Towards Detecting Previously Undiscovered Interaction Types in Networked Systems

Wenjie Jia, Linyuan Lü *Member, IEEE*, Manuel Sebastian Mariani, Yueyue Dai, and Tao Jiang *Fellow, IEEE*

***Abstract*—Studying networked systems in a variety of domains, including biology, social science and internet of things, has recently received a surge of attention. For a networked system, there are usually multiple types of interactions between its components, and such interaction type information is crucial since it always associated with important features. However, some interaction types which actually exist in the network may not be observed in the metadata collected in practice. This paper proposes an approach aiming to detect previously undiscovered interaction types (PUITs) in networked systems. The first step in our proposed PUIT detection approach is to answer the following fundamental question: is it possible to effectively detect PUITs without utilizing metadata other than the existing incomplete interaction type information and the connection information of the system? Here, we first propose a temporal network model which can be used to mimic any real network and then discover that some special networks which fit the model shall a common topological property. Supported by this discovery, we finally develop a PUIT detection method for networks which fit the proposed model. Both analytical and numerical results show this detection method is more effective than the baseline method, demonstrating that effectively detecting PUITs in networks is achievable. More studies on PUIT detection are of significance and in great need since this approach should be as essential as the previously undiscovered node type detection which has gained great success in the field of biology.**

***Index Terms*—Networked systems, previously undiscovered interaction type, interaction type detection.**

I. Introduction

NETWORKED systems, ordinarily modeled as networks, in a broad range of fields, including biology [1]–[3], the study of human behavior [4]–[6] and the field of internet of things [7]–[10], usually involve multiple interaction types between components leading them to exhibit heterogeneous structures [11]–[13]. An example is the spreading network of COVID-19 which is composed as follows: (1) initialize the virus spreading network (VSN) $\mathbf{G}$ as an empty network without nodes and edges; (2) if an infected person v_i infects another one v_j, then add nodes v_i and v_j to $\mathbf{G}$ and connect v_i and v_j with a directed edge $v_i \rightarrow v_j$ (if the node to add already exists then do nothing). As COVID-19 mutates very frequently and has a lot of variants [14], the collected VSN should be a joint spreading network of COVID-19 and its variants. Marking these viruses with different labels, we can find each directed edge in the spreading network owns a label-set. Specifically, if node v_i passes virus C_s to node v_j, then assign a label C_s to edge $v_i \rightarrow v_j$. Note that node v_i could be infected by multiple species of viruses and is able to spread these viruses to others simultaneously. Consequently, each directed edge in the spreading network should be attached with at least one label. In other words, each edge could have multiple interaction types at the same time, and multiple interaction types exist in such VSN.

However, for a given networked system, it is usually the case that our knowledge on the system is limited to its connection information between components and the collected incomplete interaction type information. On one hand, the situation that only incomplete interaction type information is available implies that there exist previously undiscovered interaction types (PUITs) in the system. For instance, in a VSN whose edge labels refer to virus species, the complete interaction type information of the network should be hardly to be gained, since the resource and time costs for large-scale RNA sequencing is extremely high. This fact implies that there could exist previously unknown variants of COVID-19 hidden in the VSN. On the other hand, besides the collected interaction type information, what information we can utilize to detect PUITs most of the time is limited to the connection information between components. These two observations motivate us to propose a PUIT detection problem for networked systems: find out the connections between components that own PUITs merely by the connection information and the collected incomplete interaction type information. Solving this problem is of great benefit. For example, these techniques would help the World Health Organization (WHO) and countries to speed up the discovering of new virus variants and reduce the large-scale RNA sequencing costs. However, we realize that the proposed problem is an unsolved detection problem since all the existing results related to this problem as far as we know cannot give it a solution.

We first map the proposed PUIT problem in networked systems to an edge label detection problem in complex networks which have been studied for years [15]–[25]. Inspired by the

Manuscript received January 26, 2022; revised April 8, 2022; accepted May 7, 2022. Date of publication XX; date of current version May X, 2022. This work was supported in part by the National Key RD Program of China with Grant number 2019YFB1803400, and National Science Foundation of China with Grant number U21B2036.

Wenjie Jia is with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, P. R. China (e-mail: wenjiejia@hust.edu.cn). Yueyue Dai and Tao Jiang are with the Research Center of 6G Mobile Communications, School of Cyber Science and Engineering, and Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology, Wuhan 430074, P. R. China (e-mail: yueyuedai@hust.edu.cn, Tao.Jiang@ieee.org). Linyuan Lü is with the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 611731, P. R. China, and the Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Huzhou 313001, P. R. China (e-mail:linyuan.lv@uestc.edu.cn). Manuel Sebastian Mariani is with the Institute of Fundamental and Frontier Sciences, University of Electronic Science and Technology of China, Chengdu 611731, P. R. China, and URPP Social Networks, University of Zurich, Zurich CH-8050, Switzerland (e-mail: Manuel.Sebastian.Mariani@gmail.com).

example on VSNs introduced above, for a networked system with multiple interaction types, we can employ an edge label to represent an interaction type and a nonempty label-set to depict what interaction types an edge has. Thus, the PUIT detection problem can be interpreted as a previously undiscovered edge label (PUEL) detection problem. Specifically, we formulate the proposed PUEL (PUIT) detection problem as follows. Let $\mathbf{G} = (\mathcal{V}, \mathcal{E}, L)$ be a network where $\mathcal{V}$ and $\mathcal{E}$ denote the set of its nodes and the set of its edges, respectively. Notation L stands for a mapping which maps each edge to the nonempty label-set associated with the edge. For convenience, we call L the complete edge label information of $\mathbf{G}$. Additionally, for an edge e, we call $L(e)$ e's complete label-set. Let $\mathcal{C} = \cup_{e\in\mathcal{E}} L(e)$. Obviously, $\mathcal{C}$ consists of all the labels that can be observed in $\mathbf{G}$. Assume that $\mathbf{G}$'s complete edge label information L is unavailable and the currently obtained edge label information of $\mathbf{G}$ is denoted by L'. Similarly, L' can be modeled as a mapping by which each edge e in $\mathcal{E}$ is mapped to a subset of $L(e)$. Generally speaking, some edges can be found without label annotated in the collected data. That is to say, L' can map an edge to an empty set. We say L' is incomplete if there is an edge e such that $L'(e) \subsetneqq L(e)$. Let $\mathcal{C}' = \cup_{e\in\mathcal{E}} L'(e)$ be the set consisting of all the known edge labels. If $\mathcal{C}' \neq \mathcal{C}$, we say there exist PUELs in the network, and the task of the PUEL detection is to find out the edges whose complete label-sets have nonempty intersections with the PUEL set $\mathcal{C} \setminus \mathcal{C}_1$.

Now, we explain why existing results are not fit for our proposed problem. Existing research on interaction type (edge label) detection [15]–[19], such as the sign prediction [20]–[25], aims to predict what observed labels an unannotated or incompletely annotated edge has. Thus, current methods are unable to annotate edges with labels that have not appeared in the networked system before. As a result, these methods can neither answer whether there are PUELs in the system nor suggest inspecting which edges in the networked system has the greatest chance to discover PUELs. Consequently, we lack methods for detecting PUELs other than the random guessing method, which refers to the baseline method in this paper.

Besides the fact that all the existing results on label detection are helpless, we are facing another fact which aggravates the difficulty of solving the problem. For a PUEL we know neither what physical meaning and features it has nor the relationships between it and other already observed labels, making the existing yet incomplete edge label information useless in solving the problem. These barriers lead us to ask a fundamental question: is the proposed PUEL detection problem solvable by non-trivial methods (methods other than random guessing)?

To answer the question, this paper leverages the system's connection information to develop a non-trivial and effective PUEL detection method for networks generated by a generic generating network model. Specifically, we first propose a temporal network model called the degradation-evolution network model. Initializing the model with some settings, the model outputs an instance sequence $(\mathbf{G}(t_1), \mathbf{G}(t_1+1), \cdots, \mathbf{G}(0), \cdots, \mathbf{G}(t_2))$ where $t_1 < 0 < t_2$. For an instance $\mathbf{G}(t')$, we call another instance $\mathbf{G}(t'')$ the evolved (degraded) version of $\mathbf{G}(t')$ if $t'' > t'$ ($t'' < t'$). We use $\mathbf{G}(+\infty)$ ($\mathbf{G}(-\infty)$) to denote $\mathbf{G}(t)$'s ultimately evolved (degraded) version. Then, studying the ultimately evolved and degraded versions of an arbitrary network, we discover that the two special networks own a particular topological property. This discovery enables us to develop a PUEL detection method which has perfect (accuracy of 100%) detection performance in networks which are ultimately evolved or degraded. For instance $\mathbf{G}(t)$ which is generated by the model and not ultimately evolved or degraded, the proposed method has various performance for detecting PUELs in $\mathbf{G}(t)$ as t varies. Finally, to evaluate the performance of the proposed method, we apply it to a number of synthetic networks generated by the model, and find that the method's average accuracy is markedly higher than the average accuracy of random guessing. This result answers the question introduced above: there are non-trivial methods for detecting PUEL in networks.

We briefly summarize the paper's contributions as follows.

1. A novel detection approach — detecting previously undiscovered interaction types in networked systems purely from the connection information and the incomplete interaction type information— is proposed which is of practice important but remains unsolved.
2. We introduce a universal network model — degradation-evolution network model — to which an arbitrary real-world directed network can be mapped.
3. We discover an important relationship between network topology and its interaction types (see Section III for details).
4. Based on this discovery, we derive an effective detection method for the unsolved PUEL detection problem.

To sum up, the proposed PUEL detection approach is not only useful but also achievable, and we believe the PUEL detection is as essential and promising as the previously undiscovered node type detection which has gained great success in the filed of biology [1]–[3].

The rest of this paper is organized as follows. In Section II, we introduce the degradation-evolution network model in detail. In Section III, we present our discovery on the topological property of networks which fit the proposed network model. In Section IV, we introduce a non-trivial PUEL detection method and a variant version which better fit a special scheme. In Section IV, we introduce several metrics to evaluate the performance of the methods proposed in Section IV. In Section VI, experimental results are exhibited. The last section concludes the whole paper.

II. Degradation-evolution network model

Let t be the time and $\mathbf{G}$ be a network. We use $t < 0$, $t = 0$ and $t > 0$ to denote the past, the present and the future, respectively. At present (i.e. $t = 0$), we initialize $\mathbf{G} = \mathbf{G}(0)$ to be an arbitrary network with n nodes and m edges with labels in $\mathcal{C}$, where $\mathcal{C} = \{C_1, C_2, \cdots, C_h\}$ represents all the labels that can be observed in the whole course of $\mathbf{G}$'s mutation process. Let $\mathbf{G}(t) = (\mathcal{V}(t), \mathcal{E}(t), L(t))$ be a temporal network with $\mathcal{V}(t) = \{v_1, v_2, \cdots, v_n\}$, for time $t = -\infty, \cdots, -1, 0, 1, \cdots, +\infty$. For an edge $v_i \to v_j$ in $\mathbf{G}(t)$, we employ notation $\mathcal{C}(i, j, t)$ to denote the label-set associated

TABLE I
FREQUENTLY-USED NOTATIONS IN THIS PAPER.

Notation	Interpretation
(Basic notations and functions)	
t	System time
v_i	Node with index i
$v_i \to v_j$	Directed edge from v_i to v_j
$[e]_{src}$	The source node of edge e
$[e]_{tar}$	The target node of edge e
$\chi_E(i,j)$	$\chi_E(i,j)=1$ if $v_i \to v_j \in E$; otherwise $\chi_E(i,j)=0$
$\mathbf{G}$ (resp. $\mathbf{G}(t)$)	Network/system (resp. network/system at time t)
$\mathcal{V}$ (resp. $\mathcal{V}(t)$)	The set of nodes in $\mathbf{G}$ (resp. $\mathbf{G}(t)$)
$\mathcal{E}$ (resp. $\mathcal{E}(t)$)	The set of edges in $\mathbf{G}$ (resp. $\mathbf{G}(t)$)
$(r,\mathbf{G})$-follower	Node v_i is v_j's $(r,\mathbf{G})$-follower if the length of the shortest simple path from v_i to v_j in $\mathbf{G}$ is r
$L'(e)$	The currently obtained edge label-set of edge e
$L(e)$	The complete edge label-set of edge e
$\mathcal{C}$	All the labels that can be observed in $\mathbf{G}$
(Basic notations used in the degradation-evolution network model)	
$\mathbf{G}_l(t)$	The l-th layer of $\mathbf{G}(t)$
$\mathcal{V}_l(t)$	The set of nodes in $\mathbf{G}_l(t)$
$\mathcal{E}_l(t)$	The set of edges in $\mathbf{G}_l(t)$
$\mathcal{C}(i,j,t)$	The label-set associated with the ordered node-pair (v_i,v_j) at t
$A_{i,l}$	Node v_i's l-attractiveness
$A_{max,l}$	$A_{max,l}=\max_{v_i\in\mathcal{V}} A_{i,j}$
$P_l(i,t)$	Node v_i's potential energy with respect to the l-th layer of $\mathbf{G}$ at t
$P(i,t)$	Node v_i's potential energy at t
$P(\mathbf{G},t)$	The system $\mathbf{G}$'s potential energy at t
$\Delta(E)$	The Delta-property of edge-set E
(Basic notations used in the PUEL detection scheme in sections IV and V)	
$\mathbf{G}$	A network with two edge labels
C_1	The already-observed edge label in $\mathbf{G}$
C_2	The previously-undiscovered edge label in $\mathbf{G}$
$\mathcal{E}_1$	The set consists of all the edges with label C_1 in $\mathbf{G}$
E_1	All the edges that are observed with C_1
E_1-D-Top	The proposed PUEL detection algorithm based on E_1
n_1-D-Top	The PUEL detection algorithm based on a random subset with n_1 elements in $\mathcal{E}_1$
$\omega(\mathbf{G}\,\|\,E_1)$	The Precision of algorithm E_1-D-Top
$\lambda(\mathbf{G}\,\|\,E_1)$	The AUC of algorithm E_1-D-Top
$\omega_{n_1}(\mathbf{G})$	The Precision of algorithm n_1-D-Top
$\lambda_{n_1}(\mathbf{G})$	The AUC of algorithm n_1-D-Top

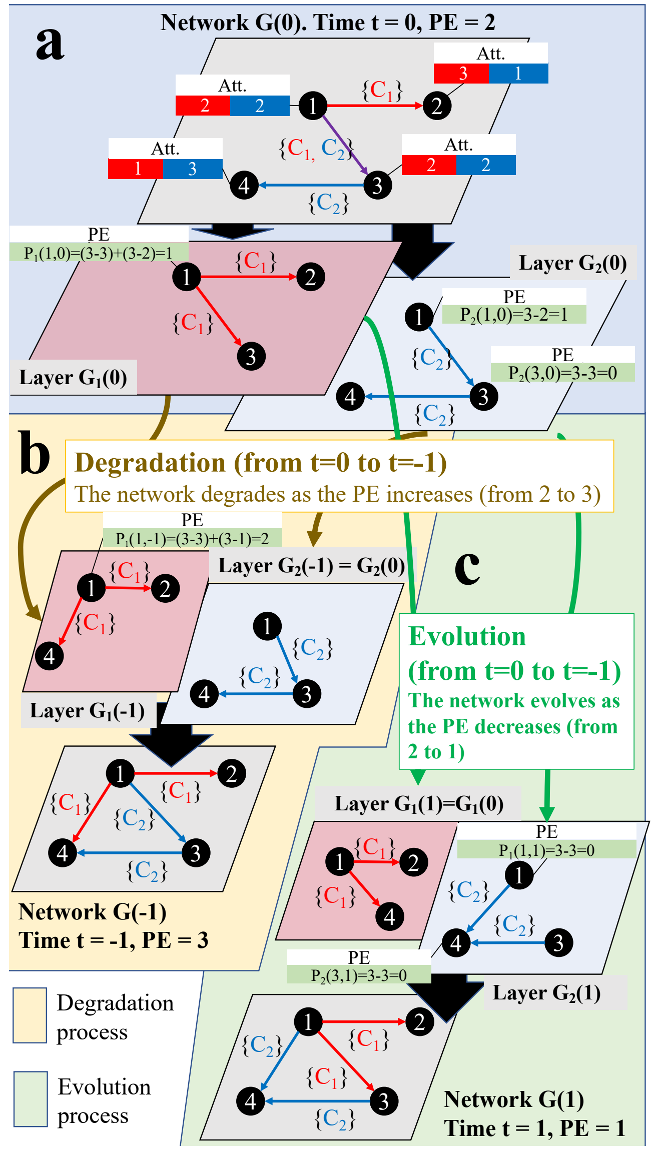

Fig. 1. An illustration of the degradation-evolution network model. In **a**, we initialize network $\mathbf{G}$ at time $t=0$ as a network consisting of 4 nodes and 3 directed edges. Each node is assigned with a pair of 1-attractiveness (att.) and 2-att. The three edges are tagged by two kinds of labels (*i.e.* C_1 and C_2). In **b**, network $\mathbf{G}(0)$ degrades into $\mathbf{G}(-1)$ as its PE increases from 2 to 3. Specifically, a randomly selected node, i.e. the node with index 1, increases its potential energy (PE) by rewiring its out-edges in layer $\mathbf{G}_1(0)$, making the network's PE increase. In **c**, network $\mathbf{G}(0)$ evolves into $\mathbf{G}(1)$ as its PE decreases from 2 to 1. Specifically, node with index 1 rewires its out-edges in layer $\mathbf{G}_2(0)$ making $P_2(1,1) < P_2(1,0)$ and $P(\mathbf{G},1) < P(\mathbf{G},0)$.

with it. As introduced before, every edge in networks should be annotated at least one label. Therefore, there exists an edge from v_i to v_j at time t if and only if $\mathcal{C}(i,j,t) \neq \emptyset$.

Regarding the subnetwork consisting of all the edges with the same label C_s and all the nodes involved in $\mathbf{G}(t)$ as the s-th layer of network $\mathbf{G}(t)$, we can divide $\mathbf{G}$ into h different layers. Specifically, we denote the l-th layer of $\mathbf{G}(t)$ as $\mathbf{G}_l(t) = (\mathcal{V}_l(t), \mathcal{E}_l(t))$ (see Fig. 1**a**). Inspired by the attractiveness model [26], we assume: (1) for $l \in \{1,2,\cdots,h\}$, every node v_i in $\mathbf{G}(t)$ is equipped with an attractiveness $A_{i,l} \geq 0$, called v_i's l-attractiveness or attractiveness associated with layer $\mathbf{G}_l(t)$ (see Fig. 1**a**); (2) different nodes should have distinct nonzero attractiveness associated with the same layer of the network; (3) in each layer of the system, a node always intends to connect to nodes with high attractiveness associated with this layer, and it is able to rewire its out-edges in the layer to better fulfill this intention.

For node v_i we define its potential energy with respect to the l-th layer at time t to be

$$P_l(i,t) = \sum_{j=1}^{n} (A_{max,l} - A_{j,l})\chi_{\mathcal{E}_l(t)}(i,j), \tag{1}$$

where $A_{max,l} = \max_{v_i\in\mathcal{V}} A_{i,l}$ and $\chi_{\mathcal{E}_l(t)}(i,j) = 1$ if $v_i \to v_j \in \mathcal{E}_l(t)$; otherwise $\chi_{\mathcal{E}_l(t)}(i,j) = 0$. We employ $P_l(i,t)$ to describe how eagerly node v_i is willing to rewire its out-edges in $\mathbf{G}_l(t)$ to connect to nodes with higher l-attractiveness at time t (see Fig. 1**a**). Further, we define v_i's potential energy and the system's potential energy at time t to be $P(i,t) = \sum_{l=1}^{h} P_l(i,t)$ and $P(\mathbf{G},t) = \sum_{i=1}^{n} P(i,t)$, respectively. A node's higher potential energy means the stronger desire for this node to rewire its out-edges, and the higher potential energy of a system indicates a more structurally unstable state of this system.

Inspired by existing rewiring processes [27]–[31], we intro-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3174086, IEEE Internet of Things Journal

duce an evolution process: at each time $t > 0$, a node rewires some of its out-edges in a layer of $\mathbf{G}(t)$ and then time t increases by 1, such that $P(\mathbf{G}, t+1) \leq P(\mathbf{G}, t)$ (see Fig. 1**c**). In addition, we assume that there also exists a degradation process. That is, at each time $t < 0$, a node rewires some of its out-edges in a layer of $\mathbf{G}(t)$ and then time t decreases by 1, such that $P(\mathbf{G}, t-1) \geq P(\mathbf{G}, t)$ (see Fig. 1**b**).

We validate our network model from two aspects. On one hand, we show that the proposed model is generic, that is, this model is able to mimic an arbitrary network. Given a networked system $\mathbf{G}' = (\mathcal{V}', \mathcal{E}', L')$, let $\mathcal{V}' = \{v_1, v_2, \cdots, v_n\}$, $\mathcal{C}' = \{C_1, C_2, \cdots, C_s\}$ be the set composed of all the label appeared in $\mathbf{G}'$ and $A_{i,w} = iw$ be node v_i's w-th attractiveness, for $i = 1, 2, \cdots, n$ and $w = 1, 2, \cdots, s$. Obviously, different nodes have distinct attractiveness associated with the same layer. Recalling the initialization of the proposed network model, we can use $\mathbf{G}'$, and $\{A_{i,w}\}_{1\leq i\leq n, 1\leq w\leq s}$ to initialize the degradation-evolution network model. Assume the output is $\{\mathbf{G}(t)\}_{-\infty<t<+\infty}$. Then, one must have $\mathbf{G}' = \mathbf{G}(0)$. This fact shows that any network with multiple edge labels can be always regarded as an instance generated by the proposed network model. On the other hand, we show that the evolution/degradation process is observed extensively in nature, that is, the dynamic mechanism of the network model is reasonable. We exemplify the evolution process with citation networks. In a citation network, a directed edge from paper v_i to paper v_j implies that v_i needs information on some topics, denoted by $C_1, C_2, \cdots, C_s$, provided by v_j. Then we can use a label-set $\{C_1, C_2, \cdots, C_s\}$ to represent edge $v_i \to v_j$. For each $h = 1, 2, \cdots, s$, there are a lot of papers which can provide information on topic C_h throughout the citation network. But the information on topic C_h provided by different papers is demanded by v_i in different degrees. Such demand degree can be modeled as the h-attractiveness. Before a paper gets published, its authors can improve the references to better support their results. Specifically, for each topic C_h, the authors always try to collect papers they need the most. The degree how v_i's authors are willing to improve the references to better meet their demand can be modeled as the potential energy of v_i, and the improvement process of v_i's references can be modeled as the decreasing process of the degree. Similar mechanisms can be also observed in online social networks like Twitter.com, internet networks, etc.

III. Topological property

In the rest of this paper, we always let $\mathbf{G}(t)$ be a network observed at t with $\mathcal{V}(t)$ and $\mathcal{E}(t)$ denoting its nodes and edges. Let $E \subseteq \mathcal{E}(t)$ be a non-empty edge-set. We use notation $\lceil E \rceil_{src}$ ($\lceil E \rceil_{tar}$) to denote the set consisting of all the source (target) nodes of edges in E. We say v_i is a $(2, \mathbf{G}(t))$-follower of v_j in $\mathbf{G}(t)$ if the length of the shortest simple path (a simple path is a path without repeated nodes) from v_i to v_j in $\mathbf{G}(t)$ is 2. Assume that different nodes have different l-attractiveness for any $l \in \{1, 2, \cdots, h\}$. Then we obtain the following lemma (see its proof in Appendix A).

Lemma 1. *Let E be an arbitrary nonempty subset of $\mathcal{E}(t)$. At time $|t| = +\infty$, if all the edges in E lie in the same layer, then there always exists a node in $\lceil E \rceil_{tar}$ having no $(2, \mathbf{G}(t))$-follower in $\lceil E \rceil_{src}$.*

We say $E \subseteq \mathcal{E}(t)$ has the Delta-property, denoted by $\Delta(E) = 1$, if $\delta(E) = \emptyset$, where $\delta(E)$ can be obtained by implementing the following procedures: (1) select a node v from $\lceil E \rceil_{tar}$ which has no $(2, \mathbf{G}(t))$-follower in $\lceil E \rceil_{src}$; (2) remove all the edges whose target node is v from E; (3) repeat (1)-(2) until no more removal is possible; and (4) set the remaining edge-set to be $\delta(E)$. In addition, we define $\Delta(E) = 0$, if $\delta(E) \neq \emptyset$. Then we have the following results (see derivations in Appendices B and C).

Lemma 2. *Implementing the removal operations on E introduced above, we obtain $\{E_1, E_2, \cdots, E_s\}$ and $\delta(E)$, where E_i denotes the edge-set removed from E in the i-th removal operation, for $i = 1, 2, \cdots, s$. Let $\delta(E) = E_{s+1}$. Then we have: (1) $E_i \neq \emptyset$, for $i = 1, 2, \cdots, s$; (2) $E_i \cap E_j = \emptyset$, for $1 \leq i < j \leq s+1$; (3) $\cup_{i=1}^{s+1} E_i = E$; (4) if $\delta(E) \neq \emptyset$, then $|\delta(E)| \geq 2$.*

Lemma 3. *Let E be a non-empty subset of $\mathcal{E}(t)$. Let $\delta^{(1)}(E)$ and $\delta^{(2)}(E)$ be two subsets of E obtained by implementing the removal procedures introduced above. Then we have $\delta^{(1)}(E) = \delta^{(2)}(E)$.*

Lemma 3 shows that mapping Δ is well-defined. Further, we obtain our main theoretical result (see its proof in Appendix D).

Theorem 1. *When $t = \pm\infty$, for an edge-set $E \subseteq \mathcal{E}(t)$ if $\Delta(E) = 0$, then none of the edges in E can have a label in common.*

The above theorem shows that when a network is ultimately evolved or degraded, its multilayer structure must follow a special topological property. Specifically, we can apply this result to judge whether these edges can share common labels. In the following, we show how to utilize this result to detect PUELs.

IV. PUEL detection method

Based on Theorem 1, we derive a method to tackle the PUEL detection problem. Let $|t| = +\infty$. Assume that only partial edges are found with labels in $\mathbf{G}(t)$. Specifically, let $\mathcal{C}(t) = \{C_{t_1}, C_{t_2}, \cdots, C_{t_s}\}$ denote all the edge label observed at time t, $s = s(t)$ be an integer, and $E_{t_l} \in \mathcal{E}_{t_l}(t)$ consist of all the edges which are observed with label C_{t_l} at time t, for $1 \leq l \leq s(t)$. If $C \in \mathcal{C}$ is an edge label with $C \notin \mathcal{C}(t)$, then we call C is PUEL at time t. Let e denote an edge in $\mathcal{E}(t)$ satisfying that $\Delta(\{e, e_{t_l}\}) = 0$ for any $e_{t_l} \in E_{t_l}$ and $l \in \{1, 2, \cdots, s(t)\}$. Then, according to Theorem 1, we have e must own a PUEL. Assembling all of such edges, we obtain an edge-set E' in which every edge has at least one new label. Our theory shows that this detection method's accuracy for $|t| = +\infty$ is perfect (100%). For $|t| < +\infty$, the accuracy of this detection method is case-dependent.

In the rest of this paper, we focus on one of the most straightforward cases of our main problem. Let $\mathbf{G} = (\mathcal{V}, \mathcal{E})$ be a network, C_1 be an already-observed edge label, $\mathcal{E}_1$ be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3174086, IEEE Internet of Things Journal

the set consisting of all the edges in $\mathbf{G}$ carrying label C_1, and $E_1 \subseteq \mathcal{E}_1$ be an edge-set consisting of n_1 edges with label C_1. Our goal is to find out a small number of edges with PUELs based on E_1 and $\mathbf{G}$'s topology. To solve this problem, we assign each edge e in $\mathcal{E}$ a score $\nabla(e|E_1)$ with $\nabla(e|E_1) = |\{e' \in E_1|\Delta(\{e, e'\}) = 0\}|$. Then take the edges with the largest nonzero scores as the algorithm's output. We use notation E_1-D-Top and E_1^* to denote the corresponding algorithm and its output, respectively. Note that for any edge e, $\nabla(e|E_1)$ is an integer and $\nabla(e|E_1) \leq |E_1| = n_1$. Thus, for small n_1, such as $n_1 = 1, 2, 3$, the difference between edges' scores is small, which could impair the performance of our method. For small n_1, we further require that the score of every edge in the output of E_1-D-Top should be n_1 (i.e, $\nabla(e|E_1) = n_1$ for $e \in E_1^*$).

V. Performance evaluation

Two standard metrics are used to quantify the accuracy of detection algorithms: Precision [32] and *area under the receiver operating characteristic curve* (AUC) [33]. Assume that in an output edge-set E_1^* consisting of n_2 edges, there are n' edges are right (*i.e.* there are n' edges are with PUELs), then the Precision of this algorithm is n'/n_2. Here, we use $\omega(\mathbf{G}|E_1)$ to denote the Precision of algorithm E_1-D-Top. Higher Precision means higher detection accuracy. Note that for a given edge-set $E_1 \subseteq \mathcal{E}_1$ with $|E_1| = n_1$, the performance of algorithm E_1-D-Top is closely related to the probability that an arbitrary edge with label C_2 gets a larger score than another arbitrary edge with label C_1, which can be quantified by AUC [33]. To measure the AUC, denoted by $\lambda(\mathbf{G}|E_1)$, we can make N independent comparisons: at each time, we randomly pick an edge with PUELs and an edge without PUELs to compare their scores. If there are N' times the edge with undiscovered labels obtaining a higher score and N'' times they have the same score, then the AUC value is $\lambda(\mathbf{G}|E_1) = (N' + 0.5N'')/N$ [34]. If all the scores are generated from an independent and identical distribution, the AUC value should be about 0.5. Therefore, the degree to which the value exceeds 0.5 indicates how much better the algorithm performs than random guessing.

In this paper, we only consider networks with small numbers of nodes and small numbers of edges, because we can run through all possible combinations of edges without PUELs and edges with PUELs to measure the AUC of the network in this scenario.

We are interested in our method's accuracy in detection PUELs in $\mathbf{G}$, when we are given an edge-set consisting of n_1 edges arbitrarily picked from $\mathcal{E}_1$. We use notation n_1-D-Top to represent algorithm E_1-D-Top, where E_1 is an arbitrary subset of $\mathcal{E}_1$ with n_1 elements. We denote $\omega_{n_1}(\mathbf{G})$ and $\lambda_{n_1}(\mathbf{G})$ as the Precision and the AUC of n_1-D-Top, respectively. Then we obtain

$$\omega_{n_1}(\mathbf{G}) = \frac{1}{\binom{|\mathcal{E}_1|}{n_1}} \sum_{E_1 \subseteq \mathcal{E}_1, |E_1| = n_1} \omega(\mathbf{G}|E_1) \tag{2}$$

and

$$\lambda_{n_1}(\mathbf{G}) = \frac{1}{\binom{|\mathcal{E}_1|}{n_1}} \sum_{E_1 \subseteq \mathcal{E}_1, |E_1| = n_1} \lambda(\mathbf{G}|E_1), \tag{3}$$

where $\binom{|\mathcal{E}_1|}{n_1} = |\mathcal{E}_1|!/[n_1!(|\mathcal{E}_1| - n_1)!]$ is a combination.

For a network $\mathbf{G} = \mathbf{G}(t)$ whose structure varies over time, we are concerned about the accuracy of the proposed algorithms applied to $\mathbf{G}$ at present ($t = 0$) and curious about both what happened to their performance in the past ($t < 0$) and what their performances will become in the future ($t > 0$). Let $\mathbf{G}$ be some network generated by our proposed degradation-evolution model. Denoting the present time as $t = 0$, we can rewrite $\mathbf{G}$ as $\mathbf{G}(0)$ without loss of generality. By our degradation-evolution model, we obtain a family of networks $\{\mathbf{G}(t)\}_{-\infty}^{+\infty}$, which depicts the whole course of $\mathbf{G}$'s mutation process. To study the overall performance of the detection methods, we introduce another parameter ν, which is given by $\nu = \nu(t) = [P(\mathbf{G}, t) - P_{min}(\mathbf{G})]/[P_{max}(\mathbf{G}) - P_{min}(\mathbf{G})]$, where $P_{max}(\mathbf{G})$ and $P_{min}(\mathbf{G})$ denote the supremum and infimum of $P(\mathbf{G}, t)$, respectively. Parameter ν ranges from 0 to 1 and describes the evolution degree of $\mathbf{G}$: the more nearly ν approaches to 0, the more stable the structure of the network is. Then we can rewrite $\{\mathbf{G}(t)\}_{-\infty}^{+\infty}$ as $\{\mathbf{G}[\nu]\}_{\nu \in [0,1]}$ and $\mathbf{G} = \mathbf{G}[\nu_0]$, where $\nu_0 = \nu(0)$. Then the average Precision over time ($\bar{\omega}_{n_1}(\mathbf{G})$) and the average AUC over time ($\bar{\lambda}_{n_1}(\mathbf{G})$) of n_1-D-Top applied to $\mathbf{G}$ can be calculated by

$$\bar{\omega}_{n_1}(\mathbf{G}) = \int_0^1 \omega_{n_1}(\mathbf{G}[\nu]) f(\nu) d\nu \tag{4}$$

and

$$\bar{\lambda}_{n_1}(\mathbf{G}) = \int_0^1 \lambda_{n_1}(\mathbf{G}[\nu]) f(\nu) d\nu \tag{5}$$

respectively, where $f(\nu)$ refers to the probability density function of ν. In this paper, we assume that a random network's potential energy follows a uniform distribution. Thus, we must have ν subjects to a uniform distribution as well.

We study the performance of n_1-D-Top applied to randomly generated networks by the degradation-evolution model. Let $\boldsymbol{\theta}$ denote the initialization settings of the degradation-evolution model. For a given $\boldsymbol{\theta}$ and a given $\nu \in [0, 1]$, the Precision and AUC of n_1-D-Top applied to a random network $\mathbf{G} = \mathbf{G}[\nu]$ which is generated by the proposed model under a specific configuration given by $\boldsymbol{\theta}$, are represented by $\omega_{n_1,\boldsymbol{\theta}}(\nu)$ and $\lambda_{n_1,\boldsymbol{\theta}}(\nu)$, and can be calculated by

$$\omega_{n_1,\boldsymbol{\theta}}(\nu) = \frac{1}{M} \sum_{i=1}^{M} \omega_{n_1}(\mathbf{G}_i) \tag{6}$$

and

$$\lambda_{n_1,\boldsymbol{\theta}}(\nu) = \frac{1}{M} \sum_{i=1}^{M} \lambda_{n_1}(\mathbf{G}_i) \tag{7}$$

where $\mathbf{G}_i$ is a random network generated by the degradation-evolution model with specific parameters $\boldsymbol{\theta}$ and admitting $\mathbf{G}_i = \mathbf{G}_i[\nu]$, for $i = 1, 2, \cdots, M$. Finally, the Precision and AUC of n_1-D-Top applied to a randomly generated network by the model with a specific parameter setting $\boldsymbol{\theta}$, can be represented as

$$\omega_{n_1,\boldsymbol{\theta}} = \int_0^1 \omega_{n_1,\boldsymbol{\theta}}(\nu) f(\nu) d\nu \tag{8}$$

and

$$\lambda_{n_1,\boldsymbol{\theta}} = \int_0^1 \lambda_{n_1,\boldsymbol{\theta}}(\nu) f(\nu) d\nu. \tag{9}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3174086, IEEE Internet of Things Journal

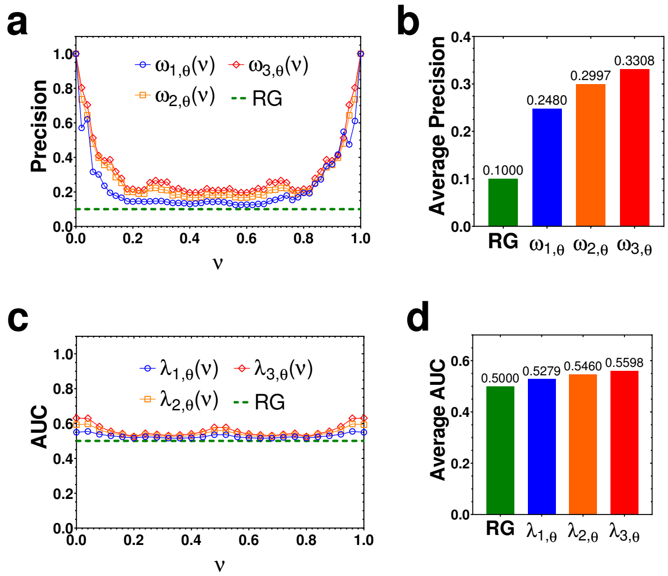

Fig. 2. The accuracy of n_1-D-Top as functions of ν in detecting rare previously unobserved labels (small n_1). The parameter settings are as follows: $n = 20$, $m = 100$, $\alpha = 0.1$ and $n_1 \in \{1, 2, 3\}$. RG represents the random guessing. Each result is averaged by over 100 independent implementations.

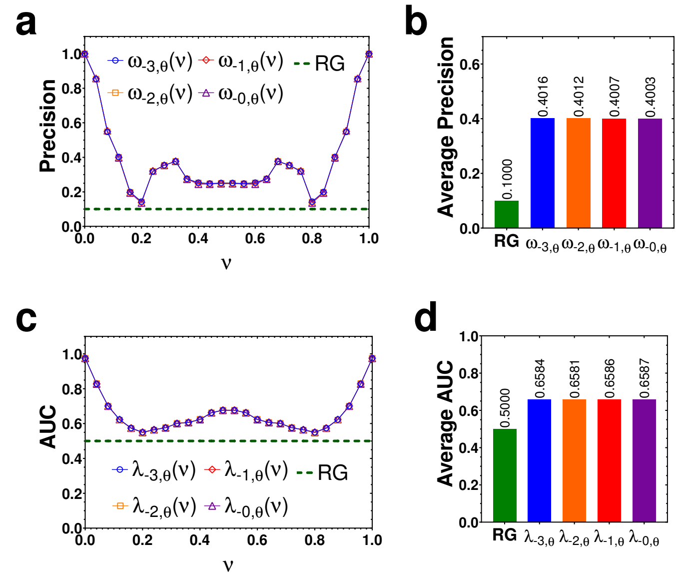

Fig. 3. The accuracy of n_1-D-Top as functions of ν in detecting rare previously unobserved labels (large n_1). The parameter settings are as follows: $n = 20$, $m = 100$, $\alpha = 0.1$ and $n_1 \in \{-3, -2, -1, -0\}$. RG represents the random guessing. Each result is averaged by over 100 independent implementations.

It follows from Eqs. (4)–(9) that

$$\omega_{n_1,\boldsymbol{\theta}} = \frac{1}{M}\sum_{i=1}^{M}\bar{\omega}_{n_1}(\mathbf{G}_i) \tag{10}$$

and

$$\lambda_{n_1,\boldsymbol{\theta}} = \frac{1}{M}\sum_{i=1}^{M}\bar{\lambda}_{n_1}(\mathbf{G}_i) \tag{11}$$

where $\mathbf{G}_i$ is a network randomly generated by the model with a specific parameter setting $\boldsymbol{\theta}$, for $i = 1, 2, \cdots, M$. By Eqs. (2)–(7), (10) and (11), we can readily investigate the Precision and AUC of n_1-D-Top applied to synthetic networks in practice.

VI. Experimental Results

Let $\mathbf{G} = (\mathcal{V}, \mathcal{E}, L)$ with $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$ and $|\mathcal{E}| = m$ be a network generated by the degradation-evolution model under the following configurations: (1) there are totally two edge labels, C_1 and C_2, which can be observed in $\mathbf{G}$; (2) label C_1 is the already-observed label and C_2 is the previously undiscovered one; (3) every edge has a unique edge label, and the percentage of edges with undiscovered label C_2 is α; and (4) node v_i's 1-attractiveness $A_{i,1}$ is $n - i$ and 2-attractiveness $A_{i,2}$ is $i - 1$, for $i = 1, 2, \cdots, n$. Let $\boldsymbol{\theta}$ represent these configurations. By the degradation-evolution network model, we obtain a family of networks $\{\mathbf{G}(t)\}_{-\infty}^{+\infty}$. For any t, every edge in $\mathbf{G}(t)$ has a unique edge label. According to the rewiring procedures introduced before, we have $\mathbf{G}(t)$ always consists of n nodes and m edges. In the following, we investigate the Precision and AUC of n_1-D-Top applied to $\mathbf{G}(t)$, for $-\infty < t < +\infty$. Note that the random guessing has a Precision of α and an AUC of 0.5 in this case.

We study the performance of n_1-D-Top with small n_1 in detecting rare unobserved labels. Figure 1 plots the Precision and the AUC of 1-D-Top, 2-D-Top and 3-D-Top as functions of ν when $\alpha = 0.1$. From Figure 2**a**, we find that when $\nu \in \{0, 1\}$ (*i.e.* $|t| = +\infty$), the algorithms always gain the perfect Precision (100%), which is consistent with our theoretical results. When ν is in the middle of the interval $[0, 1]$ (for instance, $\nu \in [0.2, 0.8]$), the Precision of each algorithm is stable, while ν approaches to 0 or 1, the Precision will increase sharply. From Figure 2**b**, we find that the average Precision of 1-D-Top, 2-D-Top and 3-D-Top (*i.e.* $\omega_{1,\boldsymbol{\theta}}$, $\omega_{2,\boldsymbol{\theta}}$ and $\omega_{3,\boldsymbol{\theta}}$) are 0.2480, 0.2997 and 0.3308, respectively. We conclude that 1-D-Top, 2-D-Top and 3-D-Top are valid and effective, since they all outperform random guessing, and improve the Precision of random guessing (10%) by 148.0%, 199.7% and 230.8%, respectively. Moreover, as shown in Figure 2b that the Precision of n_1-D-Top increases as n_1 increases, showing that detection based on more edges with observed labels would get better performance. From Figure 2**c** and Figure 2**d**, we see that the average AUC of 1-D-Top, 2-D-Top and 3-D-Top (*i.e.* $\lambda_{1,\boldsymbol{\theta}}$, $\lambda_{2,\boldsymbol{\theta}}$ and $\lambda_{3,\boldsymbol{\theta}}$) are close to 0.5 showing that n_1-D-Top has a poor performance in AUC when n_1 is small, which is consistent with our previous judgment (see Section IV).

We consider the performance of n_1-D-Top with large n_1 in detecting rare unobserved labels. We use notation $-n_1$-D-Top to represent $(m_1 - n_1)$-D-Top, where m_1 denote the total number of edges with label C_1. For example, -0-D-Top refers to the proposed detection method based on all the edges with C_1. Figure 3 plots the Precision and the AUC of $-n_1$-D-Top as functions of ν in the case of $\alpha = 0.1$ for $n_1 \in \{0, 1, 2, 3\}$. From Figures 3**a** and 3**c**, we find the four detection algorithms have almost the same accuracy, indicating that they have almost achieved the upper bound of the proposed method's performance. Figures 3**b** and 3**d** demonstrate that the proposed method can improve the Precision and AUC of random guessing by $\geq 300\%$ and $\geq 31\%$, respectively.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2022.3174086, IEEE Internet of Things Journal

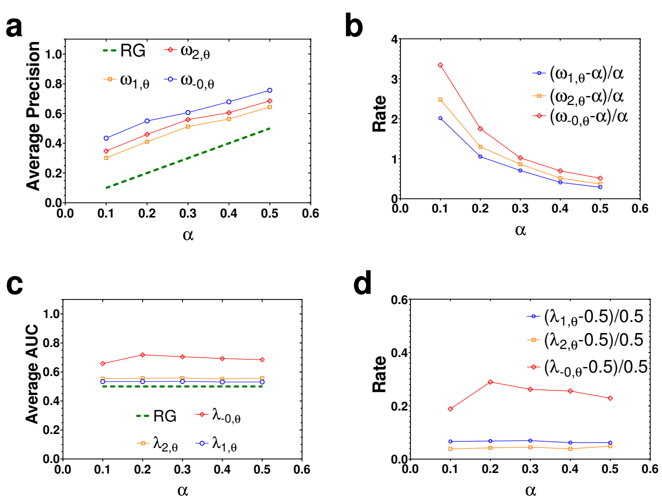

Fig. 4. The average accuracy of n_1-D-Top as functions of α in detecting previously unobserved labels. The parameter settings are as follows: $n = 20$, $m = 100$, $\alpha \in \{0.1, 0.2, \cdots, 0.5\}$ and $n_1 \in \{-0, 1, 2\}$. RG represents the random guessing. Each result is averaged by over 100 independent implementations.

When the unobserved label is not rare, we show the performance of n_1-D-Top in Figure 4. As shown above, 1-D-Top and -0-D-Top are the algorithms with the worst performance and best performance, and their performance outlines the feasible region of the accuracy of the proposed method. It can be seen from Figures 4**a** and 4**c** that as the rarity of the unobserved label (α) increases, the Precision of our method also increases linearly, while the AUC usually holds steady. Interestingly, from Figure 4**b**, we find that the rarer the unobserved label is, the more largely our method improves the Precision of random guessing.

VII. Conclusions

In this paper, we propose a PUEL detection approach aiming to find out a small set of edges with previously undiscovered edge labels. This approach is of significance since its solutions would benefit researchers in mining new features of a wide range of datasets, for instance, to discover new variants of COVID-19. Although tremendous effort has been put into interaction type detection, the PUIT detection still remains unsolved. In this paper, we strive to take a first step to overcome the proposed unsolved problem by answering the following fundamental question: is there any effective detection method other than the random guessing method? Our idea is to find out some networks in which non-trivial detection method exists. Specifically, we propose a temporal directed network model and develop an effective detection method for synthetic networks generated by the model. We focus on one of the most straightforward cases of the target problem: detecting the unobserved label in networks in which there is an already-observed label and an previously undiscovered one. Applying our method to detect the PUEL in a number of small synthetic networks, we find that our detection method is effective and has much better performance than the baseline method.

In this paper, we study a universal network model which can be used to mimic a wide range of real-world directed networks, and derive a universal PUEL detection method under this network model along an unconventional manner. Thus, the obtained method can be directly applied to any real-world directed networks. But the performance in practice may be not as good as in our simulation results since the more general the method is, the less effective for a given scheme may be. For a given real-world system, we can modify the network model to fully capture the features of the system, and then derive a much more effective method along the same line adopted in this paper. However, the main purpose of this paper is to give an answer to the fundamental question theoretically: whether the proposed PUEL detection problem is solvable by non-trivial methods (methods other than random guessing). So the performance of the proposed method is enough for our goal. We leave the research on methods for detecting PUELs in real-world systems such as virus spreading networks, citation networks, sales networks, and so on, as open problems which are also our future works.

Appendix A: Proof of Lemma 1

In the limit $t \to +\infty$ (resp. $-\infty$), we have

$$P(\mathbf{G}, t) = P_{max}(\mathbf{G}) \quad \text{(resp.} \quad P_{min}(\mathbf{G})), \qquad (12)$$

where $\mathbf{G} = \mathbf{G}(t)$, $P_{max}(\mathbf{G})$ and $P_{min}(\mathbf{G})$ denote the supremum and infimum of $P(\mathbf{G}, t)$, respectively. Assume that all the edges in E lie in $\mathbf{G}_l(t)$. Note that all the nodes in $\lceil E \rceil_{tar}$ have different l-attractiveness. Without loss of generality, let v_1 (resp. v_2) denote the node with the largest (resp. smallest) l-attractiveness in $\lceil E \rceil_{tar}$. In the following, we prove that v_1 (resp. v_2) has no $(2, \mathbf{G}(t))$-follower in $\lceil E \rceil_{src}$ when $t = +\infty$ (resp. $t = -\infty$) by contradiction. Assume $v_j \in \lceil E \rceil_{src}$ is a $(2, \mathbf{G}(t))$-follower of v_1 (resp. v_2). Obviously, we have $v_j \not\to v_1$ (resp. $v_j \not\to v_2$). By $v_j \in \lceil E \rceil_{src}$, there is $v_k \in \lceil E \rceil_{tar}$ such that $k \neq 1$ (resp. $k \neq 2$) and $v_j \to v_k \in E$. Then we have $C_l \in \mathcal{C}(j, k, t)$ and $A_{k,l} < A_{1,l}$ (resp. $A_{k,l} > A_{2,l}$). In $\mathbf{G}_l(t)$, change $v_j \to v_k$ to $v_j \to v_1$ and let t increase (resp. decrease) by 1. By Eq. (1), we have $P_l(j, t+1) - P_l(j, t) = A_{1,l} - A_{k,l} > 0$ (resp. $P_l(j, t-1) - P_l(j, t) = A_{2,l} - A_{k,l} < 0$) Then one has $P(\mathbf{G}, t) < P_{max}(\mathbf{G})$ (resp. $P(\mathbf{G}, t) > P_{min}(\mathbf{G})$) which contradicts Eq. (12). Finally, we conclude that there always exists a node in $\lceil E \rceil_{tar}$ having no $(2, \mathbf{G}(t))$-follower in $\lceil E \rceil_{src}$ when $t = +\infty$ (resp. $t = -\infty$).

Appendix B: Proof of Lemma 2

We have $E_1, E_2, \cdots, E_n$ are pairwise disjoint non-empty subsets of E, and $\delta(E)$ is the remaining set. Then, we have $\cup_{i=1}^{n+1} E_i = E$. We show that $|\delta(E)| \neq 1$ by the reverse proving. Assume $|\delta(E)| = 1$. Without loss of generality, we assume $\delta(E) = \{v_1 \to v_2\}$. Then $\lceil \delta(E) \rceil_{src} = \{v_1\}$ and $\lceil \delta(E) \rceil_{tar} = \{v_2\}$. Note that v_2 has no $(2, \mathbf{G})$-follower in $\lceil \delta(E) \rceil_{src}$. According to the removal operations, $v_1 \to v_2$ can be removed. Thus, we have $|\delta(E)| \neq 1$. In the following, we construct a network $\mathbf{G} = (\mathcal{V}, \mathcal{E})$ and a set $E \subseteq \mathcal{E}$ with $|\delta(E)| = 2$. Let $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$ and $\mathcal{E} = \{v_1 \to v_2, v_1 \to$

$v_3, v_3 \to v_1, v_3 \to v_4\}$. Let $E = \{v_1 \to v_2, v_3 \to v_4\}$. We have $\lceil E \rceil_{src} = \{v_1, v_3\}$ and $\lceil E \rceil_{tar} = \{v_2, v_4\}$. Note that v_1 is a $(2, \mathbf{G})$-follower of v_4 and v_3 is a $(2, \mathbf{G})$-follower of v_2. According to the removal operations, we have $\delta(E) = E$ and $|\delta(E)| = |E| = 2$. To sum up, we have if $\delta(E) \neq \emptyset$, then $|\delta(E)| \geq 2$.

Appendix C: Proof of Lemma 3

Let $\mathbf{G} = \mathbf{G}(t)$, $k \in \{1, 2, \cdots, s\}$ and $E_k^{(1)}$ denote the set consisting of all the edges removed from E in the k-th removal operation, and $\delta^{(1)}(E)$ be the remaining set. According to Lemma 2 (3), we have

$$\delta^{(1)}(E) \cup (\cup_{k=1}^{s} E_k^{(1)}) = E. \quad (13)$$

Case 1: $|\delta^{(2)}(E)| > 0$. First, we prove $\delta^{(2)}(E) \subseteq \delta^{(1)}(E)$ by contradiction. Assume $\delta^{(2)}(E) \nsubseteq \delta^{(1)}(E)$. It follows Eq. (13) that $\delta^{(2)}(E) \cap (\cup_{k=1}^{s} E_k^{(1)}) \neq \emptyset$. Let l be the smallest integer, such that $\delta^{(2)}(E) \cap E_l^{(1)} \neq \emptyset$. Then we have

$$\delta^{(2)}(E) \subseteq \delta^{(1)}(E) \cup (\cup_{k=1}^{s} E_k^{(1)})$$

and

$$\lceil \delta^{(2)}(E) \rceil_{src} \subseteq \lceil \delta^{(1)}(E) \cup (\cup_{k=l}^{s} E_k^{(1)}) \rceil_{src}. \quad (14)$$

Let $v_i \to v_j \in \delta^{(2)}(E) \cap E_l^{(1)}$. Note that $v_j \in \lceil \delta^{(2)}(E) \rceil_{tar}$. According to the definition of Delta-property, we have v_j has at least one $(2, \mathbf{G})$-follower in $\lceil \delta^{(2)}(E) \rceil_{src}$. By Eq. (14) we have v_j has $(2, \mathbf{G})$-followers in $\lceil \delta^{(1)}(E) \cup (\cup_{k=l}^{s} E_k^{(1)}) \rceil_{src}$. By $v_j \in \lceil E_l^{(1)} \rceil_{tar}$, we know that v_j is removed in the l-th removal operation. Then according to the removal procedures (1)-(2) of the Delta-property, we have v_j should have no $(2, \mathbf{G})$-follower in $\lceil \delta^{(1)}(E) \cup (\cup_{t=l}^{s} E_t^{(1)}) \rceil_{src}$. This leads to conflict. Therefore, $\delta^{(2)}(E) \subseteq \delta^{(1)}(E)$. Note that $|\delta^{(1)}(E)| \geq |\delta^{(2)}(E)| > 0$. Then, we obtain $\delta^{(1)}(E) \subseteq \delta^{(2)}(E)$ in the same way. Consequently, we have $\delta^{(1)}(E) = \delta^{(2)}(E)$.

Case 2: $|\delta^{(2)}(E)| = 0$. We prove $|\delta^{(1)}(E)| = 0$ by contradiction. Assume $|\delta^{(1)}(E)| > 0$. According to Case 1, we have $\delta^{(1)}(E) \subseteq \delta^{(2)}(E)$. Thus, $|\delta^{(2)}(E)| \geq |\delta^{(1)}(E)| > 0$, which contradicts the assumption that $|\delta^{(2)}(E)| = 0$. Consequently, we have $|\delta^{(1)}(E)| = 0$. Finally, we obtain $\delta^{(1)}(E) = \delta^{(2)}(E) = \emptyset$.

Appendix D: Proof of Theorem 1

Let $\mathbf{G} = \mathbf{G}(t)$ and $|t| = +\infty$. We prove Theorem 1 by showing that if all the edges in E share a common label, then $\Delta(E) = 1$. Let $E(0) = E$. According to Lemma 1, there exists a node v_{i_1} in $\lceil E(0) \rceil_{tar}$ which has no $(2, \mathbf{G})$-follower in $\lceil E(0) \rceil_{src}$. Let $E(1) = E(0) \setminus \{e \in E(0) | v_{i_1} = \lceil e \rceil_{tar}\}$. Obviously, all the edges in $E(1)$ lie in the same layer of $\mathbf{G}$. Then by Lemma 1 again, we obtain v_{i_2} in $\lceil E(1) \rceil_{tar}$ which has no $(2, \mathbf{G})$-follower in $\lceil E(1) \rceil_{src}$. Let $E(2) = E(1) \setminus \{e \in E(1) | v_{i_2} = \lceil e \rceil_{tar}\}$. Repeat this removal operation on E until all the edges in E are removed. Finally, we have $\delta(E) = \emptyset$ and $\Delta(E) = 1$.

References

[1] M. Brbić, M. Zitnik, S. Wang, A. O. Pisco, R. B. Altman, S. Darmanis, and J. Leskovec, "Mars: discovering novel cell types across heterogeneous single-cell experiments," *Nature Methods*, vol. 17, no. 12, pp. 1200–1206, 2020.

[2] S. Wang, A. O. Pisco, A. McGeever, M. Brbic, M. Zitnik, S. Darmanis, J. Leskovec, J. Karkanias, and R. B. Altman, "Leveraging the cell ontology to classify unseen cell types," *Nature Communications*, vol. 12, no. 1, pp. 1–11, 2021.

[3] P. Holme and M. Huss, "Role-similarity based functional prediction in networked systems: application to the yeast proteome," *Journal of the Royal Society Interface*, vol. 2, no. 4, pp. 327–333, 2005.

[4] L. Daqing, K. Kosmidis, A. Bunde, and S. Havlin, "Dimension of spatially embedded networks," *Nature Physics*, vol. 7, no. 6, pp. 481–484, 2011.

[5] L. Daqing, J. Yinan, K. Rui, and S. Havlin, "Spatial correlation analysis of cascading failures: congestions and blackouts," *Scientific reports*, vol. 4, no. 1, pp. 1–6, 2014.

[6] F. Xu, Y. Li, D. Jin, J. Lu, and C. Song, "Emergence of urban growth patterns from human mobility behavior," *Nature Computational Science*, vol. 1, no. 12, pp. 791–800, 2021.

[7] J. Jung, S. Chun, X. Jin, and K.-H. Lee, "Quantitative computation of social strength in social internet of things," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 4066–4075, 2018.

[8] M. Nitti, L. Atzori, and I. P. Cvijikj, "Friendship selection in the social internet of things: Challenges and possible strategies," *IEEE Internet of Things Journal*, vol. 2, no. 3, pp. 240–247, 2015.

[9] X. Bi, T. Qiu, W. Qu, L. Zhao, X. Zhou, and D. O. Wu, "Dynamically transient social community detection for mobile social networks," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1282–1293, 2021.

[10] A. M. Ortiz, D. Hussein, S. Park, S. N. Han, and N. Crespi, "The cluster between internet of things and social networks: Review and research challenges," *IEEE Internet of Things Journal*, vol. 1, no. 3, pp. 206–215, 2014.

[11] V. Gemmetto, T. Squartini, F. Picciolo, F. Ruzzenenti, and D. Garlaschelli, "Multiplexity and multireciprocity in directed multiplexes," *Physical Review E*, vol. 94, no. 4, p. 042316, 2016.

[12] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, and G. Petri, "Networks beyond pairwise interactions: structure and dynamics," *Physics Reports*, vol. 874, pp. 1–92, 2020.

[13] S. Boccaletti, G. Bianconi, R. Criado, C. I. Del Genio, J. Gómez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin, "The structure and dynamics of multilayer networks," *Physics reports*, vol. 544, no. 1, pp. 1–122, 2014.

[14] D. Vasireddy, R. Vanaparthy, G. Mohan, S. V. Malayala, and P. Atluri, "Review of covid-19 variants and covid-19 vaccine efficacy: what the clinician should know?," *Journal of Clinical Medicine Research*, vol. 13, no. 6, p. 317, 2021.

[15] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, no. 7307, pp. 761–764, 2010.

[16] A. Clauset, C. Moore, and M. E. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, 2008.

[17] D. Hric, T. P. Peixoto, and S. Fortunato, "Network structure, metadata, and the prediction of missing nodes and annotations," *Physical Review X*, vol. 6, no. 3, p. 031038, 2016.

[18] Z.-M. Ren, A. Zeng, and Y.-C. Zhang, "Structure-oriented prediction in complex networks," *Physics Reports*, vol. 750, pp. 1–51, 2018.

[19] T. Martin, B. Ball, and M. E. Newman, "Structural inference for uncertain networks," *Physical Review E*, vol. 93, no. 1, p. 012306, 2016.

[20] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1361–1370, 2010.

[21] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th International Conference on World Wide Web*, pp. 641–650, 2010.

[22] D. Song and D. A. Meyer, "Link sign prediction and ranking in signed directed social networks," *Social Network Analysis and Mining*, vol. 5, no. 1, pp. 1–14, 2015.

[23] W. Yuan, C. Li, G. Han, D. Guan, L. Zhou, and K. He, "Negative sign prediction for signed social networks," *Future Generation Computer Systems*, vol. 93, pp. 962–970, 2019.

[24] A. Khodadadi and M. Jalili, "Sign prediction in social networks based on tendency rate of equivalent micro-structures," *Neurocomputing*, vol. 257, pp. 175–184, 2017.

[25] M. Shahriari, O. A. Sichani, J. Gharibshah, and M. Jalili, "Sign prediction in social networks based on users reputation and optimism," *Social Network Analysis and Mining*, vol. 6, no. 1, pp. 1–16, 2016.

[26] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin, "Structure of growing networks with preferential linking," *Physical Review Letters*, vol. 85, no. 21, p. 4633, 2000.

[27] X.-J. Xu, X.-M. Hu, and L.-J. Zhang, "Network evolution by nonlinear preferential rewiring of edges," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 12, pp. 2429–2434, 2011.

[28] S. Ree, "Generation of scale-free networks using a simple preferential-rewiring dynamics," *Physica A: Statistical Mechanics and its Applications*, vol. 376, pp. 692–698, 2007.

[29] S. Johnson, J. J. Torres, and J. Marro, "Nonlinear preferential rewiring in fixed-size networks as a diffusion process," *Physical Review E*, vol. 79, no. 5, p. 050104, 2009.

[30] J. Ohkubo, K. Tanaka, and T. Horiguchi, "Generation of complex bipartite graphs by using a preferential rewiring process," *Physical Review E*, vol. 72, no. 3, p. 036120, 2005.

[31] F. Papadopoulos, M. Kitsak, M. Á. Serrano, M. Boguná, and D. Krioukov, "Popularity versus similarity in growing networks," *Nature*, vol. 489, no. 7417, pp. 537–540, 2012.

[32] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.

[33] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (roc) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[34] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: statistical mechanics and its applications*, vol. 390, no. 6, pp. 1150–1170, 2011.