Influence Maximization in Multi-Agent Systems by A Graph Embedding Method: Dealing with Probabilistically Unstable Links

Mincan Li, Zidong Wang, *Fellow, IEEE*, Qing-Long Han, *Fellow, IEEE*, Simon J. E. Taylor, Kenli Li, Xiangke Liao, and Xiaohui Liu

***Abstract*—This paper is concerned with the influence maximization problem under a network with probabilistically unstable links (PULs) via graph embedding for multi-agent systems (MASs). First, two diffusion models, the unstable-link independent cascade (UIC) model and the unstable-link linear threshold (ULT) model, are designed for the influence maximization problem under the network with PULs. Second, the MAS model for the influence maximization problem with PULs is established and a series of interaction rules among agents are built for the MAS model. Third, the similarity of the unstable structure of the nodes is defined and a novel graph embedding method, termed the unstable-similarity2vec (US2vec) approach, is proposed to tackle the influence maximization problem under the network with PULs. According to the embedding results of the US2vec approach, the seed set is figured out by the developed algorithm. Finally, extensive experiments are conducted to 1) verify the validity of the proposed model and the developed algorithms, and 2) illustrate the optimal solution for influence maximization under different scenarios with PULs.**

***Index Terms*—Influence maximization, graph embedding, multi-agent systems, probabilistically unstable links.**

I. Introduction

For a long time, people obtain information from their social circles in order to make decisions on their choices of commodity, transportation means, restaurants, cinemas, etc. Accordingly, the theory of social systems [29], [37], [46] has been proposed to study humans' behaviors and social activities, where it has been hypothesized that the minority plays a significant role in public opinion building [15]. Following this hypothesis, influence diffusion models have been developed to facilitate quantitative research as well as simulation of human social activities [18], [20]. Based on the diffusion models, the topic of influence maximization (IM) has gained an ever-increasing popularity in a variety of application areas such as location technology [13], big data [4] and community identification [43]. Influence maximization has been regarded as an appropriate way to simulate human activities and realistic scenarios within a social network. For example, an online influence maximization model with non-adaptive and adaptive advert sequencing has been proposed to maximize the viral marketing in social works [41].

Owing to the advance in the graph theory and the accessibility of the big database, much progress has been made recently on the social science especially the analysis and mining of social networks. Basically, the feedback loops, instabilities and cascade effects have to be taken into simultaneous consideration in simulating the complex human activities in networks [6], [45], and this poses great challenges on the implementation of such simulations. Multi-agent systems (MASs) are thought to be an effective way of solve this problem as they are capable of adapting to various conditions/constraints [3], [8]. In MASs, the special characteristics of agents (e.g. sociality, self-organization and coordination) play significant roles in improving simulation effects. More specifically, influence diffusion needs to be represented directly and in real-time by agent communications in order to make opinion-forming processes clear. So far, many researchers have been devoted to the investigation of combining influence maximization and MAS models in order to take advantage of both of them. For instance, an effective agent-based algorithm has been proposed to control the opinion formation in complex networks [1]. In addition, based on MASs, the analysis of opinion formation has been accomplished by agents in a social network.

Despite the advantages and convenience brought by MASs in modelling influence maximization, diffusion effectiveness has been limited to three main methods, i.e. the approximation algorithm, the parameterized complexity theory and the diffusion approach. Simulating a large-scale network consisting of agents with complicated links and interaction may lead to excessively long run times. Approaches to graph embedding algorithms have been proposed as way of improving computational efficiency and have drawn considerable research interest from both academic and industrial communities [28], [31].

This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB2104000, the National Natural Science Foundation of China under Grants 61625202, 61751204, 61860206011 and 62206091, the European Union's Horizon 2020 Research and Innovation Programme under Grant 820776 (INTEGRADDE), the Royal Society of the UK, and the Alexander von Humboldt Foundation of Germany.
(Corresponding author: Kenli Li.)

Mincan Li and Kenli Li are with the College of Computer Science and Electronic Engineering, Hunan University, Changsha, Hunan 410082, China; and also with the National Supercomputing Center in Changsha, Changsha, Hunan 410082, China. (emails: limc@hnu.edu.cn; lkl@hnu.edu.cn)

Zidong Wang, Simon J. E. Taylor and Xiaohui Liu are with the Department of Computer Science, Brunel University London, Uxbridge, Middlesex, UB8 3PH, United Kingdom. (emails: Zidong.Wang@brunel.ac.uk; Simon.Taylor@brunel.ac.uk; Xiaohui.Liu@brunel.ac.uk)

Qing-Long Han is with the School of Science, Computing and Engineering Technologies, Swinburne University of Technology, Melbourne, VIC 3122, Australia. (email: qhan@swin.edu.au)

Xiangke Liao is with the Collaborative Innovation Center of High Performance Computing, National University of Defense Technology, Changsha 410073, China. (email: xkliao@nudt.edu.cn).

Generally, graph embedding approaches fall into three main categories: factorization methods [7], [39], random walks [9], [38], and deep learning [14]. Factorization methods typically take a long time to run, for instance, the Spectral Clustering of GE had the time complexity $O(n^3)$. The time complexity of word2vec-based random walks are usually $O(\log n)$. As a general algorithm, the adversarial graph embedding method has been proposed to discern sensitive attributes reasonably while keeping their competitiveness for every node [19]. By using the clustering in graph embedding algorithms, the structure-based and attribute-based clustering methods have been successfully applied to many applications, and this motivates us to apply the structure-based clustering approach to the topologies of MASs in the influence maximization problem.

Up till now, existing research has been dedicated to solving the influence maximization problem under *fixed* topologies or *stable* links, and little effort has been made on influence maximization problems with *probabilistically unstable links* (PULs) that are frequently encountered in practice. The lack of corresponding results is mainly due to the fact that, in case of PULs, there would be a great deal of uncertainties on the network topology that pose substantial challenges to the influence maximization. Generally speaking, the phenomenon of the PULs can be classified into three types, i.e. intermittent links [21], probabilistic links [11], [34] and dynamic links [24]. Each type results in its distinctive uncertainty on the influence diffusion, and such uncertainties complicate the community structure and bring in substantial difficulties in influences calculation. Clearly, the ignorance of PULs in existing results would inevitably lead to information missing during the process of complicated influence diffusion.

Motivated by the above discussions, the main aim of this paper is to design a graph-embedding-based method to solve the influence maximization problem in the MASs with PULs. This problem appears to be especially difficult as we are inevitably confronted with the following three essential challenges: 1) how to design the rules of influence calculation on different diffusion models undergoing PULs? 2) how to combine influence diffusion with MASs' characteristics exhibiting agents' sociality, self-organizing and coordination? and 3) how to design a novel graph embedding algorithm to tackle influence maximization in large-scale networks with PULs? It is, therefore, the main purpose of this paper to address the challenges.

This paper is concerned with the influence maximization problem in MASs under the network that has PULs through a novel graph embedding method. The main contributions of this paper are highlighted as follows.

1) Two novel diffusion models are designed for MAS simulation on the influence maximization problem in social network with PULs.
2) A new series of communication rules of agents are proposed to optimize influence spreading in MASs with PULs, which makes the computation of the graph-embedding method more efficient than in previous work.
3) A novel graph embedding method, i.e. the unstable-similarity2vec (US2vec) algorithm, is proposed for dealing with uncertain links where neighborhood information and ability evaluations of agents are simultaneously taken into account, thus making diffusion prediction more precise than previous research.
4) A purposely selected method for seed set is put forward, which takes into account the diffusion capability, the information updating and the predictions and obtains a lager diffusion range on IM than before.

The rest of this paper is organized as follows. Section II discusses the related work of influence maximization algorithms and their applications especially with MASs and existing graph embedding approaches. The problem formulation is given in Section III. In Section IV, the models and the proposed algorithms for the influence maximization problem of MASs are provided. Section V presents the experiment results, related analysis, and the comparisons with other state-of-the-art approaches. Conclusions are drawn in Section VI with a discussion of future research topics.

II. Related Work

The influence maximization problem has been proved to be NP-hard in [10] and applied to a wide range of scenarios. In recent years, several methodologies have been proposed to find a seed set for achieving near-optimal influence spread in influence maximization problems where approximation has been shown to be extremely effective [2], [12], [55]. Using approximation, the influence maximization problem has often been treated as a combinatorial optimization problem where the solutions (at the approximation step) gradually enter into near-optimal asymptotic bounds. For example, in [56], the semi-definite-based algorithm has been designed to keep the approximation ratio higher than 1-$1/e$ if the ratio of the seeds to the total number of nodes resides in a certain range. It should be noted that, although approximation algorithms have been proposed to generate solutions that are close to near-optimal asymptotic bounds, most of these algorithms have suffered from the scalability issues. As network size (or links) increase, computation time becomes untenable. It is worth noting that these algorithms have all been implemented with certain and stable networks. As such, it remains unclear whether these existing algorithms are still applicable within an uncertain and unstable environment [26].

Compared with approximation approaches, heuristic methods have advantages in run time and scalability [22], [48]. For example, a K-shell decomposition-based heuristic algorithm called KDBH has been proposed in [33], which has proven to be a quick-finding approach on the most influential spreaders. It is noted that heuristic algorithms have not provided a worst-case bound on influence spread. As for community-based solution approaches [36], [47], the community detection of the underlying social network has been introduced as an intermediate step to improve scalability. For example, the K-ECC model has been leveraged in [51] to measure the cohesiveness of subgraphs and find the influential community. It is worth mentioning that all the above-mentioned methods cannot deal with the rapid growth and the dynamic changing of the large-scale network, whereas the graph embedding can be regarded as an appropriate way of tackling scalability.

Graph embedding methods can be categorized into factorization based, random walk based, and deep learning based schemes. Basic graph embedding approaches, such as large-scale information network embedding (LINE) [40], high-order proximity preserved embedding (HOPE) [30] and structural deep network embedding (SDNE) [44], have been proposed to accommodate the sparsity of huge real-world networks. Structure preserving embedding (SPE) [35] (as a factorization-based graph embedding approach) has been proposed to reconstruct the input graph exactly. According to these methods, the structure information of networks can be embedded into a low-dimensional space. With the structure information in the low dimensional space, graph embedding can easily tackle huge and complicated networks. The attributed node to vector (AN2VEC) method [23] attempts to disentangle information shared by the structure of a network and its nodes' features. A manifold graph embedding method [50] has been developed to solve community discovery problems through a structure information propagation mechanism.

III. Problem Formulation

In this section, the PUL-based influence maximization problem will be defined in a mathematically rigorous way.

The population of the PUL-based influence maximization problem can be abstract as a graph, in which nodes represent human or other entities and edges denotes the relationships among the population. Denote $G \triangleq (V, E, P)$ as a 3-tuple network where V is the set of nodes, E is the set of edges and P represents the set of probabilities of edges. Here, $P \triangleq \{\{P_1\}, \{P_2\}\}$ is a 2-tuple set where P_1 denotes the success probabilities of the influence diffusion on edges in E and P_2 denotes the stability probabilities of the edges (i.e., the probabilities for the edges to be stable).

Unstable Link: Every edge is a link and has a stability parameter p_2 (which is included in P_2 of P). The value of p_2 denotes the probability of successful connection of the linkage. A link is said to be unstable if $p_2 \in (0, 1)$, and is said to be connected (stable or available) if $p_2 = 1$. To be specific, when the influence is spread on a network with PULs, the availabilities of unstable links are uncertain because of $p_2 \in (0, 1)$ while the stable links are always available because of $p_2 = 1$. If the links are available, the success of influence spreading is decided by a fixed probability which is p_1 ($p_1 \in P_1$). At the same time, the availability of any unstable link is decided by the two linked agents and the value of p_2 for this link. When the influence is spreading on an unstable link, two linking probabilities are given by two agents, respectively, which are linked by the unstable link. As long as the two linking probabilities are both larger than p_2, the unstable link is considered to be available. The details of this process will be illustrated in Section IV-A.

In order to deal with influence maximization with PULs, the structural similarity with instability can be defined as follows:

Structural Similarity:

$$\begin{aligned} f_q(u, v) =& f_{q-1}(u, v) + g(a(R_q(u)), a(R_q(v))) \\ &+ g(b(R'_q(u)), b(R'_q(v))), \\ &q \geq 0, |R_q(u)|, |R_q(v)| > 0 \end{aligned} \tag{1}$$

where *u* and *v* are two nodes in the network *G*. $f_q(u, v)$ means the similarity of *u* and *v* within the distance of *q* (hop count) of themselves. $R_q(u)$ denotes a set of nodes whose minimal distance is *q* from node *u*. $R'_q(u)$ denotes a set of nodes whose minimal distance is *q* from node *u* only through PULs. It should be notice that, if $R'_q(u) = \varnothing$ then let $b(R'_q(u)) = \{0\}$. $a(R_q(u))$ and $b(R_q(u))$ represent an ordered degree sequence of the nodes in $R_q(u)$ and $R'_q(u)$, respectively. $g(a(R_q(u)), a(R_q(v)))$ calculates the distance between the two ordered degree sequences using the modified Dynamic Time Warping method [16], [57]. It should be noticed that $q \geq 0$. Thus, when $q = 0$, we let $f_{-1} = 0$.

Remark 1: We use Fig. 1 as a demonstration of the structural similarity, where a_1 and a_{66} are two nodes that are far away from each other in the same network. In their distance of $q = 1$, they are structurally similar (including the similarity in instability): 5 degrees marked by blue lines, 2 PULs marked by blue and dotted lines, and 2 triangular structures ($\triangle a_1 a_2 a_{10}$ and $\triangle a_1 a_4 a_7$ to a_1; $\triangle a_{66} a_{63} a_{64}$ and $\triangle a_{66} a_{67} a_{68}$ to a_{66}). Similar situations apply to a_9 and a_{68} whose degrees are marked by red lines. Eq. (1) can be applied to the two pairs of node (a_1, a_{66} and a_9, a_{68}) to measure their similarities. The structural similarity of a_1 and a_{66} in distance $q = 1$ is:

$$\begin{aligned} f_1(a_1, a_{66}) &= f_0(a_1, a_{66}) + g(a(R_1(a_1)), a(R_1(a_{66}))) \\ &\quad + g(b(R'_1(a_1)), b(R'_1(a_{66}))) \\ &= 0 + g(a\{a_2, a_3, a_4, a_5, a_{10}\}, a\{a_{63}, a_{64}, a_{65}, a_{67}, a_{68}\}) \\ &\quad + g(b\{a_5, a_7\}, b\{a_{65}, a_{68}\}) \\ &= g((5, 5, 4, 3, 3, 2), (4, 4, 4, 3, 3, 3)) \\ &\quad + g((2, 1), (1, 1)) \\ &= 1 + 1 = 2. \end{aligned}$$

The structural similarity of a_9 and a_{68} can be obtained by following the same calculation procedure with the result of $f_1(a_9, a_{68}) = 1$.

Influence Diffusion: The process of influence diffusion relies on the diffusion model. According to the rule of the diffusion model, inactive nodes can be activated by some of their active neighbor nodes at the current time $t = t_0$ and such an activation could be successful or unsuccessful (with probabilities in P_1). Such a diffusion step (at $t = t_0$) is then repeated at the time $t = t_0 + 1$, and this kind of repetition is called an iteration of influence diffusion. If there is eventually no node that can be activated, the iteration stops, which means that the influence diffusion is finished. It should be noticed that the reconnection of every unstable link takes place only once at every iteration of the influence diffusion.

Influence Maximization: Given a network $G \triangleq (V, E, P)$ and a positive integer *k*, find a node set *S* consisting of *k* nodes (also called seed nodes) from *V*, such that the expected number of activated nodes during the spreading process under the diffusion model (which is also termed as influence diffusion) is maximized.

Two diffusion models for influence maximization problem with PULs have been proposed as follows.

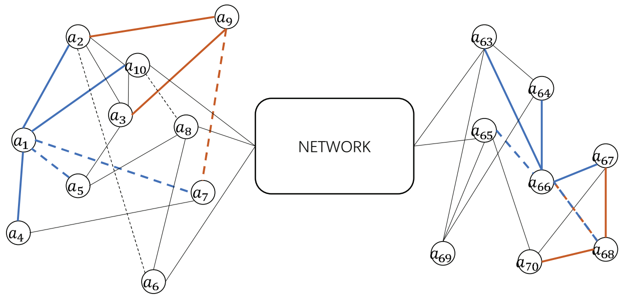


Fig. 1. Examples of the Structural Similarity with PULs

Unstable-link Independent Cascade (UIC) Model: The independent cascade (IC) model is one of the basic diffusion models [32]. The newly activated nodes are required by the IC model to influence their neighbors according to the corresponding activating probabilities of the edges in every iteration. The main idea of this diffusion remains within the UIC model, where the diffusion probability on an unstable link is defined by the UIC model as:

$$p_1'(u,v) = p_1(u,v)\sigma^n, \ \sigma \in (0,1) \tag{2}$$

Because of the PULs' continuous connections, the diffusion probabilities on these links decrease by the attenuated parameter σ. *n* denotes the number of reconnections of the edge (u,v).

Unstable-link Linear Threshold (ULT) Model: The linear threshold (LT) model is a popular and general diffusion model [17]. In the LT model, every node has its activated threshold and different influence value on its neighbors. As long as the accumulated influence value exceeds the threshold of an individual node, it will be activated. In particular, in the ULT model, PULs always have continuous connections. On one hand, the influence diffusion is attenuated by numerous re-connections; on the other hand, the influence value diffused on the unstable link may not be positive.

The influence value diffused on an unstable link can be calculated in the ULT model as follows:

$$b'_{uv} = \begin{cases} -\delta^i b_{uv}, \ b_{uv} = p_1(u,v) & \gamma \le \gamma_v \\ \delta^i b_{uv}, \ b_{uv} = p_1(u,v) & \gamma > \gamma_v \end{cases} \tag{3}$$

where b'_{uv} denotes the current influence value diffused on node *v* by node *u* through unstable link (u,v), and b_{uv} indicates the influence value for *u* to influence on *v* by their first-time linking. γ_v indicates the link (u,v)'s probability of the positive influence during the diffusion. If the probability γ is larger than γ_v, the influence value will be positive. *i* represents the number of re-connections and δ indicates the attenuated parameter of the link (u,v).

According to Eq. (3), the calculation of the accumulation of influence values on node *v* can be designed as:

$$\sum_{u\in N(v)} b_{uv} = \sum_{u'\in S(v)} b_{u'v} + \sum_{u''\in US(v)} b_{u''v} \tag{4}$$

where $N(v)$ denotes the set of the nodes which have made influences on node *v*, $S(v)$ represents the set of nodes in $N(v)$ which have stable link (u',v), $US(v)$ indicates the set of the nodes in $N(v)$ which have unstable link (u'',v).

Some important notations used in this paper are listed in Table I.

TABLE I
MATHEMATICAL NOTATIONS

Symbol	Description
G	a graph, $G \triangleq V, E, P$
V	the node set in graph G, $V = \{v_1, v_2, \cdots, v_n\}$ (n is the number of nodes.)
E	the edge set in graph G, $E = \{(v_i, v_j), \cdots\}$ $(i,j) \in [1,n]$
u v	the node in V
P	the probabilities set, $P \triangleq P_1, P_2$
P_1	the success probabilities of the influence diffusion on edges in E
P_2	the stability probabilities of the edges
$f_q(u,v)$	the similarity of u and v within the distance of q
$R_q(u)$	a set of nodes whose minimal distance is q from node u
$R'_q(u)$	a set of nodes whose minimal distance is q from node u only through PULs
$a(R_q(u))$	an ordered degree sequence of the nodes in $R_q(u)$
$b(R_q(u))$	an ordered degree sequence of the nodes in $R'_q(u)$
$g(.,.)$	the distance between the two ordered degree sequences
σ	the attenuated parameter of the diffusion probabilities
δ	the attenuated parameter of one link
b'_{uv}	the current influence value diffused on node v by node u through unstable link (u,v)
b_{uv}	the influence value for u to influence on v by their first-time linking
γ	the threshold value of link $(u,v)'s$ probability of the positive influence
γ_v	the link $(u,v)'s$ probability of the positive influence during the diffusion

IV. Main Model and Approach

In this section, an influence maximization model based on an MAS is designed where two diffusion models (i.e. the UIC and ULT models) and an US2vec algorithm are established to maximize the influence diffusion.

A. *Influence maximization model on an MAS*

In this section, an influence maximization model based on agents is presented according to the **Influence Maximization** problem.

As shown in Fig. 2, every agent represents one node and the links between agents denote the edges between nodes. The dotted lines represent PULs and the full lines denote normal links. The availability of PULs is implemented by agent interactions. For example, in Fig. 2, the link between agent a_7 and a_9 is unstable and the availability probability of this link is $p_{2(7,9)}$. At every diffusion iteration, a_7 and a_9 will give probability values $p_{7(7,9)}$ and $p_{9(7,9)}$, respectively.

Remark 2: Especially, the probability value given by agent a_2 is marked as $p_{2'(.,.)}$. As long as $p_{7(7,9)} > p_{2(7,9)}$ *and* $p_{9(7,9)} > p_{2(7,9)}$, the link is available for the influence diffusion. In addition, in Fig. 2, the activated nodes are marked by grey, and the influence will be spread by them on the corresponding diffusion model until no node can be activated.

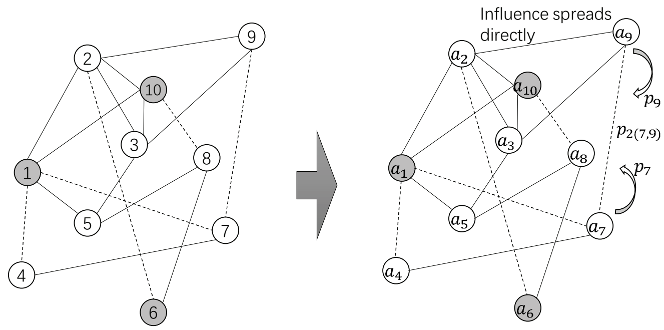


Fig. 2. The influence maximization model based on MAS

Different from traditional influence maximization models, the agent-based model proposed in this paper has a crucial advantage of agent interactions. In order to facilitate appropriate simulations (that mimic human activities) and accelerate the influence diffusion by agents' self-organizing, the rules of agent interaction are given as follows:

1) observe the interactions of the agents to decide the probability with which the PULs occur;
2) observe the success rate (for the links to be stable) and evaluate the failure rate so as to predict the availability of the successfully activated nodes; and
3) exchange the context information (including an agent's action strategies and its currently available links) between an agent and its linked ones.

It should be noticed that the interactions among agents include information exchanging and action decisions, which also helps the process of influence diffusion directly. The specific algorithms of agent interactions and how they promote the influence diffusion are shown in Section IV-B.

Next, according to the model in Fig. 1, the main algorithm of influence maximization based on MAS is shown in Algorithm 1.

Algorithm 1 Main Algorithm

Input: A network $G = (V, E, P)$ and k, a diffusion model and its attenuated parameters

1: Initialize the seed node set SN.
2: Using US2vec algorithm to embedding the nodes including spreading capability, faith evaluation, link information and structural similarity.
3: Select the node which has the biggest integrated evaluation in the results of US2vec by Algorithm 4 and put it into SN.

Output: The seed set SN.

B. Agent interactions

As mentioned in the previous section, agent interactions play significant roles in information exchange and influence diffusion. The agent interactions are divided into three aspects: the decision of the availability of PULs, evaluation of their neighbors and communication of their environment change.

Firstly, the decision of the availability of PULs occurs to any two agents which are linked by an unstable link. According to the definition of **Unstable Link**, the availability of the links is decided by the two agents and p_2. To be specific, the two agents give two probabilities and, if both probabilities are greater than p_2, then the link is available. The probability given by an agent can be comprehended as a willingness index of establishing the link of this agent at the moment, just like a relationship between two persons. Besides, the probability from one agent is not random but based on the observation of its neighbors.

Let a_i and a_j be two agents linked by an unstable link (i, j) and can observe the actions of each other, which means that every decision of a_i (a_j) on any of its unstable link can be observed by a_j (a_i). Then, the probability generated by them on the unstable link can be calculated as follows:

$$p_{i(j)(i,j)} = \begin{cases} (p_{2(i,j)}, 1), & p, p \in (n'_i/n_i, n'_j/n_j) \\ (0, p_{2(i,j)}), & 1 - p, p \in (n'_i/n_i, n'_j/n_j) \end{cases} \quad (5)$$

where $p_{i(j)(i,j)}$ denotes the probability given by agent $i(j)$. n'_i represents the number of times that agent a_i's probability on one unstable link is bigger than the unstable link's probability and n'_j indicates agent a_j's. n_i is the number of agent a_i's historical actions and n_j indicates the number of agent a_j's. p is a random value generated in the range of $(n'_i/n_i, n'_j/n_j)$. It should be mentioned that the occurrence of the first communication between the two agents has an arbitrary probability.

It can be seen that n'_i/n_i is an evaluation of agent a_i's faith, reflecting its average willingness on all its PULs. Then, the faith evaluation of agent a_i can be defined as:

$$f_{a_i} = n'_i/n_i \quad (6)$$

Secondly, the evaluation of an agent's neighbors means the judgement of its diffusion capability. The diffusion capability dc_{a_i} of a_i can be calculated by Eq. (7):

$$dc_{a_i} = \left(nl_2 + \left\lfloor \sum_{(i,j) \in Un_{a_i}(2)} p_2(i,j) \right\rfloor \right) / tl_2 \quad (7)$$

where nl_2 denotes the number of stable links among the 2-hop range of a_i, tl_2 means the amount of links of 2-hop range of a_i and $Un_{a_i}(2)$ represents the set of PULs in the same range.

Thirdly, the agent communication of environment change is an effective way for information update and prediction. In every selection step of the seed node, a basic link prediction of their neighbors can be calculated by every agent's observation. Then, the selection of a seed node can be guided by predictions. For instance, if agent a_i has any changes on its PULs, it will tell all neighbors linked to it. At the same time, the prediction of availability of its PULs can be calculated by:

$$pre_{a_i} = \left\lfloor \left(\frac{1}{I} \sum_{q=1}^{I} nt'_q \right) / nt \right\rfloor \quad (8)$$

where pre_{a_i} indicates the prediction number of a_i's PULs, q is the diffusion iteration number, nt is the number of a_i's PULs and nt'_q represents the number of the PULs available in the q^{th} iteration.

According to Eqs. (6)-(8), the information vector of agent a_i can be defined as $IV_{a_i} = (f_{a_i}, dc_{a_i}, pre_{a_i})$. Note that the information vector IV_{a_i} for agent a_i is embedded by the algorithm US2vec, which means the node information is also included in the process of graph embedding. The solution of the seed set will be improved and supported by the embedding of the nodes' information.

The interactions of every agent in every diffusion iteration are shown in Algorithm 2. All the interactions are based on the observations of the agent's neighbors. Every decision and action of neighbors can be observed by agent-self through links. To summarize, one agent observes and records neighbors' behaviors, then evaluates neighbors and decides the corresponding probabilities.

Algorithm 2 Interaction Algorithm for Every Agent a_i

```
Obtain the information of neighbors' actions by observations.
According to the data of observations:
for q=1 to |Neighbor_{a_i}| do
    if a_q and a_i are linked by an unstable link then
        Give the probability on this unstable link by Eq. (5);
        According to the probabilities of the two agents, establish the link or not.
    end if
    Calculate the neighbor a_q's faith evaluation by Eq. (6);
    Evaluate a_q's diffusion capability by Eq. (7);
end for
Predict the availability of a_i's PULs by Eq. (8);
Generate a_i's IV_{a_i};
Update the record of information and ready to be accessed to;
```

C. Unstable-Similarity2vec

Since only the structural information (e.g. degrees and circles) has been taken into consideration in the general structure-based graph embedding methods, such methods are not appropriate for the influence maximization problem under PULs. According to the model in Section IV-C, the diffusion capability, faith evaluation and link information among agents should be taken into the calculation of the US2vec algorithm. The US2vec method is designed in Algorithm 3.

In line 1 of Algorithm 3, the similarity between any two agents and the initialized information vectors are obtained. After that, the distance between the agents is calculated and the weighted layered graph is drawn. According to the similarity and the information vector, the weight between any two nodes in every layer can be obtained. Then, the stepping probability in the random walk is calculated by:

$$p_{st} = \begin{cases} \frac{e^{-f_q(u,v)}}{\sum_{t \in V} e^{-f_q(u,t)}}, & \text{stay in the current layer } q \text{ and stepping from } a_u \text{ to } a_v \\ \frac{w(u_q,u_{q+1})}{w(u_k,u_{q+1})+w(u_q,u_{q-1})}, & \text{go to layer } q+1 \\ \frac{w(u_q,u_{q-1})}{w(u_k,u_{q+1})+w(u_q,u_{q-1})}, & \text{go to layer } q-1 \end{cases} \quad (9)$$

where p_{st} indicates the stepping probability and $w(u_q, u_{q+1})$ means the weight from the layer u_q to u_{q+1}. The calculation of $w(u_q, u_{q+1})$ is shown in Eq. (10):

$$w(u_q, u_{q+1}) = \log(\Gamma_q(u) + \Gamma'_q(u) + e) \quad (10)$$

where $\Gamma_q(u)$ ($\Gamma'_q(u)$) indicates the number of node u's stable (unstable) links whose weight is larger than the average weight of all stable (unstable) links in layer q. Besides, $w(u_q, u_{q-1})$ is set as 1, which is referred from the struc2vec method [57]. In Eq. (9), the walking will stay in the current layer by the probability of the first mathematical expression. If the walking changes the layer, it will go to the next layer by the second expression and go to the last layer by the third expression. It can be noticed that the sum of the second and the third expressions is 1.

Algorithm 3 US2vec Algorithm

Input: A network $G = (V, E, P)$

```
1: Calculate the similarity between any two nodes according to the definition of structural similarity and q in Eq. (1) and access to the initialized information vectors of these two nodes among q-hop neighbors.
2: Measure the information vectors of any two nodes by distance calculation: DC ← d(u, v).
3: Obtain the edge weight on any two nodes by e^{f_q(u,v)d(u,v)}.
4: Build weighted and layered graph by different q-values.
5: Implement random walk method to generate a sequence of nodes to determine the context of every node according to the stepping probability by Eq. (9).
6: Training the model by the word2vec method (Skip-Gram) according to the sequences generated in Step 5.
7: Use the trained model to classify all the nodes.
```

Output: The classification sets CS.

In Algorithm 3, the initializing of information vectors is implemented by agent interactions. In this step, the third rule of agent interaction is executed by all agents for 100 times, thus, the information vector of any agent can be calculated with the historical action records by Eqs. (6)-(8). On line 2, the distance between IVs of any two nodes have been recorded in the set DC. According to the similarities and distances obtained on lines 1-2, the weighted and layered graph has been established on lines 3-4. It should be noticed that not only the similarities but also the embedding of the information vectors are calculated in US2vec on line 3. Then, the model for node classifying has been generated by random walk and word2vec training on lines 5-6.

Based on the results of US2vec, the generation of seed nodes is described by Algorithm 4.

In Step 1 of Algorithm 4, the rule of selecting seed nodes is shown. According to the classifications from Algorithm 3, in every iteration, agents with the maximum diffusion capability are added into the seed set SN. In the case that the number of agents (whose diffusion capability equals $f_{a_{max}}$) is more than one, the filling of the seed set is accelerated and thus, the selection method is efficient. With the generated new seed nodes and the currently activated nodes, the information of all nodes is updated for the next iteration.

Algorithm 4 Selection Algorithm

Input: A network $G = (V, E, P)$, the classified set CS.

1: Select the agents who have the best diffusion capability $dc_{a_{max}}$ in every classification.
2: Update the prediction information by agents communications.
3: Measure the inactivated nodes by the updated information vectors and predictions values.
4: Mark the agent activating state by influence spreading on the corresponding diffusion model.
5: Repeat Steps 1 and 4 until the node set is full.

Output: The seed node set SN.

V. Experiments and Analysis

In this section, the proposed procedure of US2vec for the influence maximization problem on the MAS with two diffusion models is tested to show the effectiveness of the developed algorithm by evaluating the influence spreading results. Firstly, the spreading results of several influence maximization algorithms are compared with US2vec under the two diffusion models on the MAS in Section V-A. Secondly, in order to test the effectiveness of US2vec, the results under different link settings are presented in Section V-B. Finally, the whole model and algorithm are analyzed in Section V-C. The whole framework is implemented in C++ and Python, and tested in the environment of the TensorFlow 2.0 CPU vision.

A. Experiments on various algorithms for IM

This subsection focuses on the results of the influence spreading by running several algorithms: the random algorithm, meta heuristic (MH), maxdegree method, stru2vec algorithm and US2vec algorithm. The random algorithm focuses on selecting the seed nodes randomly to establish the seed set, and its results always present a fluctuant trend. The influence maximization problem is considered as a $0 - 1$ integer linear programming problem by the meta heuristic algorithm and has been solved by using the meta heuristic idea. The maxdegree method is a heuristic algorithm based on degree centrality, which means it chooses a seed node that has the max degree.

In order to compare the US2vec approach with other existing ones, the Twitter social network database is chosen for this part of the experiments, because of its rich characters and real-world links. Then, in order to test the robustness and effectiveness of US2vec, the modified Twitter database is introduced for further experiments. The original database is sampled and modified into a new one that has a higher similarity on the structure through the following steps:

1) *Users sampling.* First, the users who have the maximum links are sampled and, according to these users, the rest of the users are sampled by the Breadth-First-Search method until the sample set is full.
2) *Weight setting.* This step sets the weights for every link of the sampled users. First, the weight for diffusion probability is set as the occurrence frequency of this link in the original database and the index of link stability is set as a random value in the range of $(0, 1)$ based on the probability of p or set as 1 according to the probability $1 - p$, where p is the percentage of unstable links in the settings of the experiment. As soon as the new database is obtained, every user has its observation and record on their neighbors in the MAS model.

In order to display the performance of the two diffusion models, 500 users are sampled from the original database. The results of the five methods are shown in Fig. 3 and Fig. 4.

By sampling 500 users in the database and reset their link parameters, the five algorithms are run 100 times on the same new database, and the average percentages of diffusion results are shown in Fig. 3. It can be seen from Fig. 3 that on the UIC model, the best performance belongs to the US2vec algorithm and the random method gets the worst results. With the growing number of the seed set size, all the percentages of influence diffusion of algorithms is improved. The reason for this phenomenon is that the more the number of seed nodes is, the larger the scope of the influence is. The maxdegree and random approaches have relatively smaller percentages of influence diffusion due to the fact that, the basic greedy and random idea cannot deal with the PULs among users even though the seed set size is growing. The MH method and stru2vec have better performances than the maxdegree and random as unwanted users are considered in the meta heuristic which helps to avoid the influence diffusion on PULs to a certain extent. Compared with US2vec, the simple structural similarity is calculated in stru2vec without unstable similarities, which leads to the worst performance.

The diffusion results on the ULT model are shown in Fig. 4. Comparing with Fig. 3, the lines of the five methods have similar trends. Because there are negative influence calculations in the ULT model through Eq. (3), the maximum of the diffusion percentage is lower than that in the UIC model in Fig. 3. According to Fig. 3 and Fig. 4, it can be seen that US2vec can deal with PULs in the network and obtain a better result than the existing methods designed using the two proposed diffusion models.

B. The effectiveness of US2vec

In order to test the effectiveness and efficiency, the algorithms are run on different sizes of the database. 600, 1200, 1800, 2400 and 3000 users are, respectively, sampled in the original database. After modifying the sampled database as discussed previously, the information of PULs and agent observations are set. According to Section V-A, it can be seen that better performance is obtained by the US2vec, stru2vec and MH methods. For the purpose of showing the advantage of the US2vec method on the network with PULs, these three methods are chosen to run 100 times on the two diffusion models and the results are presented in Fig. 5 and Fig. 6.

From Fig. 5, the MH approach gets a trend where the number of activated nodes gradually descends. The reason for this phenomenon is that with the growing number of nodes in the database, the meta heuristic cannot handle the exponential searching space and the extremely complicated calculation. Besides, the PULs in the database are not taken

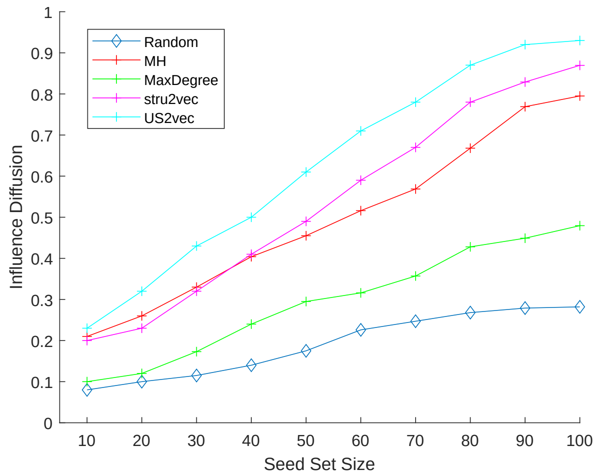


Fig. 3. The results on UIC model with 500 users

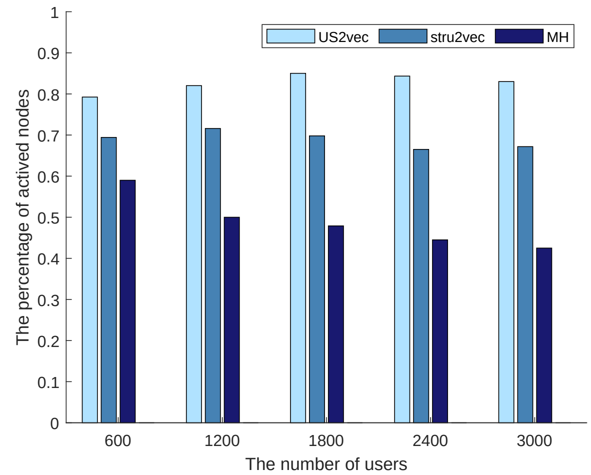


Fig. 5. The results on UIC model with different user numbers

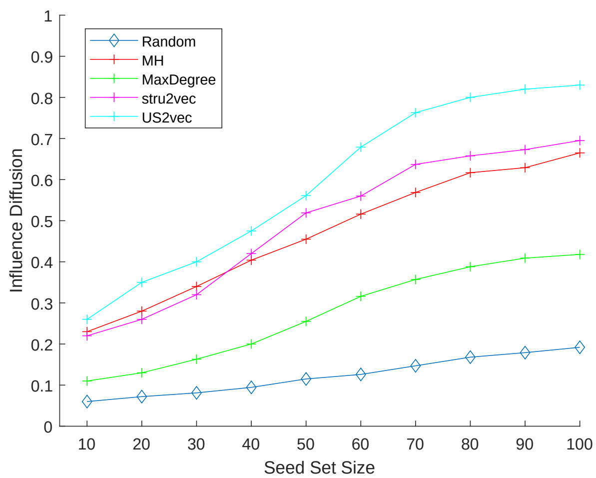


Fig. 4. The results on ULT model with 500 users

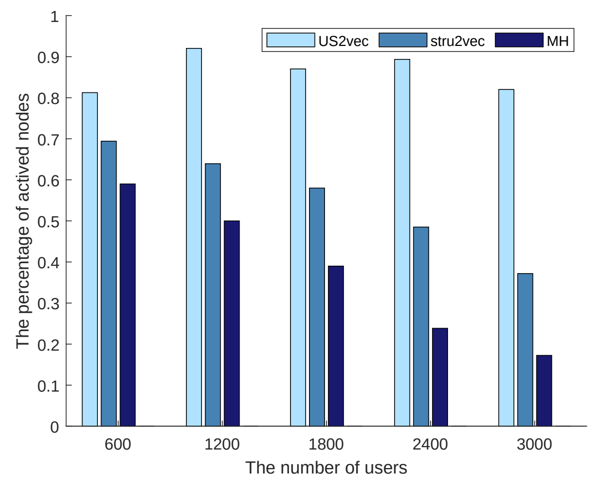


Fig. 6. The results on ULT model with different user numbers

into consideration, and a part of nodes that can be activated are missed because of broken links. In contrast, stru2vec gets better results on different node sizes because similar structures of nodes help diffusion calculations. In addition, it can be easy to figure out that the best robustness is shown by US2vec. Whatever the number of nodes is, the best performance (over 80%) is obtained by US2vec.

Similar results can also be found in the ULT model in Fig. 6. US2vec gets the best performance on different sizes of database among the three algorithms. Compared with Fig. 5, under the ULT model, MH and stru2vec methods have worse performance than US2vec due to the effect from the negative influence diffusion.

Except for the node size, the set of the percentage of PULs is also an important factor that affects the effectiveness of influence diffusion. The sampled database is modified with different settings of the percentage of PULs to test US2vec's efficiency. The results of US2vec under the two models in different settings (of the user number and the percentage of PULs) are shown in Fig. 7 and Fig. 8.

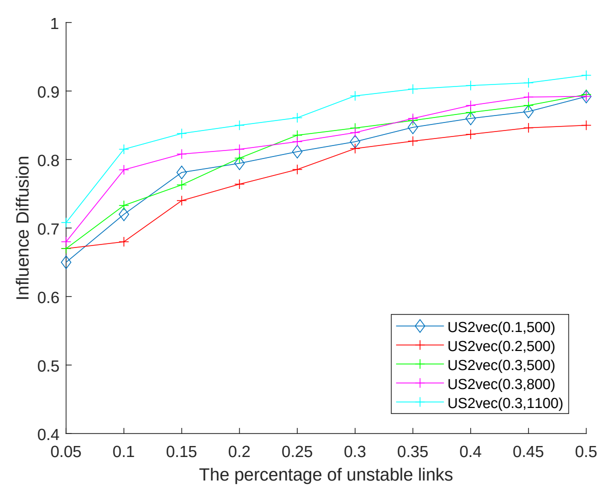


Fig. 7. The results on UIC model with different settings

From Fig. 7 and Fig. 8, one can see that the percentage of

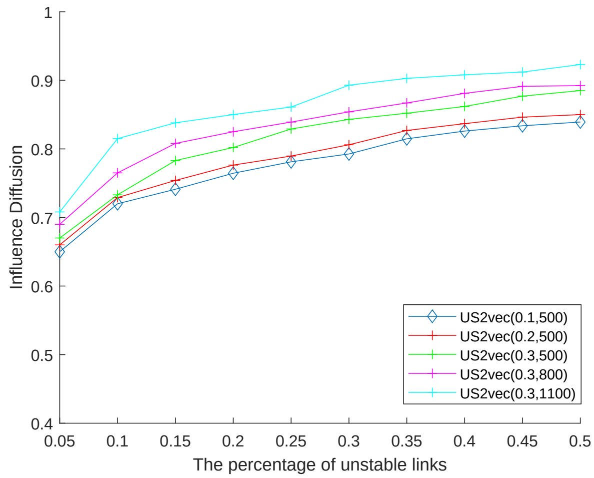


Fig. 8. The results on IC model with different settings

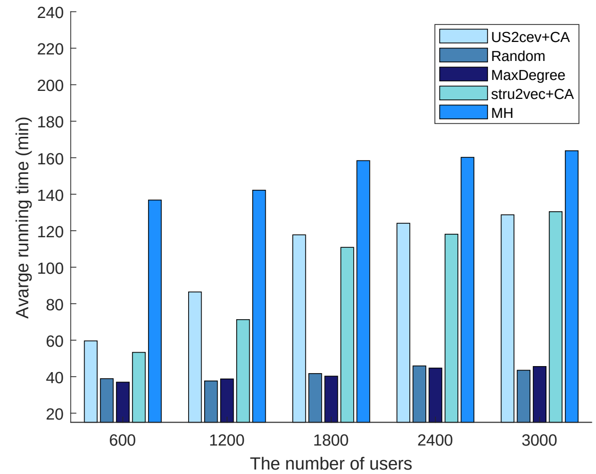


Fig. 9. The results on UIC model with different settings

activated nodes grows when US2vec runs under 500 users with different settings of the percentage of PULs including 10%, 20% and 30%. Additionally, the influence diffusion can always be insured by US2vec with 30% of PULs in the network no matter the node number is 500, 800 or 1100. It should be noticed that when the percentage of PULs is 5%, the diffusion results are in the range of $65\sim72\%$. These results are similar to that of stru2vecc because a low percentage of PULs cannot reflect the advantages of US2vec. In the two diffusion models, a stable percentage of activated nodes $70\sim90\%$ is guaranteed by US2vec.

C. Running time of the whole algorithm

To evaluate the running time of the US2vec+CA method on the influence maximization issue, several approaches are tested on the two diffusion models to compare with the proposed method. The experiments are set as: the size of seed set is 15% of the population, and the percentage of PULs is 40%. After running every approach on the two diffusion models 100 times, the average time costs are displayed in Fig. 9 and Fig. 10.

It can be seen in Fig. 9 that, the running time is quite different between various algorithms on the UIC model. For these five methods, it is obvious that the running time does not grow with the increasing population of nodes. The random and maxdegree algorithms are faster than others, and the random method is even faster than the maxdegree method. The calculation of the maxdegree method always changes because of the PULs, which requires more costs on the time. Although there is a little fluctuation on the average running time of the MH method, the performance can still be maintained as the longest one in (140,160) minutes. The performance of Struc2vec+CA and US2vec+CA are similar where the time increases with the increase of the population. The difference is that more time is required by US2vec+CA, and the reason is that 40% of links are unstable causing calculation complexities on similarities. It should be noticed that graph-based methods obtain the average running time among (55,130) minutes.

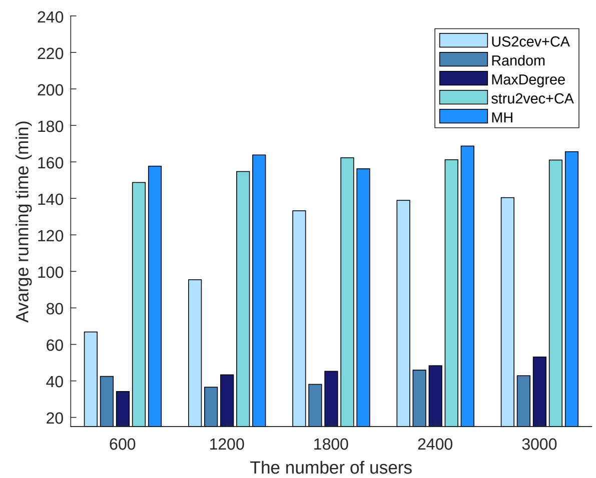


Fig. 10. The results on ULT model with different settings

Comparing the results on the UIC model in Fig. 9, the performance of five algorithms on the ULT model is exhibited in Fig. 10. Apparently, there is little difference in the performance of the random, maxdegree and MH methods on the two models, and more running time is required by Struc2vec+CA and US2vec+CA approaches. The reason is that the CA algorithm requires a long running time on the ULT model because the updating calculations of the information vector occupy a part of the running time.

D. A case study: an advertisement recommendation in a campus network

The campus network is a kind of popular vehicle for communications among university students. The topic of advertisements recommendation in a campus network is chosen as a case study for the whole algorithm. The reason for choosing the topic are as follows: 1) a campus network has a number of unstable links because of the poor quality of fiber and the low bandwidth; 2) the population of users in the campus network is apposite for the calculation of GE method and analysis of IM problem; 3) the structure of the campus network and the database of users are easy to obtain.

The database of the campus network in this case study had been obtained from the school forum in the campus network of Hunan industry polytechnic. There are 5,100 users on this forum platform and 2,000 visits can be found per day. Social relationships, online frequencies, message sending success rates, failure rates of links are included in the original database. An application advertisement of curriculum timetable is recommended to student users in this campus network, and the influence range can be maximized by US2vec and Selection algorithms.

In this advertisement recommendation, the ULT model had been applied to the campus network. Considering that students usually accept a new product through friends and classmates, and when the influence of recommendations has been accumulated over than people's threshold, they will accept the product. Thus, the ULT model is the most precise simulation for this phenomenon. Corresponding to the influence maximization model based on MAS, the real scenarios of campus network have been mapped to the model as follows.

1) The weight of the link between two users (*u* and *v*) is indicated by $p_{1(u,v)}$, and this weight denotes the influence value when the advertisement influence is spreading on this link by ULT model;
2) The connected probability of the unstable link between two users (*u* and *v*) is indicated by $p_{2(u,v)}$, in this case $p_{2(u,v)}$ reflects the connection quality of the campus network.
3) The probabilities $p_{u(u,v)}$ and $p_{v(u,v)}$ indicate the probability of being willing to communicate with each other in every iteration of influence spreading.

After data preprocessing of campus network, the network model with unstable links can be established. Because the user population of campus network is 5100, the whole network visualization is full of overlapping nodes. Thus, the visualized topology with 200 nodes is shown in Fig. 11. In Fig. 11, unstable links and stable links are indicated by dotted lines and solid lines. In order to initialize advertisement diffusion, all nodes are not activated and marked as blue nodes.

The experiments of maximizing advertisement diffusion in the campus network are running with different settings: the percentage of seed set is set as 8%, 10% and 15%, respectively; the node population is set as 200, 1000, 3000 and 5100. With the running of US2vec and SA algorithms on ULT models 100 times, the average number of activated nodes are displayed in Fig. 12.

In Fig 12, when the number of seed nodes is growing, the range of diffusion is becoming large, which means an increasing number of users are accepting the advertisement. It is no doubt that excellent effectiveness was obtained in the campus network, the diffusion percentage always more than 80% no matter how many the user population is. Especially, the visualized network of 200 users (the percentage of seed nodes is 8%) is shown in Fig. 13, activated nodes are indicated by the yellow nodes. Efficient contributions are made by the ULT model and the two main algorithms (US2vec and SA) on dealing with unstable links, and the MAS model based on GE

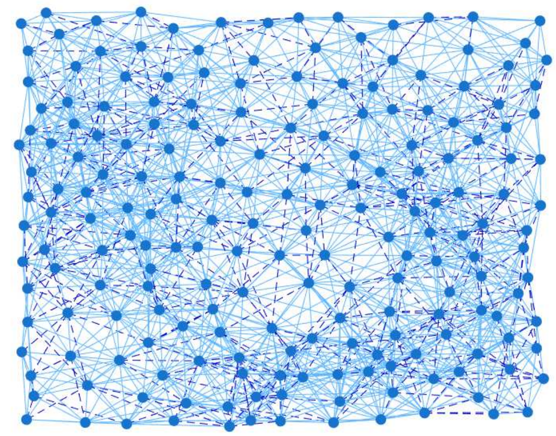


Fig. 11. The visualized topology of 200 users in the campus network

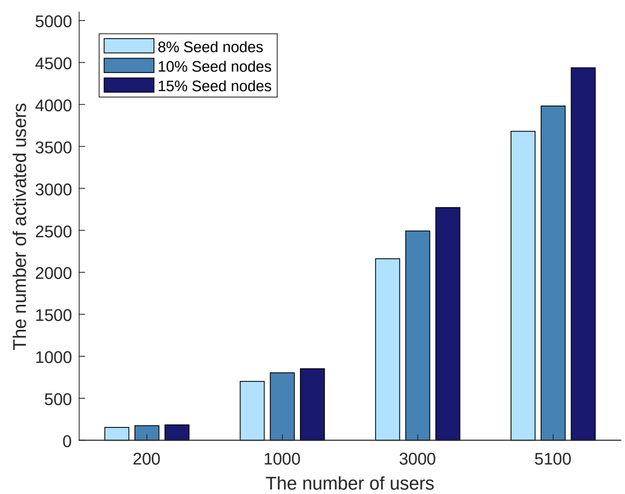


Fig. 12. The results of advertisement recommendation among different number of users

has been successfully applied to the maximization problem on advertisement recommendation in the campus network.

VI. Conclusion

In this paper, the influence maximization issue under the network with PULs has been addressed via the graph embedding method on MASs. To address the difficulty of influence diffusion on PULs, two diffusion models and a novel MAS model have been proposed. To be specific, the calculations of influence values have been included in the two diffusion models and the agent interaction rules designed for influence diffusion have been defined by the MAS model. A solid foundation of influence diffusion for the network with PULs has been established by the three models. Then, according to the structural similarity of PULs and agent interactions, the nodes have been embedded by a novel graph embedding method, i.e. the US2vec algorithm. It should be pointed out that the integrated information of the faith evaluation, the

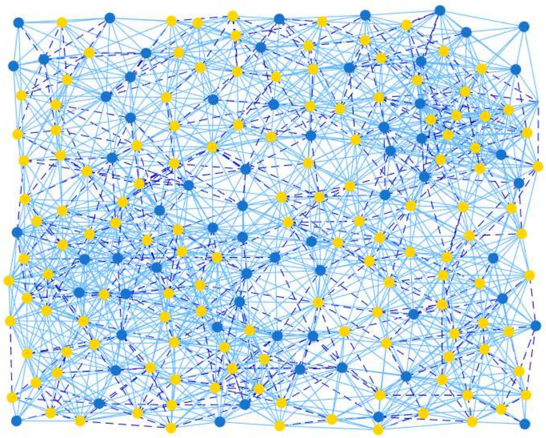


Fig. 13. The result of advertisement recommendation among 200 users in the campus network

diffusion capability and the prediction of availability have been embedded simultaneously in this algorithm. The diffusion effectiveness has been accelerated by the embedding of integrated information of agents. Based on the results of the US2vec algorithm, the seed set can be figured out by the CA method. By activating the nodes in the seed set and diffusing influence in the corresponding diffusion model, the range of influence diffusion has been maximized. Finally, the perfect performance of the US2vec algorithm and the CA method have been verified by extensive experiments for different parameters under different scenarios.

To summarize, the proposed MAS model and the US2vec algorithm have shown excellent superiority in solving the influence maximization problem under the network with PULs. Accurate and efficient solutions can be brought out for realistic problems by combining the powerful simulation capacity of MASs, the node classification and the link prediction of graph embedding algorithms. Thus, in the future, our next study direction is to apply novel graph embedding ideas (based on the MAS model) to the realistic problem so as to look into the structures of society or rules of human activities [25], [27], [42], [52], [53]. It should be mentioned that the study for graph embedding on the dynamic graph can be a new challenge for MAS simulations on the graph evolution of the real network [5], [54].

REFERENCES

[1] O. AskariSichani and M. Jalili, "Influence maximization of informed agents in social networks," *Applied Mathematics and Computation*, vol. 254, pp. 229–239, 2015.

[2] S. Banerjee, M. Jenamani and D. K. Pratihar, "A survey on influence maximization in a social network," *Knowledge and Information Systems*, pp.1–39, 2020.

[3] G. Bao, L. Ma and X. Yi, Recent advances on cooperative control of heterogeneous multi-agent systems subject to constraints: A survey, *Systems Science & Control Engineering*, vol. 10, no. 1, pp. 539-551, Dec. 2022.

[4] S. Chen, X. Yin, Q. Cao, Q. Li and H. Long, "Targeted influence maximization based on cloud computing over big data in social networks," *IEEE Access*, vol.8, pp.45512–45522, 2020.

[5] Y. Chen, K. Ma and R. Dong, Dynamic anti-windup design for linear systems with time-varying state delay and input saturations, *International Journal of Systems Science*, vol. 53, no. 10, pp. 2165-2179, Jul. 2022.

[6] Y. Chen, Q. Song, Z. Zhao, Y. Liu and F. E. Alsaadi, Global Mittag-Leffler stability for fractional-order quaternion-valued neural networks with piecewise constant arguments and impulses, *International Journal of Systems Science*, vol. 53, no. 8, pp. 1756-1768, Jun. 2022.

[7] G. Chowdhury, M. Srilakshmi, M. Chain and S. Sarkar, "Neural Factorization for Offer Recommendation using Knowledge Graph Embeddings," *eCOM@ SIGIR*, 2019.

[8] H. Chu, D. Yue, C. Dou and L. Chu, "Adaptive PI control for consensus of multiagent systems with relative state saturation constraints," *IEEE Transactions on Cybernetics*, vol. 51, no. 4, pp. 2296–2302, 2021.

[9] G. Giakkoupis, H. Saribekyan and T. Sauerwald, "Spread of Information and Diseases via Random Walks in Sparse Graphs," *34rd International Symposium on Distributed Computing (DISC 2020)*, 2020.

[10] D. Kempe, J. M. Kleinberg and É. Tardos, "Maximizing the spread of influence through a social network," *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 137–146, 2003.

[11] F. Hosseini, A. James and M. Ghaderi, "Probabilistic Virtual Link Embedding Under Demand Uncertainty," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1552–1566, 2019.

[12] K. Huang, J. Tang, K. Han, X. Xiao, W. Chen, A. Sun, X. Tang and A. Lim, "Efficient approximation algorithms for adaptive influence maximization," *arXiv preprint arXiv:2004.06469*, 2020.

[13] E. Güney, M. Leitner, M. Ruthmair and M. Sinnl, "Large-scale influence maximization via maximal covering location," *European Journal of Operational Research*, vol. 289, no. 1, pp. 144–164, 2020.

[14] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Knowledge-Based Systems*, vol. 151, pp. 78–94, 2018.

[15] E. Katz and P. F. Lazarsfeld, "The part played by people in the flow of mass communications, transaction publishers, piscataway," *New Jersey*, 1966.

[16] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," *Proceedings of the 2001 SIAM International Conference on Data Mining*, pp. 1–11, 2001.

[17] M. Kimura, K. Saito and H. Motoda, "Solving the contamination minimization problem on networks for the linear threshold model," *Pacific Rim International Conference on Artificial Intelligence*, pp. 977–984, 2008.

[18] M. Kimura, K. Saito, K. Ohara and H. Motoda, "Learning information diffusion model in a social network for predicting influence of nodes," *Intelligent Data Analysis*, vol. 15, no. 4. pp. 633–652, 2011.

[19] M. Khajehnejad, A. A. Rezaei, M. Babaei, J. Hoffmann, M. Jalili and A. Weller, "Adversarial Graph Embeddings for Fair Influence Maximization over Social Networks," *arXiv preprint arXiv:2005.04074*, 2020.

[20] A. Kuhnle, M. A. Alim, X. Li, H. Zhang and M. T. Thai, "Multiplex Influence Maximization in Online Social Networks With Heterogeneous Diffusion Models," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 418–429, 2018.

[21] W. Lalouani, M. F. Younis and N. Badache, "Interconnecting isolated network segments through intermittent links," *Journal of Network and Computer Applications*, vol. 108, pp. 53–63, 2018.

[22] A. Landherr, B. Friedl and J. Heidemann, "A critical review of centrality measures in social networks," *Business & Information Systems Engineering*, vol. 2, no. 6, pp. 371–385, 2010.

[23] S. Lerique, J. L. Abitbol and M. Karsai, "Joint embedding of structure and features via graph convolutional networks," *Applied Network Science*, vol. 5, no. 1, pp. 1–24, 2020.

[24] J. Li and J. J. Cimino, "Auditing dynamic links to online information resources," *AMIA Annual Symposium Proceedings*, vol. 2007, pp. 448, 2007.

[25] W. Li, Y. Niu and Z. Cao, Event-triggered sliding mode control for multi-agent systems subject to channel fading, *International Journal of Systems Science*, vol. 53, no. 6, pp. 1233–1244, 2022.

[26] X. Li, Q. Song, Y. Liu and F. E. Alsaadi, Nash equilibrium and bang-bang property for the non-zero-sum differential game of multi-player uncertain systems with Hurwicz criterion, *International Journal of Systems Science*, vol. 53, no. 10, pp. 2207-2218, Jul. 2022.

[27] X. Li, Q. Song, Z. Zhao, Y. Liu and F. E. Alsaadi, Optimal control and zero-sum differential game for Hurwicz model considering singular systems with multifactor and uncertainty, *International Journal of Systems Science*, vol. 53, no. 7, pp. 1416–1435, 2022.

[28] Y. Lu, W. K. Wong, Z. Lai and X. Li, "Robust flexible preserving embedding," *IEEE Transactions on Cybernetics*, vol. 50, no. 10, pp. 4495–4507, 2020.

[29] R. F. Naill, "Optimizing models of social systems," in *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 201–207, 1976.

[30] M. Ou, P. Cui, J. Pei, Z. Zhang and W. Zhu, "Asymmetric transitivity preserving graph embedding," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1105–1114, 2016.

[31] S. Pan, R. Hu, S.-F. Fung, G. Long, J. Jiang and C. Zhang, "Learning graph embedding with adversarial training methods," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2475–2487, 2020.

[32] K. Saito, R. Nakano and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," *International Conference on Knowledge-based and Intelligent Information and Engineering Systems*, pp.67–75, 2008.

[33] L. Qiu, J. Yu, W. Jia, X. Fan and W. Gao, "A K-shell decomposition based heuristic algorithm for influence maximization in social networks," *International Conference on Genetic and Evolutionary Computing*, pp. 711–722, 2018.

[34] F. Qu, X. Zhao, X. Wang and E. Tian, Probabilistic-constrained distributed fusion filtering for a class of time-varying systems over sensor networks: a torus-event-triggering mechanism, *International Journal of Systems Science*, vol. 53, no. 6, pp. 1288–1297, 2022.

[35] B. Shaw and T. Jebara, "Structure preserving embedding," in *Proceedings of the 26th Annual International Conference on Machine Learning*, pp.937–944, 2009.

[36] S. Srinivas and C. Rajendran, "Community detection and influential node identification in complex networks using mathematical programming," *Expert Systems with Applications*, vol. 135, pp.296–312, 2019.

[37] W. C. Stirling and M. A. Goodrich, "Conditional preferences for social systems," *2001 IEEE International Conference on Systems, Man and Cybernetics. e-Systems and e-Man for Cybernetics in Cyberspace (Cat. No. 01CH37236)*, vol. 2, pp. 995–1000, 2001.

[38] J. Stokes and S. Weber, "On Random Walks and Random Sampling to Find Max Degree Nodes in Assortative Erdos Renyi Graphs," *2016 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, 2016.

[39] Y. Sun, N. Bui, T. Hsieh and V. G. Honavar, "Multi-view Network Embedding via Graph Factorization Clustering and Co-regularized Multi-view Agreement," *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 1006–1013, 2018.

[40] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan and Q. Mei:, "LINE: Large-scale Information Network Embedding," *Proceedings of the 24th International Conference on World Wide Web*, pp. 1067–1077, 2015.

[41] S. Tang, "When Social Advertising Meets Viral Marketing: Sequencing Social Advertisements for Influence Maximization," *AAAI*, 2018.

[42] H. Tao, H. Tan, Q. Chen, H. Liu and J. Hu, H_∞ state estimation for memristive neural networks with randomly occurring DoS attacks, *Systems Science & Control Engineering*, vol. 10, no. 1, pp. 154-165, Dec. 2022.

[43] D. A. Vega-Oliveros, L. F. Costa and F. A. Rodrigues, "Influence maximization by rumor spreading on correlated networks through community identification," *Communications in Nonlinear Science and Numerical Simulation*, vol. 83, pp. 105094, 2020.

[44] D. Wang, P. Cui and W. Zhu, "Structural deep network embedding," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1225–1234, 2016.

[45] L. Wang, S. Liu, Y. Zhang, D. Ding and X. Yi, Non-fragile l_2-l_∞ state estimation for time-delayed artificial neural networks: an adaptive event-triggered approach, *International Journal of Systems Science*, vol. 53, no. 10, pp. 2247-2259, Jul. 2022

[46] L. Wang, Z. Yu, F. Xiong, D. Yang, S. Pan and Z. Yan, "Influence spread in geo-social networks: A multiobjective optimization perspective," *IEEE Transactions on Cybernetics*, vol. 51, no. 5, pp. 2663–2675, 2021.

[47] Y. Wang, G. Cong, G. Song and K. Xie, "Community-based greedy algorithm for mining top-k influential nodes in mobile social networks," *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1039–1048, 2010.

[48] C. Wilson, A. Sala, K. P. N. Puttaswamy and B. Y. Zhao, "Beyond Social Graphs: User Interactions in Online Social Networks and their Implications," *ACM Transactions on the Web (TWEB)*, vol. 6, no. 4, pp. 1–31, 2012.

[49] X. Wu, L. Fu, J. Meng and X. Wang, "Maximizing Influence Diffusion over Evolving Social Networks," *Proceedings of the Fourth International Workshop on Social Sensing*, pp. 6–11, 2019.

[50] S. Xu, S. Liu and L. Feng, "Manifold graph embedding with structure information propagation for community discovery," *Knowledge-Based Systems*, vol. 208, pp. 106448, 2020.

[51] Y. Xu, R. Sun, Y. Wu, C. Chen and X. Wang, "Querying Influential Maximum Connected Community in Large Graphs," *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 757–758, 2020.

[52] H. Yu, J. Hu, B. Song, H. Liu and X. Yi, Resilient energy-to-peak filtering for linear parameter-varying systems under random access protocol, *International Journal of Systems Science*, in press, DOI: vol. 53, no. 11, pp. 2421-2436, Aug. 2022.

[53] L. Yu, Y. Cui, Y. Liu, N. D. Alotaibi and F. E. Alsaadi, Sampled-based consensus of multi-agent systems with bounded distributed time-delays and dynamic quantisation effects, *International Journal of Systems Science*, vol. 53, no. 11, pp. 2390-2406, Aug. 2022.

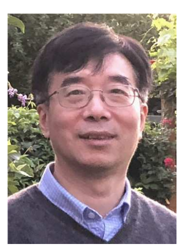
[54] Y. Zhao, X. He, L. Ma and H. Liu, Unbiasedness-constrained least squares state estimation for time-varying systems with missing measurements under round-robin protocol, *International Journal of Systems Science*, vol. 53, no. 9, pp. 1925-1941, Jul. 2022.

[55] H. Zheng, N. Wang and J. Wu, "Non-Submodularity and Approximability: Influence Maximization in Online Social Networks," *2019 IEEE 20th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*, pp. 1–9, 2019.

[56] Y. Zhu, W. Wu, Y. Bi, L. Wu, Y. Jiang and W. Xu, "Better approximation algorithms for influence maximization in online social networks," *Journal of Combinatorial Optimization*, vol. 30, pp. 97–108, 2015.

[57] R. F. R. Leonardo and S. H. P. Pedro and F. R. Daniel , "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 385–394, 2017.

Mincan Li received the Ph.D. degree in computer science and technology from the Department of Information Science and Engineering, Hunan University, Changsha, China, in 2021. She is currently a postdoctoral researcher in the Department of Information Science and Engineering, Hunan University, Changsha, China. Her research interests include multi-agent systems, data engineering, many-objective optimization and machine learning.



Zidong Wang (SM'03-F'14) received the B.Sc. degree in mathematics from Suzhou University, Suzhou, China, in 1986, and the M.Sc. degree in applied mathematics and the Ph.D. degree in electrical engineering from Nanjing University of Science and Technology, Nanjing, China, in 1990 in 1994, respectively.

He is currently Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, U.K. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the UK. Prof. Wang's research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published more than 700 papers in international journals. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for *International Journal of Systems Science*, the Editor-in-Chief for *Neurocomputing*, the Editor-in-Chief for *Systems Science & Control Engineering*, and an Associate Editor for 12 international journals including *IEEE Transactions on Automatic Control*, *IEEE Transactions on Control Systems Technology*, *IEEE Transactions on Neural Networks*, *IEEE Transactions on Signal Processing*, and *IEEE Transactions on Systems, Man, and Cybernetics-Part C*. He is a Member of the Academia Europaea, a Member of the European Academy of Sciences and Arts, an Academician of the International Academy for Systems and Cybernetic Sciences, a Fellow of the IEEE, a Fellow of the Royal Statistical Society, and a member of program committee for many international conferences.

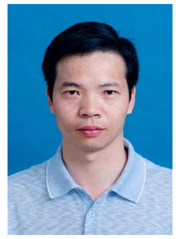
Qing-Long Han received the B.Sc. degree in Mathematics from Shandong Normal University, Jinan, China, in 1983, and the M.Sc. and Ph.D. degrees in Control Engineering from East China University of Science and Technology, Shanghai, China, in 1992 and 1997, respectively.

Professor Han is Pro Vice-Chancellor (Research Quality) and a Distinguished Professor at Swinburne University of Technology, Melbourne, Australia. He held various academic and management positions at Griffith University and Central Queensland University, Australia. His research interests include networked control systems, multi-agent systems, time-delay systems, smart grids, unmanned surface vehicles, and neural networks.

Professor Han was awarded The 2021 Norbert Wiener Award (the Highest Award in systems science and engineering, and cybernetics) and The 2021 M. A. Sargent Medal (the Highest Award of the Electrical College Board of Engineers Australia). He was the recipient of The 2021 IEEE/CAA Journal of Automatica Sinica Norbert Wiener Review Award, The 2020 IEEE Systems, Man, and Cybernetics (SMC) Society Andrew P. Sage Best Transactions Paper Award, The 2020 IEEE Transactions on Industrial Informatics Outstanding Paper Award, and The 2019 IEEE SMC Society Andrew P. Sage Best Transactions Paper Award.

Professor Han is a Member of the Academia Europaea (The Academy of Europe). He is a He is a Fellow of The International Federation of Automatic Control and a Fellow of The Institution of Engineers Australia. He is a Highly Cited Researcher in both Engineering and Computer Science (Clarivate Analytics). He has served as an AdCom Member of IEEE Industrial Electronics Society (IES), a Member of IEEE IES Fellows Committee, and Chair of IEEE IES Technical Committee on Networked Control Systems. He is Co-Editor-in-Chief of IEEE Transactions on Industrial Informatics, Deputy Editor-in-Chief of IEEE/CAA JOURNAL OF AUTOMATICA SINICA, Co-Editor of Australian Journal of Electrical and Electronic Engineering, an Associate Editor for 12 international journals, including the IEEE TRANSACTIONS ON CYBERNETICS, IEEE INDUSTRIAL ELECTRONICS MAGAZINE, Control Engineering Practice, and Information Sciences, and a Guest Editor for 14 Special Issues.

Simon J. E. Taylor received the B.Sc. (Hons.) degree in Industrial Studies and the M.Sc. degree in Computer Studies from Sheffield Hallam University, Sheffield, U.K., in 1986 and 1988, respectively, and the Ph.D. degree in Distributed Simulation from Leeds Metropolitan University, Leeds, U.K., in 1993. He is currently Professor of Computing and Vice-Dean (Research) for the College of Engineering, Design and Physical Sciences at Brunel University. He is also Co-Director of the Modelling & Simulation Group at the same University, a group that has contributed to the area for around 30 years. His research has focused on digital infrastructures and Modelling & Simulation and has impacted over 3 million students and 300 universities in Africa, over 30 SMEs and several large enterprises including the Ford Motor Company and Sellafield. He co-founded the UK Operational Research Society's Journal of Simulation and the UK Simulation Workshop series and continues to be an editor at the Journal. He is a former chair and a member of ACM SIGSIM's Steering Committee, an advisory board member of the NSCU Simulation Archive and executive chair of the international Simulation Exploration Experience. He is has been in various programme committee positions in the IEEE/ACM Winter Simulation Conference for over 20 years and is the General Chair of WSC 2025. He continues to be interested in advances in Modelling & Simulation, international development and Open Science, as well as helping younger colleagues to rapidly start their careers.



Kenli Li received the Ph.D. degree in Computer Science from Huazhong University of Science and Technology, China, in 2003. He was a visiting scholar at the University of Illinois at Urbana-Champaign from 2004 to 2005. He is currently a Cheung Kong Professor of Computer Science and Technology at Hunan University (HNU), the Vice-President of the HNU, the Dean of the College of Information Science and Engineering of HNU, and the Director in the National Supercomputing Center in Changsha. His major research interests include parallel and distributed processing, high-performance computing, and big data management. He has published more than 350 research papers in international conferences and journals such as IEEE-TC/TPDS, HPCA, ISCA, SC, MM, AAAI, DAC, ICDE, etc. He is a Fellow of the CCF and a Senior Member of the IEEE. He is currently serving or has served as an Associate Editor for IEEE-TC, IEEE-TII, and IEEE-TSUSC.

Xiangke Liao received the B.S. degree in Computer Science and Technology from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 1985, and the M.S. degree in Computer Science and Technology from the National University of Defense Technology, Changsha, China, in 1988. He is currently a Full Professor and the Dean of the College of Computer Science, National University of Defense Technology. His research interests include parallel and distributed computing, high-performance computer systems, operating systems, cloud computing, and networked embedded systems.

Xiaohui Liu received the B.Eng. degree in computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in computer science from Heriot-Watt University, Edinburgh, UK, in 1988. He is currently a Professor of Computing at Brunel University London, Uxbridge, U.K., where he conducts research in artificial intelligence and intelligent data analysis, with applications in diverse areas including biomedicine and engineering.