to some evolving networks.

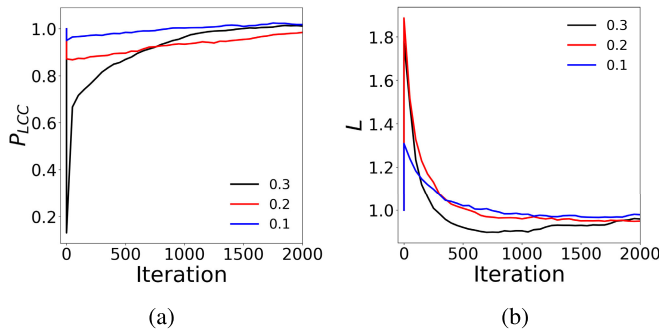


Fig. 9. Recovery of (a) P_{LCC} and (b) L over iterations after different ratios of deliberate attack in a dynamic network model based on a birth-death process. L represents the ratio of the current average path length to the initial average path length. The network is generated by the continuous time model with $a = 1$. Blue, red, and black lines represent the results for the attack ratios of 0.1, 0.2, and 0.3, respectively.

For both the continuous and discrete time networks, the parameters are set to $a = 1$ and $c = 0.1$. The attack frequencies for the continuous-time network are set to 0.01, 0.02, and 0.03, while for the discrete-time network, they are 0.1, 0.2, and 0.3. From Figs. 7 and 8, it can be observed that P_{LCC} gradually recovers after the attack, and L initially increases with the growth of P_{LCC} . After P_{LCC} recovers to its maximum value, then L gradually decreases to 1 as the network evolves. The total time for the average shortest path length to fully recover is longer than that for connectivity. Additionally, a notable pattern is observed in the recovery after high-ratio attacks: although a higher attack ratio results in a greater reduction in network connectivity, the resilience is stronger in that case compared to when the attack ratio is lower. This is because a higher attack ratio creates a larger hub node gap, reducing competition pressure among other nodes, and allowing new hub nodes to appear more quickly. Additionally, we introduced a dynamic scale-free network based on a birth-death process for comparison [30], with the results presented in Fig. 9. The results in Fig. 9 are also averaged over independent attacks on ten independently generated networks with identical parameter settings, the attacks were deliberate, with attack rates of 0.1, 0.2, and 0.3. From the figure, it can be observed that this network demonstrates stronger attack resistance compared to our model. However, the regularity of its recovery process is less pronounced. Specifically, L is most affected at an attack rate of 0.2, while the P_{LCC} recovery curve at an attack rate of 0.3 differs significantly from those at attack rates of 0.2 and 0.1.

C. Fitting Real Networks

Figs. 10–12 and Table III demonstrate the model’s accuracy in replicating the real networks. We utilized a fly brain network (FBN) from networkrepository.com [31], [32], an email-Eu-core temporal network (EN) from snap.stanford.edu, and a mutually liked Facebook pages network (FPN) about foods from networkrepository.com [31]. The FBN has 1781 nodes and 33 641 edges, the EN was generated using email data from a large European research institution. The e-mails only represent communication between institution members, it has

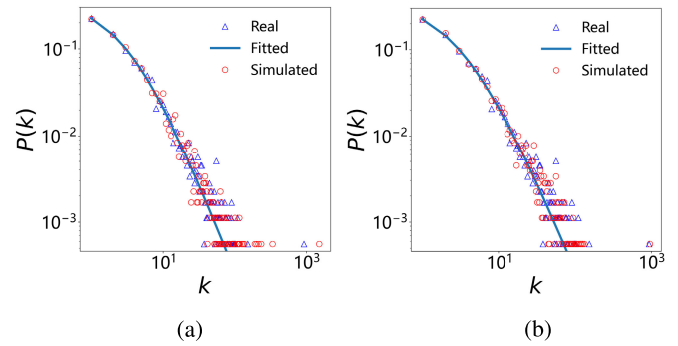


Fig. 10. Fitting results of (a) continuous and (b) discrete time network model to a FBN. In these subplots, the blue triangles represent data from the real networks, the blue lines represent the fitting curves, and the red circle denotes the data obtained from the simulation network.

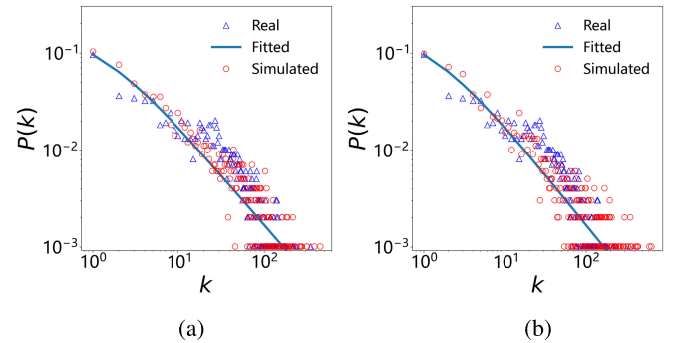


Fig. 11. Fitting results of a (a) continuous and (b) discrete time network model to an EN. In these subplots, the blue triangles represent data from the real networks, the blue lines represent the fitting curves, and the red circle denotes the data obtained from the simulation network.

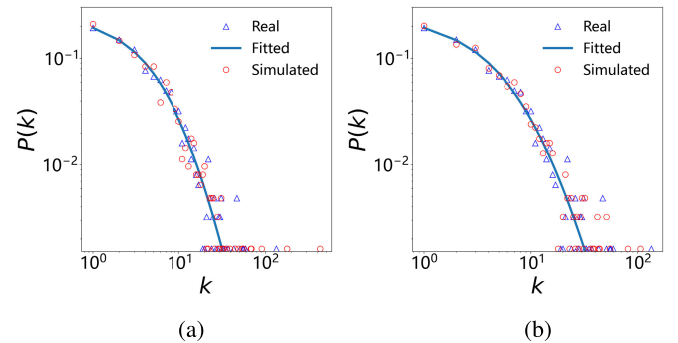


Fig. 12. Fitting results of (a) continuous and (b) discrete time network model to a FPN. In these subplots, the blue triangles represent data from the real networks, the blue lines represent the fitting curves, and the red circle denotes the data obtained from the simulation network.

986 nodes and 24 929 edges in the static graph. The FPN is about mutually liked Facebook pages, nodes represent the pages and edges are mutual likes among them, it has 620 nodes and 2102 edges.

Initially, the degree distribution of the networks was fit using (8) and (20) to determine the values of a and c . Subsequently, these parameters were used to evolve the network model. Figs. 10–12 show that in the continuous-time network model and discrete network model, the degree distributions of the simulated networks closely match those of the real networks. Table III presents a comparison of the KL, JS, P_{LCC} , average shortest path $\langle l \rangle$, and average clustering coefficient

TABLE III
RESULTS AND PARAMETERS OF DELIBERATE ATTACK ON THE NETWORKS

	FBN			EN			FPN		
	real	continuous	discrete	real	continuous	discrete	real	continuous	discrete
a		2.119			1.040			4.586	
c		3.382			1.069			15.672	
KL	-	0.027	0.015	-	0.076	0.080	-	0.040	0.032
JS	-	0.008	0.004	-	0.027	0.032	-	0.010	0.005
P_{LCC}	0.994	0.995	0.994	1.000	1.000	1.000	1.000	1.000	0.995
$\langle l \rangle$	2.910	2.267	2.874	2.586	2.412	2.197	5.008	2.603	3.971
C	0.262	0.344	0.195	0.407	0.290	0.543	0.330	0.288	0.252

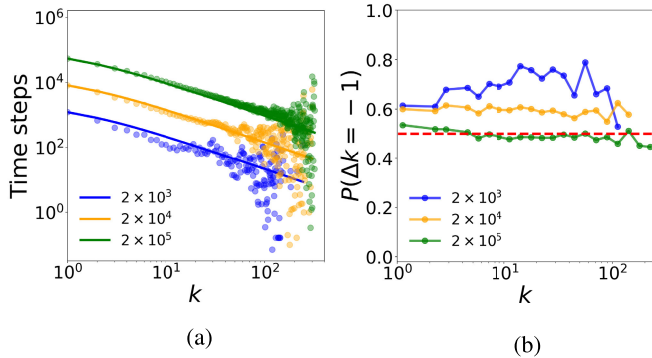


Fig. 13. Average residence time and degree transition patterns of nodes in the EN. (a) Average residence time of nodes at different degree values, where the scatter points represent the empirical results from the real network and the solid lines denote the simulated results. (b) Probability that nodes with degree k decrease to $k - 1$ during a degree transition. Degree values are grouped logarithmically from $10^{0.1x}$ to $10^{0.1(x+1)}$ to reduce fluctuations for large degrees. In both panels, the blue, yellow, and green curves correspond to edge lifetimes of 2×10^3 , 2×10^4 , and 2×10^5 , respectively. The red dashed line in (b) indicates the reference level $P(\Delta k = -1) = 0.5$.

C between the real and simulated networks. The model aligns well with the connectivity characteristics of all three real-world networks, with the average shortest path length l fitting well for the FBN and EN, though it is smaller for the FPN. The FPN network has an average shortest path length of 5.008, while the continuous and discrete-time models exhibit shorter path lengths of 2.603 and 3.971, respectively. It is also noticeable that, while the model closely fits $\langle l \rangle$ for the FBN and EN networks, the evolved networks tend to have generally smaller path lengths.

In terms of clustering coefficient C , the model's fit is less accurate compared to P_{LCC} and $\langle l \rangle$, with larger discrepancies between the model and real networks. For degree distribution, the model fits the real data quite well, with both discrete and continuous-time models showing KL and JS divergences of less than 0.05 for FBN and FPN. However, for EN, the KL and JS divergences are higher, reaching 0.08 and 0.032, respectively. Overall, the proposed network model effectively captures the degree distribution and connectivity features of real scale-free networks, provides a reasonable approximation of the average shortest path length with a slight underestimation, but fails to accurately reproduce clustering characteristics. The relatively high clustering coefficients shown in Table III can be attributed to the degree-dependent rewiring mechanism under a fixed-size constraint. Since edge creation and deletion events are driven by nodes' degree-change tendencies, nodes

with higher degrees are more likely to interact and form new edges among themselves or with shared neighbors at a higher frequency. This preferential interaction among active high-degree nodes promotes the repeated formation of triangular connections, leading to an elevated level of clustering compared with purely random rewiring models.

To verify the validity of the proposed assumption regarding degree variation patterns in real temporal networks, Fig. 13 presents the residence time and transition behavior of nodes in EN at different degree values. Since the original dataset only records the time when each edge is added but not when it is removed, we assume the edge lifetime to be 2×10^3 , 2×10^4 , and 2×10^5 . As shown in Fig. 13(a), for all lifetime settings, the residence time exhibits a clear decreasing trend with increasing degree, following an approximately logarithmic decay. This pattern indicates that nodes with higher degrees experience more frequent degree fluctuations. To ensure consistent temporal scaling, the results from the simulated network were rescaled such that the average residence time at $k = 1$ matches that of the empirical network. After aligning the temporal scales, the simulated and empirical results remain consistent across all lifetime settings. These findings confirm that the assumption in our model—that nodes with larger degrees are more dynamically unstable—is well supported by the empirical data from EN. Fig. 13(b) characterizes the directionality of degree transitions by measuring the probability $P(\Delta k = -1)$ that a node's degree decreases by 1 when a transition occurs. To reduce fluctuations caused by sparse samples at large degrees, degree values are logarithmically binned into intervals of $10^{0.1x} - 10^{0.1(x+1)}$, and the average probability within each bin is plotted as a point. As shown in Fig. 13(b), when the edge lifetime is relatively short, node degrees tend to decrease across all degree ranges, and the current degree value has little influence on the direction of change. When the edge lifetime increases to 2×10^5 time steps, the probability $P(\Delta k = -1)$ remains consistently close to 0.5 across all degree values. In this case, the observed dynamics are consistent with the modeling assumption in our framework, where nodes are allowed to increase or decrease their degree with equal probability during degree transitions.

IV. CONCLUSION

This work introduces a dynamic network model leveraging random walk processes with variable drift coefficients. Stochastic process theory is applied to analyze the stationary degree distribution as the network evolves to a steady state. Depending on the specific setting of the drift coefficients,

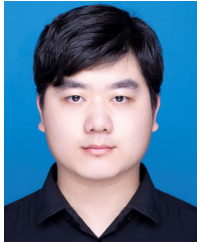
which is a monotonically increasing function of degree, the network's stationary degree distribution can exhibit a power-law distribution. This enables the modeling of dynamically evolving networks with fixed sizes and power-law degree distributions in both discrete and continuous time. Based on numerical simulations and evaluations using the KL and JS divergences, we find good agreement between the theoretical degree distribution predicted by the model and the simulation results. We also examine the impact of the power-law exponent and attack ratio on the largest connected component of the network and show that, similar to other SF network models, our network model is vulnerable to deliberate attacks but robust against random attacks. In the case of deliberate attacks, we observe the resilience of the largest connected component and the shortest path length. The results indicate that network efficiency takes longer to recover than connectivity. Additionally, the network generated by the discrete-time model exhibits relatively higher robustness. This increased resilience suggests that, under the same attack conditions, the discrete-time model can maintain better connectivity and resilience compared to the continuous-time model. Furthermore, we employ the proposed model to simulate three real-world networks and compared the resulting KL and JS divergences, the size of the largest connected component, the average shortest path length, and the clustering coefficient. Our model fits well with the degree distribution and connectivity of real networks, though it slightly underestimates the average shortest path length and shows suboptimal performance in clustering coefficient fitting. We further validate the rationality of the model's core assumption using the empirical temporal EN. By comparing the degree-dependent residence times and transition probabilities between the real and simulated networks under different edge lifetimes, we observe strong consistency after temporal scale normalization. The results confirm that the modeled mechanism accurately captures the degree fluctuation patterns observed in specific real temporal systems. Such validation is essential for assessing the practical effectiveness and applicability of the proposed model, demonstrating that it can reproduce not only the structural but also the dynamical properties of real-world evolving networks.

The significance of this model lies in its departure from the assumptions of preferential attachment and network growth that underlie conventional SF network models, allowing the generation and simulation of dynamically evolving networks with fixed sizes and controllable power-law exponents in their degree distributions. Additionally, due to the dynamic nature and convergence properties of the network model presented in this work, it can be used to describe and simulate the resilience of specific types of networks after being attacked. This provides a framework for studying the collapse and resilience of network topologies.

REFERENCES

- [1] A.-L. Barabási and Z. N. Oltvai, "Network biology: Understanding the cell's functional organization," *Nature Rev. Genet.*, vol. 5, no. 2, pp. 101–113, Feb. 2004.
- [2] J.-W. Wang and L.-L. Rong, "Cascade-based attack vulnerability on the U.S. power grid," *Saf. Sci.*, vol. 47, no. 10, pp. 1332–1336, Dec. 2009.
- [3] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-world networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [4] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct. 1999.
- [5] R. Albert and A.-L. Barabási, "Topology of evolving networks: Local events and universality," *Phys. Rev. Lett.*, vol. 85, no. 24, pp. 5234–5237, Dec. 2000.
- [6] G. Bianconi and A.-L. Barabási, "Bose-einstein condensation in complex networks," *Phys. Rev. Lett.*, vol. 86, no. 24, pp. 5632–5635, Jun. 2001.
- [7] X. Li and G. Chen, "A local-world evolving network model," *Phys. A, Stat. Mech. Appl.*, vol. 328, nos. 1–2, pp. 274–286, Oct. 2003.
- [8] C. Li and G. Chen, "Modelling of weighted evolving networks with community structures," *Phys. A, Stat. Mech. Appl.*, vol. 370, no. 2, pp. 869–876, Oct. 2006.
- [9] Y. Gu and J. Sun, "A local-world node deleting evolving network model," *Phys. Lett. A*, vol. 372, no. 25, pp. 4564–4568, Jun. 2008.
- [10] A. Li, S. P. Cornelius, Y.-Y. Liu, L. Wang, and A.-L. Barabási, "The fundamental advantages of temporal networks," *Science*, vol. 358, no. 6366, pp. 1042–1046, Nov. 2017.
- [11] M. Gerlach and E. G. Altmann, "Testing statistical laws in complex systems," *Phys. Rev. Lett.*, vol. 122, no. 16, Apr. 2019, Art. no. 168301.
- [12] P. Dagum, A. Galper, and E. Horvitz, "Dynamic network models for forecasting," in *Uncertainty in Artificial Intelligence*. Amsterdam, The Netherlands: Elsevier, 1992, pp. 41–48.
- [13] P. Holme and J. Saramäki, "Temporal networks," *Phys. Rep.*, vol. 519, no. 3, pp. 97–125, 2012.
- [14] O. Sporns, D. Chialvo, M. Kaiser, and C. Hilgetag, "Organization, development and function of complex brain networks," *Trends Cognit. Sci.*, vol. 8, no. 9, pp. 418–425, Sep. 2004.
- [15] B. A. Huberman and L. A. Adamic, "Growth dynamics of the world-wide web," *Nature*, vol. 401, no. 6749, p. 131, Sep. 1999.
- [16] E. Bullmore and O. Sporns, "Complex brain networks: Graph theoretical analysis of structural and functional systems," *Nature Rev. Neurosci.*, vol. 10, no. 3, pp. 186–198, Mar. 2009.
- [17] M. Medo, G. Cimini, and S. Gualdi, "Temporal effects in the growth of networks," *Phys. Rev. Lett.*, vol. 107, no. 23, Dec. 2011, Art. no. 238701.
- [18] B. Pi, L.-J. Deng, M. Feng, M. Perc, and J. Kurths, "Dynamic evolution of complex networks: A reinforcement learning approach applying evolutionary games to community structure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 10, pp. 8563–8582, Oct. 2025.
- [19] X. Zhang, Z. He, and L. Rayman-Bacchus, "Random birth-and-death networks," *J. Stat. Phys.*, vol. 162, no. 4, pp. 842–854, Feb. 2016.
- [20] M. Feng, L. Deng, and J. Kurths, "Evolving networks based on birth and death process regarding the scale stationarity," *Chaos, Interdiscipl. J. Nonlinear Sci.*, vol. 28, no. 8, Aug. 2018, Art. no. 083118.
- [21] M. Feng, Y. Li, F. Chen, and J. Kurths, "Heritable deleting strategies for birth and death evolving networks from a queueing system perspective," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 10, pp. 6662–6673, Oct. 2022.
- [22] Y. Li, M. Feng, and J. Kurths, "Evolving network modeling driven by the degree increase and decrease mechanism," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 53, no. 9, pp. 5369–5380, Sep. 2023.
- [23] Z. Zeng, M. Feng, and J. Kurths, "Temporal network modeling with online and hidden vertices based on the birth and death process," *Appl. Math. Model.*, vol. 122, pp. 151–166, Oct. 2023.
- [24] P. Holme, "Modern temporal network theory: A colloquium," *Eur. Phys. J. B*, vol. 88, no. 9, pp. 1–30, Sep. 2015.
- [25] R. L. G. Raimundo, P. R. Guimarães, and D. M. Evans, "Adaptive networks for restoration ecology," *Trends Ecol. Evol.*, vol. 33, no. 9, pp. 664–675, Sep. 2018.
- [26] G. Petri and A. Barrat, "Simplicial activity driven model," *Phys. Rev. Lett.*, vol. 121, no. 22, Nov. 2018, Art. no. 228301.
- [27] M. Serafino et al., "True scale-free networks hidden by finite size effects," *Proc. Nat. Acad. Sci. USA*, vol. 118, no. 2, Jan. 2021, Art. no. 2013825118.
- [28] S. Katti and A. V. Rao, "Handbook of the poisson distribution," *Technometrics*, vol. 10, no. 2, p. 412, May 1968.
- [29] B. Zhou, X. Lu, and P. Holme, "Universal evolution patterns of degree assortativity in social networks," *Social Netw.*, vol. 63, pp. 47–55, Oct. 2020.
- [30] M. Feng, Z. Zeng, Q. Li, M. Perc, and J. Kurths, "Information dynamics in evolving networks based on the birth-death process: Random drift and natural selection perspective," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 54, no. 8, pp. 5123–5136, Aug. 2024.

- [31] R. A. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. AAAI*, vol. 29, 2015, pp. 4292–4293. [Online]. Available: <https://networkrepository.com>
- [32] A. Paranjape, A. R. Benson, and J. Leskovec, "Motifs in temporal networks," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, Feb. 2017, pp. 601–610.



Yichao Yao (Student Member, IEEE) received the bachelor's degree in data science and big data technology from the College of Artificial Intelligence, Southwest University, Chongqing, China, in 2024, where he is currently pursuing the master's degree in computer science.

His research interests include complex networks, evolutionary games, stochastic processes, and non-linear science.



Minyu Feng (Senior Member, IEEE) received the dual Ph.D. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, and Humboldt University of Berlin, Berlin, Germany, in 2018.

Since 2019, he has been an Associate Professor at the College of Artificial Intelligence, Southwest University, Chongqing, China. He has published more than 70 peer-reviewed papers in authoritative journals, such as *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*,

IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, and *IEEE TRANSACTIONS ON CYBERNETICS*. His research interests include complex systems, evolutionary game theory, computational social science, and mathematical epidemiology.

Dr. Feng is a Senior Member of China Computer Federation (CCF) and Chinese Association of Automation (CAA). Currently, he serves as a Subject Editor for *Applied Mathematical Modelling*, an Editorial Advisory Board Member for *Chaos*, and an Editorial Board Member for *Humanities and Social Sciences Communications*, *Scientific Reports*, and the *International Journal of Mathematics for Industry*. Besides, he is a reviewer for *Mathematical Reviews* of the American Mathematical Society.



Matjaž Perc (Member, IEEE) received the Ph.D. degree in physics from the University of Maribor, Maribor, Slovenia, in 2006.

He is currently Professor of physics at the University of Maribor, a Staff Researcher at the Community Healthcare Center Dr. Adolf Drolc Maribor, Maribor, and an Adjunct Professor at Kyung Hee University and Korea University, Seoul, Republic of Korea.

Dr. Perc is a member of Academia Europaea and European Academy of Sciences and Arts, and among top 1% most cited physicists according to 2020, 2021, 2022, 2023, and 2024 Clarivate Analytics data. He was also a recipient of the Young Scientist Award for Socio and Econophysics from the German Physical Society in 2015, and the 2017 USERN Laureate. In 2018, he received the Zois Award, which is the highest national research award in Slovenia. In 2019, he became a Fellow of the American Physical Society. Since 2021 he has also been the Vice-Dean of Natural Sciences at the European Academy of Sciences and Arts.



Jürgen Kurths received the B.S. degree in mathematics from the University of Rostock, Rostock, Germany, the Ph.D. degree from the Academy of Sciences, German Democratic Republic, Berlin, Germany, in 1983, the Honorary degree from N. I. Lobachevsky State University, Nizhny Novgorod, Russia, in 2008, and the Honorary degree from Saratov State University, Saratov, Russia, in 2012.

From 1994 to 2008, he was a Full Professor with the University of Potsdam, Potsdam, Germany.

Since 2008, he has been a Professor of nonlinear dynamics with Humboldt University, Berlin, and the Chair of the Research Domain Complexity Science, Potsdam Institute for Climate Impact Research, Potsdam. He has authored more than 700 papers, which are cited more than 53 000 times (*H*-index: 106). His main research interests include synchronization, complex networks, time-series analysis, and their applications.

Dr. Kurths is a fellow of the American Physical Society, the Royal Society of Edinburgh, and the Network Science Society. He is a member of the Academia Europaea. He was a recipient of the Alexander von Humboldt Research Award from India, in 2005, and from Poland in 2021, the Richardson Medal of the European Geophysical Union in 2013, and the Eight Honorary Doctorates. He is a Highly Cited Researcher in Engineering. He is the Editor-in-Chief of *Chaos* and on the editorial boards of more than ten journals.