Deterministic ripple-spreading model for complex networks

Xiao-Bing Hu,[1,2] Ming Wang,[1] Mark S. Leeson,[2] Evor L. Hines,[2] and Ezequiel Di Paolo[3,4]
[1]*State Key Laboratory of Earth Surface Processes and Resource Ecology, Beijing Normal University, China*
[2]*School of Engineering, University of Warwick, Coventry, United Kingdom*
[3]*Ikerbasque, Basque Science Foundation, Centre for Research on Life, Mind and Society, University of the Basque Country, San Sebastian, Spain*
[4]*The Centre for Computational Neuroscience and Robotics, Department of Informatics, University of Sussex, Brighton, United Kingdom*
(Received 19 July 2010; revised manuscript received 18 January 2011; published 25 April 2011)

This paper proposes a deterministic complex network model, which is inspired by the natural ripple-spreading phenomenon. The motivations and main advantages of the model are the following: (i) The establishment of many real-world networks is a dynamic process, where it is often observed that the influence of a few local events spreads out through nodes, and then largely determines the final network topology. Obviously, this dynamic process involves many spatial and temporal factors. By simulating the natural ripple-spreading process, this paper reports a very natural way to set up a spatial and temporal model for such complex networks. (ii) Existing relevant network models are all stochastic models, i.e., with a given input, they cannot output a unique topology. Differently, the proposed ripple-spreading model can uniquely determine the final network topology, and at the same time, the stochastic feature of complex networks is captured by randomly initializing ripple-spreading related parameters. (iii) The proposed model can use an easily manageable number of ripple-spreading related parameters to precisely describe a network topology, which is more memory efficient when compared with traditional adjacency matrix or similar memory-expensive data structures. (iv) The ripple-spreading model has a very good potential for both extensions and applications.

DOI: 10.1103/PhysRevE.83.046123 PACS number(s): 89.75.Hc, 05.10.−a

I. INTRODUCTION

Complex networks, i.e., networks whose structure is irregular and dynamically evolving in time, describe a wide range of systems in nature and society, and are all around us in our daily life [1–6]. In the past few decades many efforts have been made to model and analyze various complex networks [1,2]. Most network models can be classified as stochastic, because they have a typical feature in common: A stochastic model abstracts one or a few network properties or parameters in order to capture in quantitative terms the underlying organizing principles of complex networks, and these network properties or parameters can estimate what the network looks like, but cannot guarantee an exact or unique topology. In other words, different network topologies may have exactly the same values for the specified properties or parameters. For instance, in the classic random graph theory [7], the connection probability is the core parameter of the network model. In the generalized random graph model [8], the degree distribution following a power law is used as the input in order to be able to describe the scale-free character of real networks that the classic random graph model cannot capture. In the theory of evolving networks [9], the parameter, so-called preferential attachment, is often used to model scale-free real networks. In a recently reported spatially embedded random network model [10], the connection probability is formulated as a function of distance between nodes. In either of the above network models, even if the input is fixed, e.g., the connection probability, power law for degree distribution, or preferential attachment is fixed, the output of the model is enormous, and what the output topology exactly looks like is largely determined by chance. Therefore, they are all stochastic models. Except those network properties which explicitly depend on the input parameters, the output topology of the models is largely unpredictable or uncertain in terms of other network properties. As a result, it is difficult to apply these models in network design, such as the topology optimization problem.

In this paper we will propose a deterministic complex network model inspired by the natural ripple-spreading phenomenon. The model is hereafter called the ripple-spreading network model (RSNM). In this model, some ripple-spreading related parameters (RSRPs) are defined as input. Unlike stochastic models, once the values of input are fixed, the output network topology will also be fixed and unique in the model. Therefore, we can adjust these RSRPs in order to improve the network topology in terms of concerned network properties. Furthermore, there is great freedom and flexibility to modify and extend the RSNM. For instance, the proposed RSNM can easily be extended to a semideterministic version and a stochastic version. Another big advantage of the proposed RSNM is, compared with conventionally adjacent matrix or other memory-expensive data structures, the RSNM can actually use a handful of parameters to record the topology of a complex network, no matter the network scale. The memory efficiency of RSNM makes it very friendly to population-based optimization algorithms in the optimization of large-scale network topology.

The remainder of this paper is organized as follows. Section. II explains the details of the proposed RSNM. Section. III analyzes some important properties of the model. Section. IV discusses some potential modifications and extensions of the proposed model, and Sec. V gives some simulation results. The paper ends with some conclusions and discussions on future work in Sec. VI.

 ©2011 American Physical Society

II. BASIC RIPPLE-SPREADING NETWORK MODEL (RSNM)

A. Basic idea of RSNM

The basic natural ripple-spreading phenomenon is as follows. Suppose a bunch of stakes are randomly distributed in a quiet pond. Then suddenly a stone is thrown into the pond, and an initial ripple is generated from the point where the stone hits the quiet water. When the ripple reaches a near stake, a new ripple is generated around the stake due to the reflection effect. Hereafter, for the sake of consistency, we call such a new ripple as a responding ripple, and the ripple which triggers the responding ripple as a stimulating ripple. A responding ripple may trigger new responding ripples around other stakes. As the initial stimulating ripple spreads out, more and more responding ripples are stimulated around stakes. However, since the point energy on the stimulating ripples decays as the ripples spread out, those responding ripples triggered at a late phase could hardly be noticed.

Now, we replace the stakes with a set of wireless vibration sensors. Each sensor can detect and measure the point energy when a stimulating ripple reaches the sensor. If the point energy is above a preset threshold, then the sensor will generate a responding ripple, whose initial energy is a function of the point energy of the stimulating ripple, e.g., multiplying the point energy by an amplifying factor. If the stimulating ripple comes from another sensor, then the current sensor will compare the point energy against another preset threshold, once above which a permanent communication will be established between the two sensors. Obviously, the amplifying function of a sensor makes it possible for the information associated with the stone hitting the quiet pond to propagate through the whole wireless sensor network, even though the point energy of the initial stimulating ripple may decrease quickly below the threshold. Suppose each sensor can generate no more than one responding ripple, in other words, each sensor can be activated no more than once. Then, after all ripples decay, we will get a network according to those permanent communications, i.e. links, established between wireless sensors, i.e., nodes. Clearly there are some factors affecting the final network topology: for instance, how many stones hit the pond to generate initial stimulating ripples, where do they hit the pond, what is the mass of each stone (will determine the initial energy of the associated stimulating ripple), what are the preset values for the thresholds, and what is the amplifying factor of each sensor? By mathematically formulating these factors and the relationships between them, we can get a deterministic model for complex networks. The mechanism of this ripple-spreading model is intuitively illustrated in Fig. 1.

B. Ripple-spreading related parameters (RSRPs)

Here we give the mathematical descriptions of those factors discussed above, which hereafter we call ripple-spreading related parameters (RSRPs).

The first group of RSRPs are associated with the epicenters of initial stimulating ripples (EISRs). Suppose N_{EISR} stones of different mass hit the pond in different points at different time instants, i.e., there are N_{EISR} EISRs, EISR i, $i = 1, \ldots, N_{\mathrm{EISR}}$, has an initial point energy of $\mathrm{E}_{\mathrm{EISR}}(i)$, its coordinates are $[x_{\mathrm{EISR}}(i), y_{\mathrm{EISR}}(i)]$, and it is not active until time instant $T_{\mathrm{EISR}}(i)$.

The second group of parameters is related to the nodes which are to be connected in order to generate a network. For the sake of simplicity but without losing generality, we suppose all nodes are distributed in a limited two-dimentional space in this paper. It is also assumed that the total number of nodes, N_{N}, and their locations, $[x_{\mathrm{N}}(i), y_{\mathrm{N}}(i)]$, are already given and fixed. To get different topologies from this fixed set of nodes, we introduce three RSRPs to each node, $\alpha(i)$, $\beta_{\mathrm{R}}(i)$, and $\beta_{\mathrm{L}}(i)$, which are the amplifying factor, the threshold to generate a responding ripple, and the threshold to establish a link, for node i, $i = 1, \ldots, N_{\mathrm{N}}$, respectively.

With the above RSRPs, the proposed complex network modeling process can be mathematically described as follows.

Step 1. Initialize the current time instant, i.e., $t = 0$. Initialize the current point energy of each EISR as

$$e_{\mathrm{EISR}}(i,t) = E_{\mathrm{EISR}}(i), \quad i = 1, \ldots, N_{\mathrm{EISR}}, \tag{1}$$

Since each node has no initial energy, i.e., $E_{\mathrm{N}}(i) = 0$, therefore its current point energy is

$$e_{\mathrm{N}}(i,t) = E_{\mathrm{N}}(i) = 0, \quad i = 1, \ldots, N_{\mathrm{N}}. \tag{2}$$

Assume each EISR or node has a ripple with a current radius of 0, i.e., $r_{\mathrm{EISR}}(i,t) = 0$ or $r_{\mathrm{N}}(i,t) = 0$.

Step 2. If the stopping criteria is not satisfied, do:

Step 2.1. Let $t = t + 1$.

Step 2.2. Check t against $T_{\mathrm{EISR}}(i)$. If $t > T_{\mathrm{EISR}}(i)$, then update the current radius and point energy of EISR i as follows:

$$r_{\mathrm{EISR}}(i,t) = r_{\mathrm{EISR}}(i,t-1) + s, \tag{3}$$

$$e_{\mathrm{EISR}}(i,t) = f_{\mathrm{decay}}[E_{\mathrm{EISR}}(i), r_{\mathrm{EISR}}(i,t), t], \tag{4}$$

where s is the spreading speed of ripples, i.e., the change in the radius of a ripple during one time instant, and f_{decay} is a function defining how the point energy decays as the ripple spreads out. A typical decaying function may be

$$f_{\mathrm{decay}}[E_{\mathrm{EISR}}(i), r_{\mathrm{EISR}}(i,t), t] = \eta \frac{E_{\mathrm{EISR}}(i)}{2\pi r_{\mathrm{EISR}}(i,t)}, \tag{5}$$

where η is a coefficient and π is the mathematical constant. Clearly η has an important influence on the decaying speed of ripples, and will therefore affect the final network topology.

Step 2.3. Check which new nodes are reached by the ripples of EISRs. Suppose $D_{\mathrm{EISR}}(i,j)$ is the distance between EISR i and node j. If $E_{\mathrm{N}}(j) = 0$ and $D_{\mathrm{EISR}}(i,j) \leqslant r_{\mathrm{EISR}}(i,t)$, then node j is reached by the ripple associated with EISR i. If $e_{\mathrm{EISR}}(i,t) \geqslant \beta_{\mathrm{R}}(j)$, then node j is activated by EISR i, and generates a responding ripple with

$$E_{\mathrm{N}}(j) = \alpha(j) e_{\mathrm{EISR}}(i,t), \tag{6}$$

and $e_{\mathrm{N}}(j,t) = E_{\mathrm{N}}(j)$.

Step 2.4. If $e_{\mathrm{N}}(i,t-1) > 0$, $i = 1, \ldots, N_{\mathrm{N}}$, then update the current radius and point energy of the ripple starting from node i in a similar way to EISRs, i.e.,

$$r_{\mathrm{N}}(i,t) = r_{\mathrm{N}}(i,t-1) + s, \tag{7}$$

$$e_{\mathrm{N}}(i,t) = f_{\mathrm{decay}}[E_{\mathrm{N}}(i), r_{\mathrm{N}}(i,t), t]. \tag{8}$$

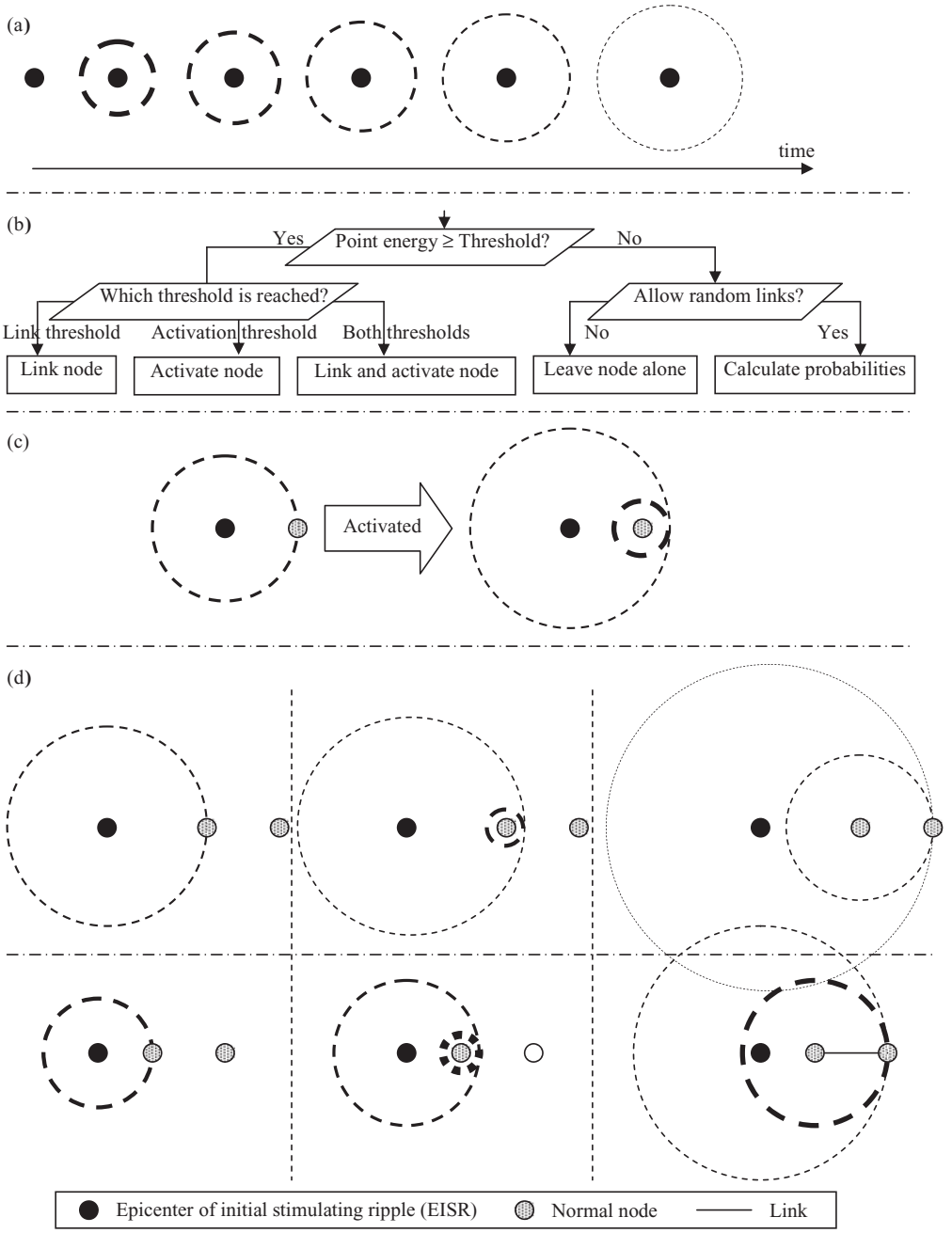

FIG. 1. Mechanism of using ripple-spreading method to model networks: (a) The point energy of a stimulating ripple decays as it spreads (the strength of point energy is reflected as the thickness of the ripple). (b) At least one threshold needs to be specified, against which the point energy of a stimulating ripple is compared to determine whether or not a node should be linked and/or activated, or to calculate the probability of a node being linked and/or activated. (c) A point-energy amplifying factor needs to be set up, so that when a node is activated by a stimulating ripple, the starting point energy of the responding ripple is determined by multiplying the point energy of the stimulating ripple by the factor. (d) Due to the decaying point energy, the threshold, and the amplifying factor, the location of an EISR will affect the network topology (in the first case, no link, while in the second case, linked).

Basically, Eqs. (3)–(8) show that, no matter where the ripples originate, they should have the same spreading speed and the same decaying function, just as in the nature world. However, this may be amendable in order to get a more complicated artificial model.

Step 2.5. Check which new nodes are reached by the ripples of other nodes. Suppose $D_{\mathrm{N}}(i,j)$ is the distance between node i and node j. If $E_{\mathrm{N}}(j) = 0$ and $D_{\mathrm{N}}(i,j) \leqslant r_{\mathrm{N}}(i,t)$, then node j is reached by the ripple generated by node i. If $e_{\mathrm{N}}(i,t) \geqslant \beta_{\mathrm{R}}(j)$, then node j is activated by node i, and generates a responding ripple with

$$E_{\mathrm{N}}(j) = \alpha(j) e_{\mathrm{N}}(i,t), \tag{9}$$

and $e_{\mathrm{N}}(j,t) = E_{\mathrm{N}}(j)$. If $e_{\mathrm{N}}(i,t) \geqslant \beta_{\mathrm{L}}(j)$, then a connection between node i and node j is established, i.e.,

$$A(i,j) = A(j,i) = 1, \tag{10}$$

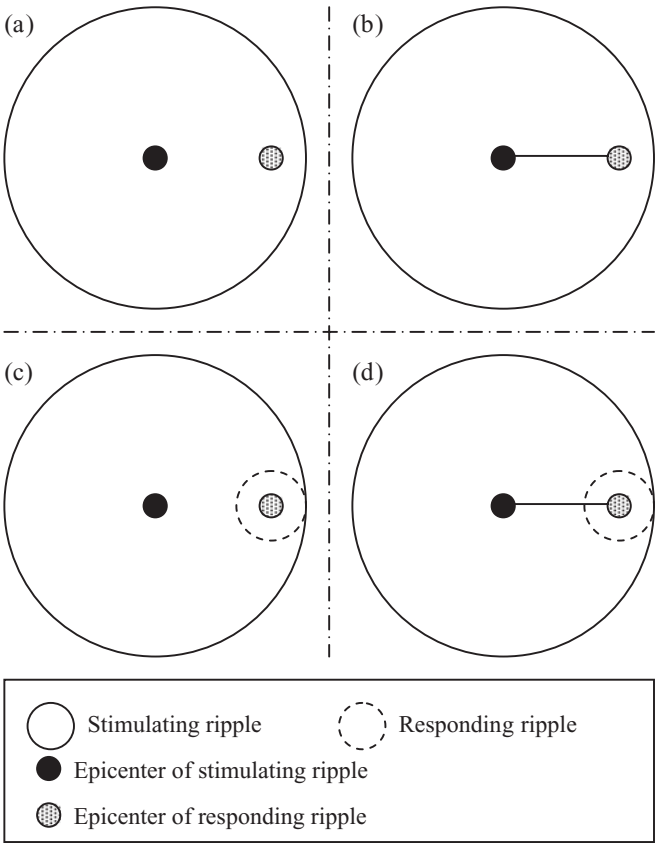

FIG. 2. Two behaviors and four combinations: (a) Not linked and not activated; (b) linked but not activated; (c) not linked but activated; (d) linked and activated.

where A is the adjacency matrix which records the network topology.

Different stopping criteria may be used in step 2. For instance, the current time instance is beyond a specific time window, no EISR or node has a current point energy above any threshold, or the upper bound for the number of total links is reached.

From the above network modeling process, one can see there are two basic behaviors for each node: being activated and being connected. The evolving of a complex network is based on these two behaviors, which are mainly determined by two thresholds, $\beta_{\mathrm{R}}(i)$ and $\beta_{\mathrm{L}}(i)$. These two behaviors may have four combinations, as illustrated in Fig. 2, which contribute to the complex evolving behavior of a random network.

Also from the modeling process proposed above, one can see that the network topology is largely determined by the RSRPs. Actually, every given set of values for the RSRPs will result in a unique topology, and given different values for the parameters, different topologies will be generated. Figure 3 gives four simple examples about how the values of the RSRPs affect the output topologies. In the four networks in Fig. 3, all other RSRPs have the same values, but the locations of four EISRs are different. As a result, there are four different topologies. It should be pointed out that the feature that every given set of values for the model parameters will lead to a unique topology is distinct from other models of random networks, where the model parameters (e.g., the connection probability in the classic random graph theory and the preferential attachment in the theory of evolving networks) cannot uniquely determine network topology. Because of this deterministic feature, we call the proposed model a deterministic model of random networks. Despite the deterministic feature of the proposed model, random network properties may be captured by the random setup of the RSRPs, as will be analyzed in the next section.

With these RSRPs, the proposed RSNM can easily and naturally capture and/or reflect spatial and temporal features in many real-world complex networks. For instance, the impact of earthquakes through a risk chain and the breakout of plagues in a community can be described as evolving networks triggered by a few initial simulating ripples; the threshold of nodes is related to the vulnerability of infrastructure and the immunity of individuals, and the amplifying factor of nodes to connectivity of infrastructure and the social activeness of individuals.

III. FURTHER ANALYSIS ON PROPERTIES OF RSNM

In this section, we aim to analyze some properties of the proposed RSNM. In particular, we will give some conditions under which the proposed model will generate a network topology with certain properties.

For a network with N_{N} nodes, the number of total potential edges between nodes is

$$N_{\mathrm{PE}} = N_{\mathrm{N}}(N_{\mathrm{N}} - 1)/2. \tag{11}$$

For each pair of nodes (i, j), let $D_{\mathrm{N}}(i, j)$ denote the length of the potential edge between node i and node j, i.e., the direct distance between node i and node j, regardless of whether there is an actual connection between the pair of nodes. Since all nodes are distributed in a limited space, we can assume L_{U} and L_{L} are the maximal and minimal length in all potential edges, respectively. For the sake of simplicity, unless specified otherwise, we assume all ripples have the same spreading speed s and the same point-energy decaying function f_{decay} as given by Eq. (5), and all nodes have the same activating threshold β_{R}, the same connecting threshold β_{L}, and the same amplifying factor α.

A. Completely connected graph

We have the following conditions under which the proposed RSNM will generate a completely connected graph, where all potential edges between nodes are established.

Condition 1. If the inequality

$$\eta \frac{\alpha \beta_{\mathrm{R}}}{2\pi L_{\mathrm{U}}} \geqslant \max(\beta_{\mathrm{R}}, \beta_{\mathrm{L}}) \tag{12}$$

holds, then once a node is activated by any initial stimulating ripple, the RSNM will generate a completely connected graph. For proof, see Appendix A.

Clearly, if $\beta_{\mathrm{R}} \geqslant \beta_{\mathrm{L}}$, then Condition 1 can be simplified as follows.

Condition 2. If the inequality

$$\eta\alpha \geqslant 2\pi L_{\mathrm{U}} \tag{13}$$

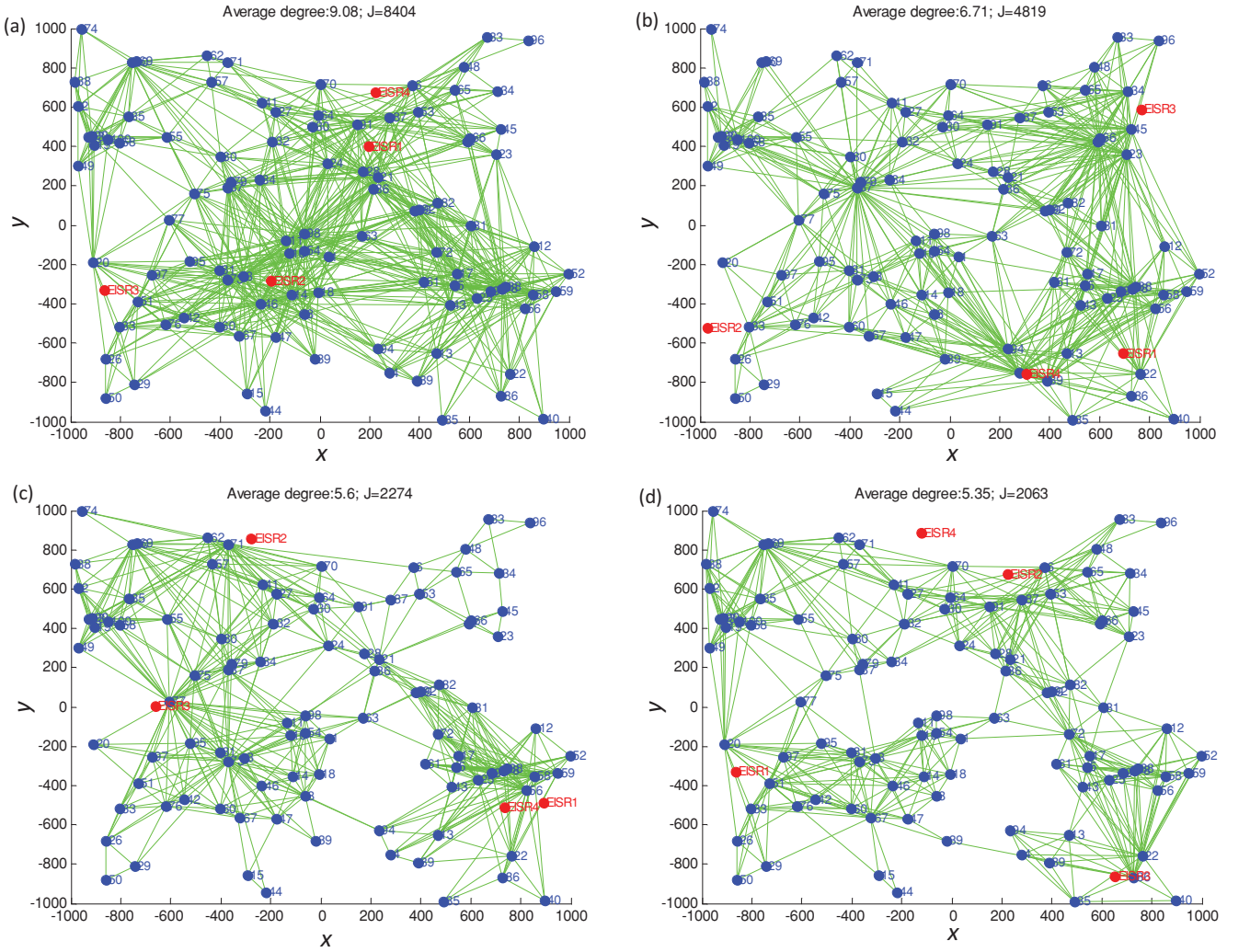

FIG. 3. (Color) An illustration regarding how the distribution of EISRs affects network topology: $E_{\mathrm{EISR}} = 18850.0$, $\beta_{\mathrm{R}} = \beta_{\mathrm{L}} = 5.0$, $\alpha = 1036.7$, and the coordinates (x,y) of four EISRs are (194.6, 397.9), (−194.9, −281.1), (−865.0, −329.3), and (222.4, 672.6) in (a), (692.6, −652.8), (−969.9, −521.3), (766.0, 588.9), and (304.2, −755.3) in (b), (891.0, −488.7), (−276.9, 853.9), (−660.1, 4.9), and (736.3, 508.) in (c), and (−865.0, −329.3), (222.4, 672.6), (653.8, −863.3), and (−123.3, 886.3) in (d), respectively. J is an objective function defined as the sum of the power of the node degree deviation from the desirable average degree.

holds, then once a node is activated by any initial stimulating ripple, the RSNM will generate a completely connected graph.

Since the distribution of nodes is fixed, L_{U} is then fixed. From the above two conditions, one can see that the RSRPs $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}$, α, and η play a crucial role in generating a completely connected graph.

It should be noted that even if the above conditions are not satisfied, the RSNM could still generate a completely connected graph, as long as the initial stimulating ripples have a huge enough initial energy, such as satisfying the following condition.

Condition 3. For an initial stimulating ripple i, $i = 1, \ldots, N_{\mathrm{EISR}}$, if

$$\frac{E_{\mathrm{EISR}}(i)}{\max_{j=1,\ldots,N_{\mathrm{N}}} D_{\mathrm{EISR}}(i,j)} \geqslant \max\left(\frac{2\pi\beta_{\mathrm{R}}}{\eta}, \frac{4\pi^2 L_{\mathrm{U}}\beta_{\mathrm{L}}}{\alpha\eta^2}\right), \tag{14}$$

then the RSNM will generate a completely connected graph. For proof, see Appendix B.

B. Connecting all nodes

Real-world networks are almost always not completely connected, and usually a small portion of potential edges are actually established. Therefore, compared with how to generate a completely connected graph, people are more interested in how to generate a graph where all nodes are connected.

Before we give some conditions for the proposed RSNM to connect all nodes, we need to introduce a unique definition of the distance between nodes: maximum TSP (traveling salesman problem) edge length, denoted as D_{TSP}. Suppose we can find the shortest open TSP route to connect all nodes, then D_{TSP} is the length of the longest edge in this shortest open TSP route.

Condition 4. If the inequality

$$\eta \frac{\alpha \beta_{\mathrm{R}}}{2\pi D_{\mathrm{TSP}}} \geqslant \max(\beta_{\mathrm{R}}, \beta_{\mathrm{L}}) \tag{15}$$

or

$$\eta\alpha \geqslant 2\pi D_{\mathrm{TSP}} \quad \text{and} \quad \beta_{\mathrm{R}} \geqslant \beta_{\mathrm{L}} \tag{16}$$

holds, then once a node is activated by any initial stimulating ripple, the RSNM will generate a graph where all nodes are connected. For proof, see Appendix C.

Actually, we can replace D_{TSP} with some other appropriately defined distances in order to connect all nodes. For instance, suppose we know every open TSP route. Each open TSP route has its own longest edge. Then we find and use the minimal one in such longest edges, which is obviously no larger than D_{TSP}. This will ease Condition 4. However, it is very difficult, if not impossible, to find the minimal longest edge in all open TSP routes. Actually, it is already very difficult and time consuming to find D_{TSP}, particular in the case of a large N_{N}. Therefore, from a practical point of view, it is necessary to replace D_{TSP} with a distance which is easy to calculate, particularly in the case of large-scale networks. To this end, we can construct an open TSP route efficiently as follows: (i) Start with a randomly chosen node, and connect this node to its nearest node; (ii) connect the newly connected node to its own nearest unconnected node, until all nodes are connected. Then we can use the longest edge in this open TSP route to replace D_{TSP} in Condition 4. So far, Condition 4 can guarantee that the graph generated by the RSNM will include at least one chain connecting all nodes [the definition of the chain is illustrated in Fig. 4(a)]. If we use close TSP routes rather than open TSP routes, then Condition 4 can guarantee that the graph generated by the RSNM will include at least one loop connecting all nodes [the definition of the loop is illustrated in Fig. 4(b)].

The influence of initial stimulating ripples can be partially reflected in the following condition.

Condition 5. For an initial stimulating ripple i, $i = 1, \ldots, N_{\mathrm{EISR}}$, if

$$\frac{E_{\mathrm{EISR}}(i)}{\min_{j=1,\ldots,N_{\mathrm{N}}} D_{\mathrm{EISR}}(i,j)} \geqslant \max\left(\frac{2\pi\beta_{\mathrm{R}}}{\eta}, \frac{4\pi^2 L_{\mathrm{U}}\beta_{\mathrm{L}}}{\alpha\eta^2}\right) \tag{17}$$

holds, then the RSNM will generate a graph where all nodes are connected. For proof, see Appendix D.

By easing Condition 4, we can derive another very useful condition, which cannot guarantee that all nodes will be connected, but can guarantee that all nodes will be activated.

Condition 6. If the inequality

$$\eta\alpha \geqslant 2\pi D_{\mathrm{TSP}} \tag{18}$$

holds, then once a node is activated by any initial stimulating ripple, then every other node will be activated under the RSNM. For proof, see Appendix E.

C. Cluster

A cluster is a subgraph where every node connects to all other nodes within it. Conditions 1 and 2 will be very useful to estimate clusters in the resulting graph. For given N_{N} nodes, once the values of the RSRPs, such as $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}$, α, and η, are set up, then with Conditions 1 and 2 we can estimate whether a certain subset of the N_{N} nodes will form a cluster under the RSNM. First, we need to check what is the distance of the longest potential edge between the subset of nodes. Then, we replace the L_{U} with this distance in Conditions 1 and 2 to see if a cluster will be formed. Usually, less nodes included in the subset means a smaller distance of the longest potential edge in the subset, and once $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}$, α, and η are given and fixed, this implies that Conditions 1 and 2 are more likely to be satisfied, and therefore a cluster is more likely to form. Actually, once $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}$, α, and η are given and fixed, then according to Conditions 1 and 2, we can work out two threshold values. Then for any subset of nodes with its longest potential edge shorter than the threshold values, as long as one node within this subset is activated (such as if Condition 6 is satisfied), we know for sure that this subset of nodes will form a cluster. It should be noted that all the above conditions, i.e., Conditions 1–6, are sufficient conditions. Therefore, even if a subset of nodes has its longest potential edge larger than the threshold values calculated according to Conditions 1 and 2, it does not means this subset of nodes will not form a cluster under the given values for $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}$, α, and η. Actually, many other factors, such as the initial stimulating ripples, also have an important influence on the forming of clusters. So, a cluster may still form even though Conditions 1 and 2 are not satisfied.

D. Degree distribution

The degree distribution of a graph generated by the RSNM is a function of the spatial distribution of N_{N} nodes and the RSRPs. We can roughly estimate the degree distribution according to the following condition.

Proposition 1. Suppose every node is activated under a given set of values for $\beta_{\mathrm{R}}, \beta_{\mathrm{L}}, \alpha$, and η (such as when Condition 6 is satisfied). Let D_{DD} be the value of $(\eta\alpha\beta_{\mathrm{R}})/(2\pi\beta_{\mathrm{L}})$, i.e.,

$$D_{\mathrm{DD}} = \frac{\eta\alpha\beta_{\mathrm{R}}}{2\pi\beta_{\mathrm{L}}}. \tag{19}$$

Then, in the graph generated by the RSNM, the number of nodes whose degree is at least k is no smaller than the number

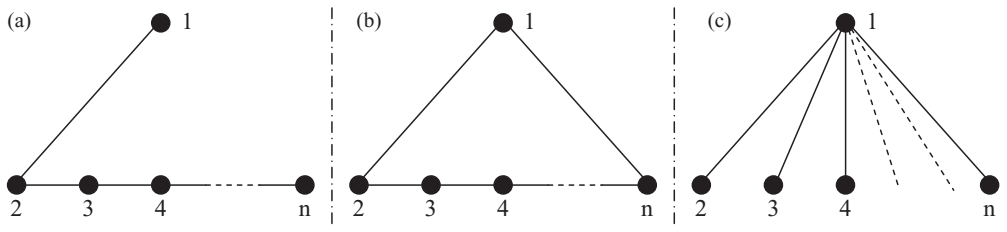

FIG. 4. Network structural motifs corresponding to Conditions 4 and 5: (a) Chain; (b) loop; (c) fan.

of nodes, each of which has k other nodes to which the distance is no larger than D_{DD}.

Degree distribution is an important statistic measure for describing a random network. Obviously, Proposition 1 shows that some spatial factors, such as the distribution of the N_{N} nodes, play an important role in determining the topology of resulting networks. This echoes the fact that many real-world networks are formed at least partially according to some spatial factors. For example, people make friends largely with those who live spatially close to them (although they may move far away from each other after they have become friends). Another example is that things such as the behavior of people, the function of organizations, and the culture of countries are relatively easy to be influenced by those spatially around them. A third example is that the spreading of infective diseases usually has a distinguishable spatial pattern. In these examples, small-world features appear and spatial factors are the main cause of such small-world features. Based on Proposition 1, one can see that the RSNM proposed in this paper can well capture such spatial small-world features in real-world random networks.

E. Scale-free network topology

As is now well known, many real-world complex networks are scale free [1,2]. Simply speaking, a scale-free network has a power-law degree distribution, or put in an intuitive way, most nodes only have one or a few links to other nodes while there are a very few nodes, each of which may connect many other nodes. Here, assuming that the nodes may have different thresholds and amplifying factors, we will give some conditions which may cause the RSNM to generate scale-free topologies.

Proposition 2. Suppose all nodes will be activated, and the following inequalities hold: for node n,

$$\eta\frac{\alpha(i)\beta_{\mathrm{R}}(i)}{2\pi D_{\mathrm{N}}(i,n)} \geqslant \beta_{\mathrm{L}}(n), \quad i=1,\ldots,N_{\mathrm{N}} \quad \text{and} \quad i\neq, \quad \text{or} \quad \eta\frac{\min_{i=1,\ldots,N_{\mathrm{N}}}\alpha(i)\min_{i=1,\ldots,N_{\mathrm{N}}}\beta_{\mathrm{R}}(i)}{2\pi L_{\mathrm{U}}} \geqslant \beta_{\mathrm{L}}(n), \tag{20}$$

$$\eta^{k}\frac{(\max_{i=1,\ldots,N_N}\alpha(i))^{k-1}\max_{i=1,\ldots,N_{\mathrm{EISR}}}E_{\mathrm{EISR}}(i)}{(2\pi)^{k}(L_L)^{k-1}\min_{i=1,\ldots,N_{\mathrm{EISR}};j=1,\ldots,N_{\mathrm{N}}}D_{\mathrm{EISR}}(i,j)} < \min_{i=1,\ldots,N_{\mathrm{N}};i\neq n}\beta_{\mathrm{L}}(i), \quad k=1,\ldots,N_{\mathrm{N}}. \tag{21}$$

Then node n will connect to all other nodes, while there will be no connection between other nodes. In other words, a winner-takes-all network, or a pure fan with node n as the hub, will be generated. For proof, see Appendix F.

Proposition 3. Suppose the following inequalities hold: for node n,

$$\eta\frac{\alpha(n)\beta_{\mathrm{R}}(n)}{2\pi D_{\mathrm{N}}(n,i)} \geqslant \beta_L(i), \quad i=1,\ldots,N_{\mathrm{N}} \quad \text{and} \quad i\neq, \quad \text{or} \quad \eta\frac{\alpha(n)\beta_{\mathrm{R}}(n)}{2\pi L_{\mathrm{U}}} \geqslant \max_{i=1,\ldots,N_{\mathrm{N}}}\beta_{\mathrm{L}}(i), \tag{22}$$

$$\eta\frac{\max_{i=1,\ldots,N_{\mathrm{EISR}}}E_{\mathrm{EISR}}(i)}{2\pi\min_{i=1,\ldots,N_{\mathrm{EISR}};j=1,\ldots,N_{\mathrm{N}}}D_{\mathrm{EISR}}(i,j)} < \min_{i=1,\ldots,N_{\mathrm{N}};i\neq n}\beta_{\mathrm{R}}(i), \tag{23}$$

$$\eta^{2}\frac{\alpha(n)}{4\pi^{2}L_{\mathrm{L}}}\max_{i=1,\ldots,N_{\mathrm{EISR}}}\left(\frac{E_{\mathrm{EISR}}(i)}{D_{\mathrm{EISR}}(i,n)}\right) < \min_{i=1,\ldots,N_{\mathrm{N}}}\beta_{\mathrm{R}}(i). \tag{24}$$

Then once node n is activated, a winner-takes-all network with node n as the hub will be established. For proof, see Appendix G.

Basically, from Propositions 2 and 3, one can see that, if a node has a very small link threshold and/or very large amplifying factor, while all other nodes have a very large link threshold, ripple threshold, and/or a very small amplifying factor, then a winner-takes-all network, which is an extreme case of a scale-free network, is likely to be generated by the proposed RSNM. If only a subset of nodes, not all N_{N} nodes in the network, satisfy the conditions in Propositions 2 or 3, then a winner-takes-all pattern will appear in the subset. Suppose a very few subsets, which may vary in size and overlap each other, exist satisfying the conditions in Propositions 2 or 3, and together they cover all nodes in the network. Then it is likely a scale-free network topology will be generated, where a few hub nodes have connections to most other nodes.

However, one may say that either Proposition 2 or 3 fails to give an explicit relationship between the RSRPs and a general power law. As will be further discussed in Sec. IV, the proposed RSNM is a rather complex and flexible model, and many RSRPs as well as the spatial distribution of nodes work together to determine the output network topology. Compared with using a simple power law to model scale-free networks, the complexity and flexibility of RSNM may be more helpful to disclose why a power law exhibits in a particular real-world complex network. For instance, allowing different $\beta_{\mathrm{R}},\beta_{\mathrm{L}}$, and α for the nodes actually reflects the diversity between real-world individuals represented by nodes. In the study of cultures, a country with a rich history may be more resistant to the influence of external cultures, therefore, we may apply larger β_{R} and β_{L} to the associated node in the network; in the study of airline route networks, a hub airport is usually associated with a major economic city, so we may set up α and β_{L} according to, say, the population and the gross domestic product (GDP) of the associated city. By adjusting the RSRPs and comparing the output topology with the real-world network under consideration, e.g., a hub-and-spoke airline route network [11–13], we can then study the influence of different real-world factors by analyzing the values of the RSRPs.

IV. MODIFICATION AND EXTENSION OF RSNM

It should be emphasized that this paper aims to introduce the RSNM, a unique idea to model complex networks. Sections II and III have only described a very basic model of RSNM. Therefore, it is necessary here to give some discussions regarding the full potentials of the proposed RSNM.

A. Flexibility in modification

At first glance, the RSNM proposed in this paper seems much more complicated than those existing models for

complex networks such as discussed in Refs. [7–10]. The good news is, because of the complexity of the RSNM, there is great freedom and flexibility to modify the basic RSNM described in Secs. II and III in order to change and/or improve certain properties of RSNM, or to study certain networks in a specific problem.

As discussed in Sec. III E, allowing different thresholds and amplifying factor for nodes may lead to scale-free networks. Another way to make the RSNM to generate power-law degree distribution is to introduce energy feedback behavior. Let us suppose that when the ripple of a node activates another node and establishes a link between the two nodes, some energy will feedback from the newly activated node to the stimulating ripple, in other words, the point energy of a node will be increased every time when its ripple activates and connects another node. This phenomenon is common in many social networks. For example, the reputation and/or influence of a person will increase when he or she has won trust from more other persons. It is easy to see that the effect of energy feedback behavior is similar to the effect of preferential attachment in Ref. [9]. Therefore, this energy feedback behavior can also contribute to the generation of scale-free topologies.

According to Proposition 1, a node in the basic RSNM will connect to and only to those nodes which are within the distance range estimated by D_{DD}. However, in a real-world random network, a node may not link to a spatially close node but to a further away node. Therefore, we need to modify the basic RSNM to allow more random connections. Actually, this job can be done easily. For instance, we can use different thresholds and amplifying factor for nodes. In many real-world networks, nodes often have different characteristics, e.g., people are different in terms of physical fitness and immunity in the study of the breakout of plagues. Therefore, we can randomly assign to nodes different thresholds and amplifying factors according to the relevant statistic study, say, a survey on people's physical fitness level and immunity. In this case, it is possible that a node may connect to another further away node rather than some nearby nodes. These kinds of connections will be random in some sense because the characteristics of nodes are generated in a random manner. Another way to achieve random connections to further away nodes is to allow energy feedback behavior. Let us assume that a node may be activated when and only when it is reached by a ripple in the current time instant. In other words, if a node is reached but not activated by a ripple, then this node will never be activated by this ripple even if the ripple's energy is enhanced later through energy feedback. In this way, even if all nodes have the same characteristics, a node may not connect to nearby nodes, but it could link to some further away nodes.

Actually, there are many other possibilities to modify the basic RSNM. In particular, it is very easy to modify the basic RSNM in order to develop a unique model for the study of a specific problem. For instance, the impact of earthquakes may not spread as a circle, so we may redefine the shape of ripples accordingly; in the study of the breakout of plagues, a person may be reinfected after he or she has recovered from previous infection, so we can allow nodes to be activated for multiple times; we can even allow a node to generate several ripples of different speeds and amplitudes at one time, in order to simulate, say, the comprehensive international influence of a country (obviously, the military power, economy, and culture of a country may have different influencing behaviors and patterns). These possible modifications make the proposed RSNM highly suitable for the study of various real-world complex network systems.

B. Extension to pure stochastic model

As emphasized in Sec. II, the proposed RSNM is a deterministic model of random networks, i.e., a random network is described by a few RSRPs, and once the values of the parameters are given, the network topology will be uniquely determined. The deterministic feature of the RSNM is achieved by the ripple-spreading process proposed above, and the random features of networks are reflected by the random setup of RSRPs. However, it should be noted that the deterministic model is just one submodel achievable by the idea of using ripple-spreading processes to model random networks. Actually, by slightly modifying some parts in the deterministic model, one can get a semideterministic model and a stochastic model with exactly the same ripple-spreading concept.

Steps 2.3 and 2.5 in the network modeling process in Sec. II define the details of the two behaviors of each node. In the deterministic model, if and only if a threshold is reached, the associated behavior will occur. Therefore, once the distribution of nodes is given and fixed, the network topology is thoroughly determined by the RSRPs, as illustrated in Fig. 5(a).

Based on the deterministic model, one can easily introduce some stochastic features as following. In Step 2.3 or 2.5, suppose a node is newly reached by a ripple. If the current point energy of the stimulating ripple is above a threshold of this node, then the node behaves as described in Step 2.3 or 2.5. In the case where no threshold of the node is reached, no action is defined for the node in the deterministic model, while in the semideterministic model, the node may still be activated or connected according to a certain probability function. For instance, when no threshold is reached, i.e.,

$$\beta_{\mathrm{R}}(i) > e_{\mathrm{EISR}}(j,t) \quad \text{or} \quad \beta_{\mathrm{R}}(i) > e_{\mathrm{N}}(j,t), \tag{25}$$

$$\beta_{\mathrm{L}}(i) > e_{\mathrm{EISR}}(j,t) \quad \text{or} \quad \beta_{\mathrm{L}}(i) > e_{\mathrm{N}}(j,t), \tag{26}$$

node i, the node which is newly reached by a ripple, will generate a responding ripple at a probability of $p_{\mathrm{R}}(i)$,

$$p_{\mathrm{R}}(i) = 2^{\omega_{\mathrm{R}}(1-\frac{\beta_{\mathrm{R}}(i)}{e_{\mathrm{EISR}}(j,t)})} \quad \text{or} \quad p_{\mathrm{R}}(i) = 2^{\omega_{\mathrm{R}}(1-\frac{\beta_{\mathrm{R}}(i)}{e_{\mathrm{N}}(j,t)})}, \tag{27}$$

and/or establish a connection at a probability of $p_{\mathrm{L}}(i)$,

$$p_{\mathrm{L}}(i) = 2^{\omega_{\mathrm{L}}(1-\frac{\beta_{\mathrm{L}}(i)}{e_{\mathrm{N}}(j,t)})}, \tag{28}$$

where $\omega_{\mathrm{R}} > 0$ and $\omega_{\mathrm{L}} > 0$ are tail-off coefficients. The probability functions defined by Eqs. (27) and (28) imply that with less point energy, the smaller the probability of being activated or connected, as illustrated in Fig. 5(b). Obviously, in a final network topology of the semideterministic model, some connections are thoroughly determined by the RSRPs, just as in the deterministic model, while the other connections are largely established in a random manner.

The third submodel is a stochastic model, where all connections in a final network topology are determined partially

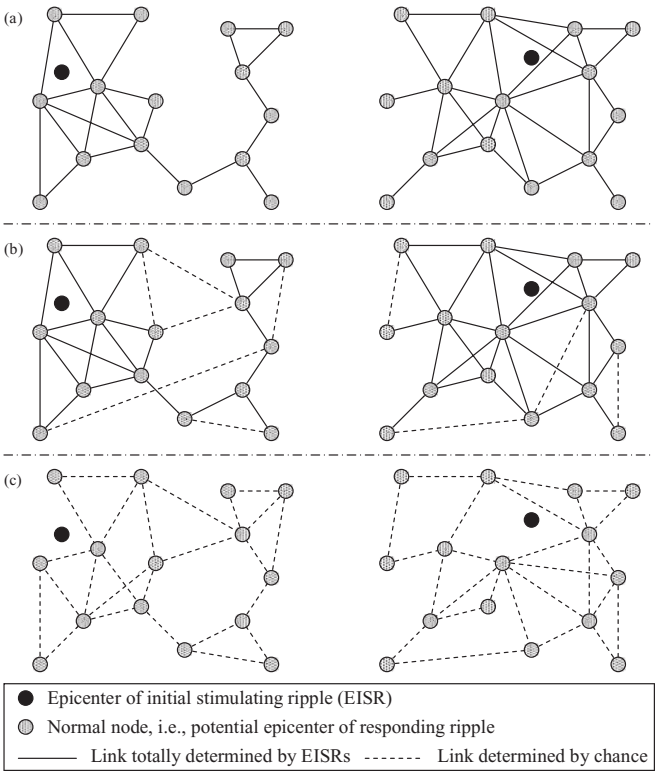

FIG. 5. Three ripple-spreading submodels of random networks. (a) Deterministic model: Whether or not a node will be linked totally depends on the point energy of the stimulating ripple. In other words, the node will be linked if and only if the point energy is above a specific threshold value. Therefore, for a given set of nodes, once the threshold value and the energy amplifying factor are set up, the network topology is completely determined by EISRs, i.e., the number, the distribution, and the energy of EISRs. Following are two examples, where different locations for the EISR result in different networks. (b) Semideterministic model: A node will be linked if the point energy of the stimulating ripple is above a specific threshold value. Otherwise, the probability of the node being linked is a function of the threshold value and the point energy. Therefore, for a given set of nodes, once the threshold value and the energy amplifying factor are set up, some links are completely determined by EISRs, while other links are generated somehow by chance, as illustrated in the following. (c) Stochastic model: The probability of a node being linked is a function of the threshold value and the point energy. Therefore, for a given set of nodes, once the threshold value and the energy amplifying factor are set up, all links are determined partially by EISRs, and partially by chance, as illustrated in the following.

by the RSRPs, e.g., the distribution of EISRs, and partially by chance, i.e., according to certain probability functions such as

$$p_{\mathrm{R}}(i) = 2^{\omega_{\mathrm{R}}(1-\frac{E_{\max}}{e_{\mathrm{EISR}}(j,t)})} \quad \text{or} \quad p_{\mathrm{R}}(i) = 2^{\omega_{\mathrm{R}}(1-\frac{E_{\max}}{e_{\mathrm{N}}(j,t)})}, \tag{29}$$

$$p_{\mathrm{L}}(i) = 2^{\omega_{\mathrm{L}}(1-\frac{E_{\max}}{e_{\mathrm{EISR}}(j,t)})} \quad \text{or} \quad p_{\mathrm{L}}(i) = 2^{\omega_{\mathrm{L}}(1-\frac{E_{\max}}{e_{\mathrm{N}}(j,t)})}, \tag{30}$$

where

$$E_{\max} = \max\left[\alpha(j)E_{\mathrm{EISR}}(i)\right], \quad i = 1,\ldots,N_{\mathrm{EISR}}, \quad j = 1,\ldots,N_{\mathrm{N}}. \tag{31}$$

Clearly, Eq. (31) guarantees no probability is larger than 1.

From the three submodels discussed above, one can see that, although the RSNM originally aims to provide a deterministic method to describe random networks, it is completely compatible to the stochastic features of random networks. The stochastic features can be introduced on two levels. One level is composed of the RSRPs. By assigning random values to these parameters, one can get random topologies, even in the deterministic model. The other level includes the probability functions such as given by Eqs. (27)–(30). These probability functions allow stochastic features even for a set of fixed RSRPs, just as in the semideterministic model and the stochastic model.

Similar to the deterministic model, there is also great freedom and flexibility to modify the semideterministic model and the stochastic model. For instance, the probability functions given in Eqs. (27)–(30) are just some examples, and may be changed according to the specific problem under consideration. Besides continuous and monotonic probability functions as Eqs. (27)–(30), piecewise and U-shaped probability functions could also be the options in the study of certain problems.

C. Combination with population-based algorithms

The proposed RSNM makes it possible to design highly efficient algorithms to resolve the network topology optimization problem. As is well known, the optimization of network topology is a NP-hard (non-deterministic polynomial-time hard) problem. Population-based algorithms, such as genetic algorithms (GAs), have the potential to resolve this problem [14–16], but the widely used data structures, such as adjacency matrix and list of edges, to record network topologies may usually jeopardize the scalability and practicability of population-based algorithms. Basically, the adjacency matrix is memory expensive and therefore can be used hardly by population-based algorithms to optimize large-scale networks. List of edges is not friendly to some important evolutionary operations such as crossover in GAs. If list of edges is adopted in GAs, then additional computationally expensive measures have to be introduced, otherwise crossover will become destructive rather than effective to find good topologies. For instance, direct crossover of two lists of edges may cause the same edge to appear in a chromosome twice, which is infeasible; it is difficult to identify common edges shared by two lists of edges [16,17]. Thanks to the RSNM reported in this paper, a population-based algorithm may only need to evolve some RSRPs, regardless of the network scale, in order to optimize the network topology. For example, suppose we want to generate evenly distributed connections between nodes in Fig. 3, where the desirable average degree is 6, and then the objective function J may be the sum of the power of the node degree deviation from the desirable average degree. From Fig. 3 one can see that, even by simply evolving the location of EISRs, the value of the objective function J may be reduced significantly, e.g., from 8404 in Fig. 3(a) to 2063 in Fig. 3(d). Figure 6 gives a simple example to illustrate different data structures that GAs may use to construct chromosomes for the optimization of network topologies. Based on RSRPs rather than the adjacency matrix or list of edges, population-based algorithms will have better scalability and higher computational efficiency in the optimization of large-scale

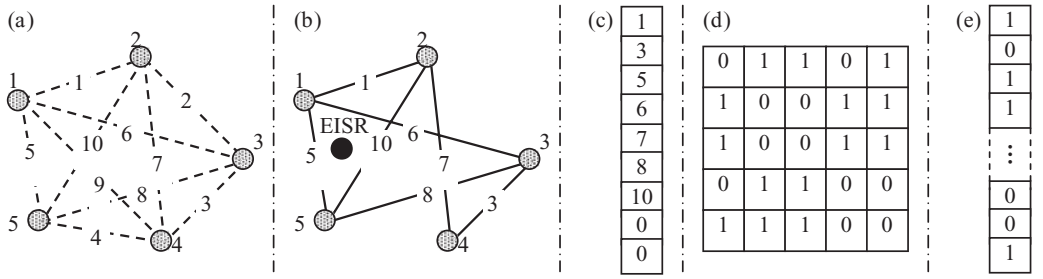

FIG. 6. Chromosome structures: (a) Label nodes and links. (b) An actual network. (c) Corresponding vector representation [each gene records the serial number of an actual connection, i.e., a gene $C(j)=i>0$ means link i is deployed, $C(j)=0$ has no meaning. The length of a chromosome is the upper bound for actual connections, L_{UB}, which is no more than $N(N-1)/2$]. (d) Corresponding matrix representation [a gene $C(i,j)=1$ means node i and node j are linked. The size of a chromosome is N^2. For undirected networks, the matrix may be simplified into a vector of $N(N-1)/2$ genes]. (e) Binary representation based on the RSNM (the binary string records the values in binary format of ripple-spreading related parameters, such as coordinates and energy of EISR, thresholds, and amplifying factor; chromosome size depends on the required minimal sampling steps).

networks. In particular, the proposed RSNM combined with population-based algorithms is suitable for describing and optimizing spatially embedded networks, which have a solid real-world application background. For instance, communications networks may involve range-dependent links; social networks may involve distance-limited interactions between agents existing in some (possibly abstract) space; transport networks have an obvious spatial embedding; certain spatial patterns often exist in breakout of infectious diseases while, more generally, technological and commercial networks frequently feature some cost-per-distance constraint on connectivity. Developing such applications is out of the scope of this paper, but is definitely an important direction for future research work on the proposed RSNM.

V. SOME SIMULATION RESULTS

In this section simulations will be conducted to study some basic properties of the proposed RSNM. The aim here is not to provide a comprehensive experimental analysis of the model, but to help readers to better understand the concept introduced in this paper. To this end, three existing network models are used for comparative purposes, and they are: (i) the random graph model reported by Erdös and Rényi [7], denoted as RGM, (ii) the grown graph model reported by Barabási and Albert [18], denoted as GGM, and (iii) the spatially embedded random network model reported by Barnett *et al*. [10], denoted as SERNM. Basically, in the RGM, edges are placed at random between a given set of nodes, in the GGM, both edges and vertices are added, and one end of each edge is added with linear preferential attachment, and in the SERNM, edges are added with a probability which is a function of the distance between the associated nodes. In the experiment, some of the most commonly used network properties will be calculated for the networks generated by each model, and then a necessary comparative analysis will be conducted. These network properties are the average path length (APL) between connected nodes, the average distance of established edges (ADEE), the clustering coefficient (CC), the assortativity (ASSO), and the degree distribution (DD).

The design and the setup of the experiment are described as following. In each test, 100 nodes are randomly distributed in the rectangular area defined by two points: $(-1000,-1000)$ and (1000,1000). Based on these 100 nodes, the RSNM is first applied to generate a random network, and the established edges in the new generated network are counted. Then the other three models are applied to the same set of nodes and three networks are generated, and every network should have the same number of established edges as the RSNM network. Then the properties of these four networks generated in this test are calculated and saved. Another new test is then carried out starting with another 100 randomly generated nodes. In total, 100 random tests are conducted. Then we calculate the average network properties for each model based on the 100 networks it has generated during the 100 tests. Regarding the setup of model parameters, for the sake of simplicity, the RSNM only allows EISRs to change randomly, while all other RSRPs are given and fixed during the experiment. These fixed RSRPs are given as $E_{\mathrm{EISR}} = 18850.0$, $\beta_{\mathrm{R}} = \beta_{\mathrm{L}} = 5.0$, and α=1036.7. In the RGM, the probability for random connection is set as 0.15. In the GGM, the probability for linear preferential attachment is formulated as $\min[1,0.05+0.01 \times (\text{the degree of a node})^2]$. In the SERNM, the probability for connection is formulated as a distance-dependent piecewise function: If the distance between two randomly chosen nodes is within [0,424.3], the probability varies between [1,0.8] as a linear function of distance; If the distance is within [424.3,509.1], the probability varies between [0.8,0.1]; Otherwise, the probability varies between [0.1,0]. It should be noted that the above setup of the model parameters is rather random (see Refs. [7–10], and [18]).

The simulation results are given in Table I and Fig. 7, from which one may have the following observations.

(1) The RGM and the GGM have the smallest APLs, followed by the SERNM, and the RSNM proposed in this paper has the largest APL. This is largely because, in the RGM and the GGM, there are many random connections between spatially far-away nodes, and these connections make them more likely to travel between any pair of nodes through just one intermedium node. In the SERNM, there are fewer long connections because more edges are established between nearby nodes. In the RSNM, it is almost impossible to establish an edge between two nodes where the distance between them is larger than a certain threshold. Therefore, in the RSNM

TABLE I. Comparative results between different network models.

	Average path length (APL)	Average distance of established edges (ADEE)	Assortativity (ASSO)	Clustering coefficient (CC)
RGM	2.0311	1015.0	0.0152	0.1346
GGM	2.0451	1006.6	0.0190	0.1723
SERNM	2.1920	605.7	0.0647	0.2039
RSNM	2.7106	366.5	0.2668	0.6897

network, if a pair of nodes is far away from each other, then many intermedium nodes have to be passed in order to travel from one node to the other. This is why the RSNM has the largest APL.

(2) In terms of ADEE, the conclusion is the reverse of that of APL, i.e., the RSNM has the smallest ADEE, while the RGM and the GGM have the largest. This is understandable for the same reason as in the case of APL: The RGM and the GGM have many long connections between spatially far-away nodes, while the RSNM establishes its most edges between close nodes.

(3) When multiplying APL with ADEE, one can get an interesting conclusion: In terms of traveling distance, the RSNM proposed in this paper delivers the most efficient traveling route networks (the average traveling distance between any two nodes is just $\sim$1000), while the RGM and the GGM give the worst (the average traveling distance between any two nodes is over 2000). This implies that the RSNM may be very useful in the topology design of many real-world networks, such as transportation networks and power grids, where the minimization of the overall traveling costs is often the major concern.

(4) Regarding the ASSO, the RGM and the GGM are very close to 0, which is the theoretical value for these two models [19]. This means there is no obvious preference for a network's node to attach to others that are similar or different in any way

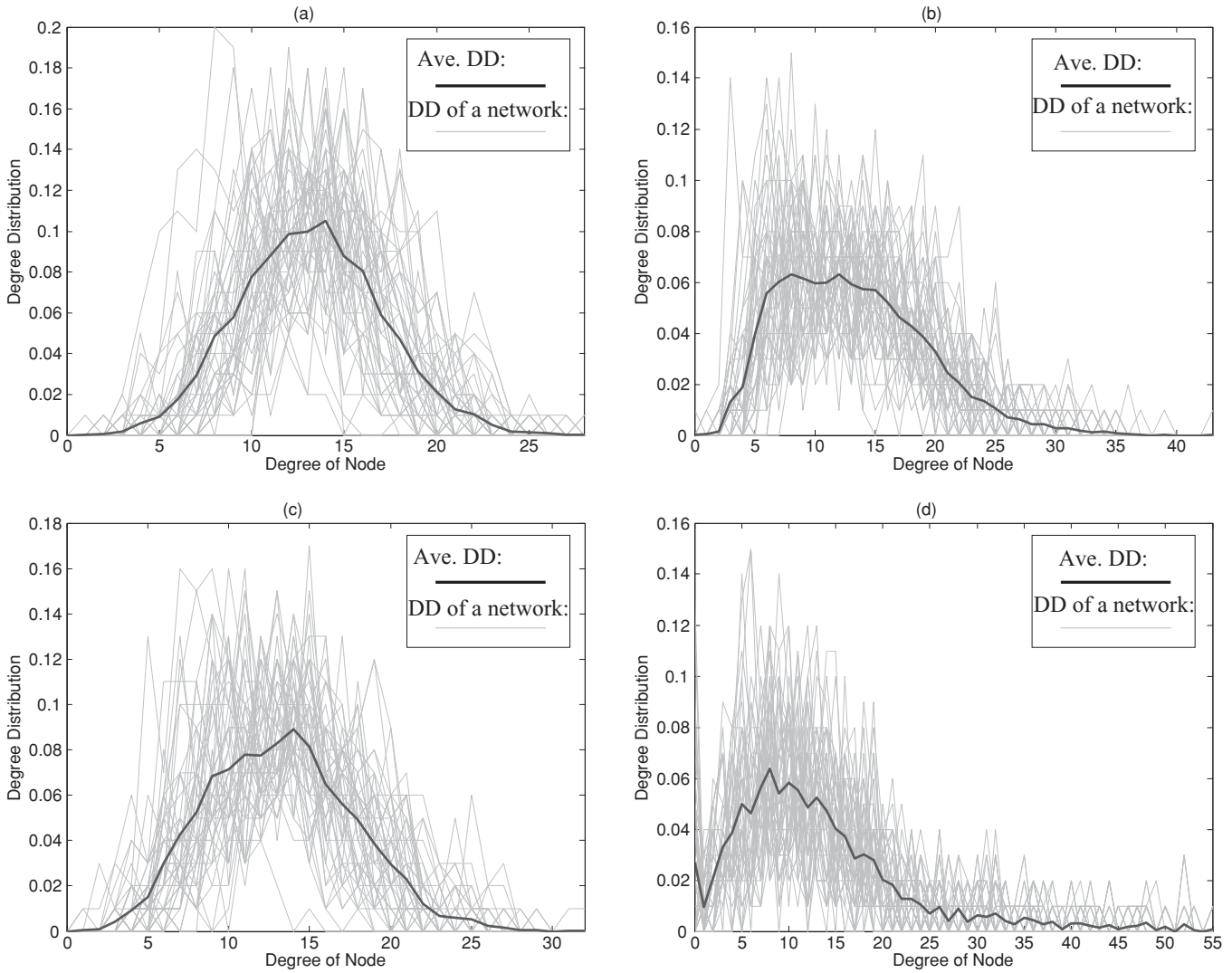

FIG. 7. Degree distributions of different network models: (a) RGM; (b) GGM; (c) SERNM; (d) RSNM.

in these two random network models. The RSNM has the largest ASSO. This is probably because, given that all nodes have the same values for the RSRPs, two close nodes between which an edge has been established are likely to be activated by the same stimulating ripple, then they may have a similar initial point energy, and as a result, they may have a similar impact on their neighborhoods.

(5) Regarding the CC, the RSNM has the largest value, which means that, when compared with other random network models, clusters are more likely to be present in the RSNM networks. Referring to the discussion in Sec. III C, one can see that the mechanism of the RSNM makes it more likely for a subset of spatially close nodes to form a cluster.

(6) In Fig. 7, the gray lines are the degree distributions of all networks in each model, and the black line is the average degree distribution of the associated model. In this experiment, one can see that the RGM has a Poisson distribution with 15 as the center. This is because the probability is set as 0.15 for the RGM, and then for 100 nodes, the average degree should be 15. For the GGM, due to the preferential attachment, a few nodes have very high degrees, while most other nodes have a relatively small degree, $\sim$10. For the reason discussed in Ref. [10], the SERNM also has a Poisson distribution. In the RSNM, the degree distribution is similar to that of GGM. This is understandable: Those nodes close to the EISRs will usually be activated with a high initial point energy, and therefore they are capable of connecting more nodes than most other nodes which are far away from the EISRs. In other words, most nodes which are far away from the EISRs only have small degrees, but those nodes close to the EISRs (the number of such nodes is relatively much smaller) will have much larger degrees. From Fig. 7(d), one may see intuitively that the RSNM is capable of matching preferential attachment, and therefore has the potential of generating scale-free networks.

(7) One may notice that, in Fig. 7(d), sometimes there are some nodes whose degree is 0, which means they are not connected to the network. This is because, due to the randomness in the generation of 100 nodes, it is likely that the RSRPs and the D_{TSP} do not satisfy either Condition 4 or 5. Allowing some nodes to remain unconnected is common in many real-world network systems. For instance, in the breakout of plagues, people living in isolated rural areas are not likely to be influenced.

(8) It should be emphasized that this section of simulation results is mainly used to help to demonstrate the concept of the proposed RSNM within a limited space. Although some observations have been made as given above, they still need to be investigated in more depth in future work. For instance, all RSRPs should be allowed to change randomly, so that their roles in the model can be studied and fully understood; a wide range of nodes rather than just 100 nodes should be used to conduct tests; in particular, some purpose-designed experiments will be important in the application of the RSNM to simulate a specific real-world complex system.

VI. CONCLUSIONS AND FUTURE WORK

Inspired by the natural ripple-spreading phenomenon, this paper reports a deterministic method to model complex networks, which is called the ripple-spreading network model (RSNM). A ripple-spreading process defined by some ripple-spreading related parameters (RSRPs) is the central piece of the model. Basically, assuming that the nodes of networks are distributed in a space (real or artificial) and the distribution of the nodes is fixed, then some initial ripples are randomly generated in the space. As an initial ripple spreads out in the space, its point energy decays gradually, and it reaches every node one by one sooner or later. By comparing the point energy of an incoming ripple with some preset thresholds for a node, it can be determined whether this node will be activated by the incoming ripple to generate a new ripple, and whether this node will be connected to the node where the incoming ripple originates. When a node is activated to generate a new ripple, the initial energy of the new ripple will be a function of the point energy of the incoming ripple. New ripples are also able not only to activate other nodes to generate more ripples, but also to establish new connections between nodes, as long as their point energy is above the relevant thresholds. As this ripple-spreading process goes on for a while, a network topology will appear. Actually, the output topology will be fully determined by the values of RSRPs, such as the locations of the epicenters for initial ripples, the thresholds to tell whether a node will be activated or connected, the energy amplifying factor, and the coefficients to define the point-energy decaying rate. Once the values for these RSRPs are given and fixed, then the output topology will be unique.

Compared with existing models for random networks, the RSNM proposed in this paper has some distinguishing advantages: (i) The ripple-spreading process embedded in the model can naturally capture many spatial and temporal characteristics of many real-world complex networks, where the development of network topology is largely and/or partially based on the spreading out of the influence of some initial trigger events. (ii) There is great flexibility and freedom to modify and extend the deterministic RSNM, e.g., to develop more stochastic versions of the model such as semideterministic RSNM and stochastic RSNM. (iii) The proposed RSNM is very friendly and compatible to population-based algorithms, which makes it possible to develop highly efficient population-based algorithms to optimize the topology of large-scale networks.

It should be pointed out that major objective of this paper is not to discuss all details of the proposed RSNM, but to introduce the concept of RSNM. Therefore, extensive efforts, both theoretical and practical, are still required in future research, in order to obtain a comprehensive understanding of the RSNM and also to fulfill the full potential of the RSNM in extensions and applications. Some directions for future research include the following: (i) Conduct a full-scale theoretical study and statistical analysis of the RSNM in terms of complex network properties, and develop new conditions and theories. (ii) Based on some real-world systems, develop some problem-specific submodels by increasing the complexity of the embedded ripple-spreading process. (iii) Test the RSNM on some real-world networks and systems and compare with other models. (iv) Develop new population-based algorithms based on the RSNM, in order to optimize the topology of some real-world networks and systems.

ACKNOWLEDGMENTS

This work was supported by the Open Fund from the State Key Laboratory of Earth Surface Processes and Resource Ecology, Beijing Normal University, China, and the International Cooperation Project (2010DFB20880) "Integrated Risk Governance—Models and Modeling".

APPENDIX A: PROOF OF CONDITION 1

First we prove that all nodes will be activated once a node is triggered by an initial stimulating ripple. For any node which is first reached and then activated by an initial stimulating ripple, one always has that the initial energy of the ripple of the node satisfies

$$E_{\mathrm{N}}(i) \geqslant \alpha\beta_{\mathrm{R}}. \tag{A1}$$

If node j has not been activated by any initial stimulating ripple, then assume it is first reached by the ripple of node i. Then one has the point energy of the ripple of node i when it reaches node j,

$$e_{\mathrm{N}}[i, D_{\mathrm{N}}(i,j)] = \eta\frac{E_{\mathrm{N}}(i)}{2\pi D_{\mathrm{N}}(i,j)} \geqslant \eta\frac{\alpha\beta_{\mathrm{R}}}{2\pi L_{\mathrm{U}}} \geqslant \max(\beta_{\mathrm{R}}, \beta_{\mathrm{L}}), \tag{A2}$$

which means node j will be activated by the ripple of node i.

Therefore once a node is activated by an initial stimulating ripple, all other nodes will be activated either by initial stimulating ripples or by the ripples of nodes.

Now we prove all potential edges between nodes will be established. According to Eq. (A2), one can see that node i will connect any node that its ripple reaches within distance L_{U}. Since $L_{\mathrm{U}} \geqslant D_{\mathrm{N}}(i,j)$ for $j = 1, \ldots, N_{\mathrm{N}}$, node i will connect all other nodes. Since every node will be activated and have its own ripple satisfying Eq. (A2), every node will connect all other nodes. Therefore, all potential edges between nodes will be established to form a completely connected graph. ■

APPENDIX B: PROOF OF CONDITION 3

Due to limited space, we skip the details of this proof, which is similar to the proof of Condition 1. One can easily see that Eq. (14) is equivalent to

$$\eta\frac{E_{\mathrm{EISR}}(i)}{2\pi \max_{j=1,\ldots,N_{\mathrm{N}}} D_{\mathrm{EISR}}(i,j)} \geqslant \beta_{\mathrm{R}}, \tag{B1}$$

$$\eta^2\frac{\alpha E_{\mathrm{EISR}}(i)}{4\pi^2 L_{\mathrm{U}} \max_{j=1,\ldots,N_{\mathrm{N}}} D_{\mathrm{EISR}}(i,j)} \geqslant \beta_{\mathrm{L}}. \tag{B2}$$

Equation (B1) guarantees that all nodes will be activated, and Eq. (B2) guarantees that every node will connect all other nodes. ■

APPENDIX C: PROOF OF CONDITION 4

Equations (15) or (16) can guarantee that at least all edges included in the shortest open TSP route will be established. Therefore, all nodes will be connected. ■

APPENDIX D: PROOF OF CONDITION 5

Similar to the proof of Condition 3, it is easy to see that Eq. (17) can guarantee that at least a fan that connects all nodes will be formed [the definition of a fan is illustrated in Fig. 4(c)]. ■

APPENDIX E: PROOF OF CONDITION 6

Equation (18) can guarantee that, starting from the first node activated by an initial stimulating ripple, every node will be activated at least by the ripple of its neighbor node along the open TSP route. ■

APPENDIX F: PROOF OF PROPOSITION 2

Proof: Equation (20) guarantees node n will be connected by all other nodes. The second inequality in Eq. (20) is much easier to check. Equation (21) makes sure no links between other nodes will be established. ■

APPENDIX G: PROOF OF PROPOSITION 3

Proof: Equation (22) guarantees that, once node n is activated, it will connect to all other nodes. The second inequality in Eq. (22) is much easier to check. Equations (23) and (24) make sure no other nodes will be activated to generate ripples, and therefore no connection will be established between these nodes. ■

[1] S. Boccaletti, V. Latora, Y. Moreno, M. Chaves, and D. U. Hwang, Phys. Rep. **424**, 175 (2006).

[2] R. Albert and A.L. Barabasi, Rev. Mod. Phys. **74**, 47 (2002).

[3] M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. E **64**, 026118 (2001).

[4] F. Liljeros, C. R. Edling, L. A. N. Amaral, H. E. Stanley, and Y. Aberg, Nature (London) **411**, 907 (2001).

[5] S. Lawrence and C. L. Giles, Nature (London) **400**, 107 (1999).

[6] D. J. Watts and S. H. Strogatz, Nature (London) **393**, 440 (1998).

[7] P. Erdös, and A. Rényi, Publ. Math. (Debrecen) **6**, 290 (1959).

[8] W. Aiello, F. Chung, and L. Lu, in *Proceedings of the 32nd ACM Symposium on the Theory of Computing* (ACM, New York, 2000), p. 171.

[9] M. E. J. Newman, Phys. Rev. E **64**, 025102(R) (2001).

[10] L. Barnett, E. Di Paolo, and S. Bullock, Phys. Rev. E **76**, 056 (2007).

[11] G. Burghouwt, J. Hakfoort, and J. R. Van Eck, J. Air Transp. Manage. **9**, 309 (2003).

[12] B. Sridhar, G. Broto Chatterji, and S. Randall Grabbe, IEEE Trans. Intell. Transp. Syst. **1**, 190 (2000).
[13] J. K. Brueckner and P. T. Spiller, J. Law Econ. **37**, 379 (1994).
[14] Z. H. Xin and H. J. Zhang, Int. Trans. Oper. Res. **9**, 427 (2002)
[15] E.-S. M. El-Alfy, Comp. Comm. **30**(9) 2010 (2007).
[16] X. B. Hu and E. Di Paolo, in *Springer book of Nature Inspired Cooperative Strategies for Optimization (NICSO 2007), Studies in Computational Intelligence (SCI)* 129, p. 495 (Springer-Verlag Berlin Heidelberg, 2008).
[17] X. B. Hu, E. Di Paolo, and L. Barnett, in *Proceedings of IEEE Congress on Evolutionary Computation 2008, CEC 2008 of IEEE World Congress on Computer Intelligence* (IEEE, Hong Kong, 2008), p. 3642.
[18] A.-L. Barabási and R. Albert, Science **286**, 509 (1999).
[19] M. E. J. Newman, Phys. Rev. Lett. **89**, 208701 (2002).