Chaos, Solitons and Fractals 139 (2020) 110287

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

Network structure reconstruction with symmetry constraint

Zihua Hang[a], Penglin Dai[a,*], Shanshan Jia[b,*], Zhaofei Yu[b]

[a] School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China
[b] Department of Computer Science and Technology, Peking University, Beijing 100871, China

ARTICLE INFO

Article history:
Received 7 March 2020
Revised 6 August 2020
Accepted 8 September 2020
Available online 25 September 2020

Keywords:
Complex network
Network structure reconstruction
Compressive sensing

ABSTRACT

Complex networks have been an effective paradigm to represent a variety of complex systems, such as social networks, collaborative networks, and biomolecular networks, where network topology is unkown in advance and has to be inferred with limited observed measurements. Compressive sensing (CS) theory is an efficient technique to achieve accurate network reconstruction in complex networks by formulating the problem as a series of convex optimization models and utilizing the sparsity of networks. However, previous CS-based works have to solve a large number of convex optimization models, which is time-consuming especially when the network scale becomes large. Further, since partial link information shared among multiple convex models, data conflict problem may incur when the derived common variables are inconsistent, which may badly degrade infer precision. To address the issues above, we propose a new model for network reconstruction based on compressive sensing. To be specific, a single convex optimization model is formulated for inferring global network structure by combing the series of convex optimization models, which can effectively improve computation efficiency. Further, we devise a vector to represent the connection states of all the nodes without redundant link information, which is used for representing the unkown topology variables in the proposed optimization model based a devised transformation method. In this way, the proposed model can eliminate data conflict problem and improve infer precision. The comprehensive simulation results shows the superiority of the proposed model compared with the competitive algorithms under a wide variety of scenarios.

1. Introduction

Complex networks have been an effective paradigm to represent a variety of complex systems, such as social networks [1], transportation networks [2], collaborative networks [3], and biomolecular networks [4], etc., where the system is composed of a large number of basic elements, called nodes and the node-to-node interaction relationship determines the network topology. For instance, international logistics companies use the global traffic networks to analyze the cost of goods transportation, which can be is a strong reference to warehouse site selection and transportation route optimization. However, the network topology of these complex networks is always unkown in advance. The popular approach to reveal the network structure is to formulate the network reconstruction problem (NRP) as the system of linear equations by performing time series of interaction activities and observing sufficient interaction measurements, which is known as the inverse problem. When the amount of measured data is sufficient, then the network topology can be easily uncovered by computing the inverse matrix. However, in reality, interaction activity observation is always time-consuming, especially in some areas, such as biology, where the molecular interaction procedure can last for several hours, even a few days. It is common that only a few number of observation data are available, which makes it as an ill-posed problem. Hence, it is nontrivial to address the network structure reconstruction problem in the condition of insufficient measurements.

In the last decades, compressive sensing (CS) has been a mainstreaming technique for network reconstruction in complex networks [5]. The basic principle of CS is to transform NRP problem into a convex optimization model, i.e., L_1-norm minimization problem based on compressive sensing. The CS is able to precisely infer the network structure with only a few number of measured data in the condition of sparse network, which can be easily satisfied in complex networks. Various CS-based models have been proposed to solve the inverse problem of network reconstruction [6]. Wang et al. [7] developed an evolutionary game based model in complex networks and proposed an efficient approach to reconstruct complex networks from small amounts of interaction data. Barranca et al. [8] developed a CS-based framework of a pulse-coupled nonlinear network and reconstructed sparse feed-forward connections based on a hidden linear structure intrinsic to the nonlinear

* Corresponding authors.
E-mail addresses: penglindai@swjtu.edu.cn (P. Dai), jssll00@126.com (S. Jia).

https://doi.org/10.1016/j.chaos.2020.110287
0960-0779/

network dynamics. Barranca and Zhou [9] developed a CS-based model of neuronal networks with the ubiquitous sparse connectivity structure for reconstructing network connections by measuring individual neuronal activity in response to a relatively small ensemble of random stimuli injected over a short time scale. In addition, Han et al. [10] developed a general framework for robust reconstruction from sparse and noisy data by decomposing the task of reconstructing the whole network into recovering local structures centered at each node based on the lasso, a convex optimization method. Huang et al. [11] transferred the prior knowledge of the symmetry of adjacency matrices into a constraint condition for the traditional CS method, which only run the optimization process once. However, the dimension of the proposed model still remains the same as the traditional CS method, which may result in high computation complexity when the network scale becomes large.

Based on the observation above, these CS-based models share some common characteristics. First, they formulate the NRP as multiple convex optimization subproblems, where the solution to one subproblem corresponds to the link information of one node in complex networks. Second, since each node shares partial link information, partial accessible link information can be shared among multiple subproblems, which can accelerate the computation efficiency. However, these common characteristics also brings several critical issues to be addressed. The global network topology can only be acquired by completely solving all these convex optimization subproblems, which is time-consuming especially when the network scale becomes large. Particularly, the number of unknown connection states in the optimization model is proportional to the quadratic power of the node number, which results in high computation complexity when the network scale becomes large. In addition, data conflict may incur when the derived common link information of two subproblems is inconsistent. Particularly, when the inaccurate link information derived by one subproblem is used for structure inferring in the subsequent subproblems, which may badly degrade infer precision.

Based on the motivation above, we propose a new model for network reconstruction based on compressive sensing. To be specific, we first formulate a single convex optimization model for global network reconstruction by combing the series of linear equations, which can effectively improve the computation efficiency without solving a large number of iteration procedures. Further, we devise a new connection state vector to represent the global structure information without redundant variables and propose a transformation method to use the new vector to replace the original variables in the proposed optimization model, which can eliminate the data conflict problem occurred in shared common variables and improve infer precision. The comprehensive simulation results shows the superiority of the proposed model compared with the competitive algorithms under a wide variety of scenarios.

The rest of the paper is organized as follows. Section 2 introduces the preliminaries of network reconstruction problem. Then, we present the proposed method for network structure inference in Section 3. Comprehensive performance evaluations are presented in Section 4. Finally, the conclusions are discussed in Section 5.

2. Network reconstruction problem

In this section, we first introduce interaction activity in complex networks. Then, network reconstruction problem is formally formulated.

First, various types of interaction activities can be performed in complex networks, such as epidemic spreading [6], game dynamics [7] and nonlinear dynamical systems [12]. For simplicity, this paper adopts one of the most popular evolutionary game models, i.e., prisoner's-dilemma game (PDG) [13], which is performed with multiple rounds in order to acquire time series of measured interaction data.

Specifically, for the PDG model, each node in the complex network is regarded as an agent who can choose one of the two strategies at each round, cooperation or defection. Then, the decision of an agent n at t_k round is denoted by a two-dimensional vector, denoted by $\boldsymbol{s}_n(t_k)$, which can be expressed either as cooperation $\left(\boldsymbol{s}_n(t_k) = (1,0)^T\right)$ or defection $\left(\boldsymbol{s}_n(t_k) = (0,1)^T\right)$. Further, the payoff matrix is formulated as follows.

$$\boldsymbol{P}_{pdg} = \begin{pmatrix} p_{cc}, & p_{cd} \\ p_{dc}, & p_{dd} \end{pmatrix}, \tag{1}$$

where p_{cc} and p_{dd} represent the payoff of mutual cooperation and mutual defection respectively, and the mixed choice of strategies gives the cooperator the suckers payoff p_{cd} and the defector the temptation p_{dc}. According to the definition of the PDG, we have $p_{dc} > p_{cc} > p_{dd} > p_{cd}$. In this paper, the value of p_{cc}, p_{cd}, p_{dc}, and p_{dd} are set to 1.0, 0, 1.2, 0, respectively, which is a common setting in these literatures [14–16].

For the game only with two agents m and n, whose strategies are $\boldsymbol{s}_m$ and $\boldsymbol{s}_n$, then their payoffs are computed as follows.

$$\begin{aligned} g_m &= \boldsymbol{s}_m^T \boldsymbol{P} \boldsymbol{s}_n, \\ g_n &= \boldsymbol{s}_n^T \boldsymbol{P} \boldsymbol{s}_m. \end{aligned} \tag{2}$$

Then, for the game with multiple agents, all the agents can play with their neighbors and the gained payoff of player n at t_k round is defined as the summation of all the gained payoffs, which is formulated as follows.

$$g_n(t_k) = \sum_{\forall m \in \Gamma_n} \boldsymbol{s}_n^T(t_k) \boldsymbol{P} \boldsymbol{s}_m(t_k), \tag{3}$$

where Γ_n is the set of agents neighboring to the agent n.

After obtaining the current payoff, the agent follows some designed rules to update its strategy, in order to maximize the payoff at the next round. The classical strategy update rules include the best-take-over rule [17], the Fermi rule [7], payoff-difference-determined updating probability [18] and proportional imitation rule [19], etc. In this paper, we adopt the simplest strategy, i.e., the random rule for strategy update. Specifically, in each round, the agent n randomly selects one agent m from Γ_n and chooses the strategy $\boldsymbol{s}_n(t_k)$ as its strategy, which can avoid the overly rapid convergence of strategies.

However, since the network structure is always unkown in advance, the set of measured data only includes the gained payoff and decision of each agent at each round. Assume that there exist N agents in the system and let a N-1-dimensional binary vector $\boldsymbol{A}_n = (a_{n1}, \ldots, a_{n,n-1}, a_{n,n+1}, \ldots, a_{nN})^T$ denote the connection state of agent n to all the agents in the system, where $a_{nm} = 1$ indicates that agents n and m are connected. Otherwise, i.e., $a_{nm} = 0$, they are disconnected. Particularly, since a_{nn} equals 0, therefore a_{nn} is excluded in connection vector $\boldsymbol{A}_n$. Accordingly, the payoff function of each agent, denoted by Eq. (3), can be replaced by:

$$g_n(t_k) = \sum_{m=1, m\neq n}^{N} a_{nm} \boldsymbol{s}_n^T(t_k) \boldsymbol{P} \boldsymbol{s}_m(t_k), \tag{4}$$

where $g_n(t)$ and $\boldsymbol{s}_n(t)$, $n = 1, 2, \ldots, N$, can be observed at each round of PDG game. After performing K rounds of PDG game, for each agent n, we can get series of strategy decisions, i.e., $\boldsymbol{s}_n(t_1)$, $\boldsymbol{s}_n(t_2)$, ..., $\boldsymbol{s}_n(t_K)$ and series of payoffs, i.e., $g_n(t_1)$, $g_n(t_2)$, ..., $g_n(t_K)$, respectively. On this basis, for each agent n, we can construct the possible payoff matrix and payoff vector, denoted by $\boldsymbol{F}_n$ and $\boldsymbol{G}_n$, respectively.

$$\boldsymbol{F}_n = \begin{pmatrix} \boldsymbol{F}_n(t_1) \\ \boldsymbol{F}_n(t_2) \\ \cdots \\ \boldsymbol{F}_n(t_K) \end{pmatrix} = \begin{pmatrix} f_{n1}(t_1) & \cdots & f_{nN}(t_1) \\ f_{n1}(t_2) & \cdots & f_{nN}(t_2) \\ \cdots & \cdots & \cdots \\ f_{n1}(t_K) & \cdots & f_{nN}(t_K) \end{pmatrix}, \tag{5}$$

$$\boldsymbol{G}_n = (g_n(t_1), \ldots, g_n(t_k), \ldots, g_n(t_K))^T, \tag{6}$$

where $\boldsymbol{F}_n$ represents possible gained payoff of player n with each player at each round. Specifically, $\boldsymbol{F}_n(t_k)$ is the k^{th} row vector of matrix $\boldsymbol{F}_n$, expressed as follows:

$$\boldsymbol{F}_n(t_k) = (f_{n,1}(t_k), \ldots, f_{n,n-1}(t_k), f_{n,n+1}(t_k), \ldots, f_{n,N}(t_k)), \tag{7}$$

where $f_{nm}(t_k) \in \boldsymbol{F}_n(t_k)$ is set to $\boldsymbol{s}_n^T(t_k)\boldsymbol{P}\boldsymbol{s}_m(t_k)$. Since a_{nn} equals to 0, then $f_{n,n}(t_k)$ is excluded. Then, $\boldsymbol{G}_n$ represents the series of gains obtained by player n during the K rounds. On this basis, for each player n, the relationship between the payoff vector $\boldsymbol{G}_n$, possible payoff matrix $\boldsymbol{F}_n$ and connection vector $\boldsymbol{A}_n$ is expressed as follows.

$$\boldsymbol{G}_n = \boldsymbol{F}_n\boldsymbol{A}_n. \tag{8}$$

According to Eq. (8), the network reconstruction problem is transformed into solving the linear equation with observed $\boldsymbol{F}_n$ and $\boldsymbol{G}_n$. Based on theory of linear algebra, the solution of problem is to compute the inverse of matrix $\boldsymbol{F}_n$. When $\boldsymbol{F}_n$ is full rank, it is easy to compute the inverse $\boldsymbol{F}_n^{-1}$. However, in reality, the number of evolutionary round is much smaller than the number of nodes in the system, i.e., $K \ll N$, therefore there exists no inverse of $\boldsymbol{F}_n$. Fortunately, the network in complex system is usually sparse, where compressive sensing can be effectively applied to infer the network structure with insufficient measured data. Then, the global network topology can be obtained by solving a series of linear equations $\boldsymbol{G}_n = \boldsymbol{F}_n\boldsymbol{A}_n$, n= 1, 2, ..., N, where the global connection vector is denoted by $\boldsymbol{A} = (\boldsymbol{A}_1^T, \boldsymbol{A}_2^T, \ldots, \boldsymbol{A}_N^T)^T$. The compressive sensing will be discussed in detail in the following section.

3. Methodology

In this section, we first introduce the basic principle of compressive sensing, which infers the global network topology by solving a large number of convex optimization sub-problems in an iterative way. On this basis, a transformation method is proposed to combine all these sub-problems into an individual optimization model, which eliminates data conflict problem and improve computation efficiency.

First, the goal of compressive sensing is to recover the vector $\boldsymbol{X}$ from linear measurements $\boldsymbol{Y}$ and $\boldsymbol{F}$ in the form:

$$\boldsymbol{Y} = \boldsymbol{F}\boldsymbol{X}, \tag{9}$$

where $\boldsymbol{Y}$ is a K_1-dimension vector and $\boldsymbol{F}$ is a $K_1 \times K_2$ matrix. The striking advantage of CS is to achieve accurate network reconstruction in the condition of that the number of observed measurements is much less than the number of nodes in the networks, i.e., $K_1 \ll K_2$ and the matrix $\boldsymbol{F}$ satisfies the restricted isometry property (RIP) [20]. Due to the sparsity of complex networks, the RIP property is usually satisfied. Then, network reconstruction problem defined in Eq. (9) is transformed into the following optimization model based on compressive sensing.

$$\begin{aligned} &\min ||\boldsymbol{X}||_0 \\ &s.t. \quad \boldsymbol{Y} = \boldsymbol{F}\boldsymbol{X}, \end{aligned} \tag{10}$$

where $||\boldsymbol{X}||_0$ is L_0-norm and defined as the number of non-zero elements in $\boldsymbol{X}$. However, due to non-convexity of L_0-norm, the optimization model defined in Eq. (10) is typically an NP-hard problem, which cannot be solved in polynomial time. Hence, L_1 norm is commonly used instead, which is a convex function and defined as the summation of absolute value of all the elements, i.e., $||\boldsymbol{X}||_1 = \sum_{\forall x_{ij} \in X} |x_{ij}|$. Then, many efforts have be paid on convex optimization based on L_1-norm for solving network-construction problems in complex networks [10,21]. Hence, the network reconstruction problem defined in Eq. (8) can be solved by achieving the optimal solutions of a series of convex optimization models.

$$\begin{aligned} &\min \quad ||\boldsymbol{A}_n||_1 \\ &s.t. \quad \boldsymbol{G}_n = \boldsymbol{F}_n\boldsymbol{A}_n, n = 1, 2, \ldots, N \end{aligned} \tag{11}$$

It is observed that, for any two vectors $\boldsymbol{A}_n$ and $\boldsymbol{A}_m$, they share some common variables. For instance, the element $a_{nm} \in \boldsymbol{A}_n$ should be equal to the element $a_{mn} \in \boldsymbol{A}_m$. However, when $\boldsymbol{A}_n$ and $\boldsymbol{A}_m$ are solved in parallel, the data conflict problem can possibly occur when the derived value of a_{nm} and a_{mn} are inconsistent. To overcome this issue, one common approach is to solve the N optimization models in an iterative way. When a fraction of variables is accessible in advance, they can be used to reduce the data amount for reconstruction in the subsequent optimization models. However, when the number of nodes becomes sufficient large, the iteration-based approach are time-consuming. Even worse, when the solution of common variable in the previous model is not accurate, the error will propagate through the subsequent equations, which may badly degrade the reconstruction accuracy.

Therefore, we propose a transformation method to formulate a single convex optimization model by combining all the N optimization problems and also eliminating the data conflict. Specifically, we define a new possible payoff matrix for each t_k round, denoted by $\boldsymbol{F}'(t_k)$, formulated as follows.

$$\boldsymbol{F}'(t_k) = \begin{pmatrix} \boldsymbol{F}_1(t_k) \\ \boldsymbol{F}_2(t_k) \\ \cdots \\ \boldsymbol{F}_N(t_k) \end{pmatrix} = \begin{pmatrix} f_{12}(t_k) & \cdots & f_{1N}(t_k) \\ f_{21}(t_k) & \cdots & f_{2N}(t_k) \\ \cdots & \cdots & \cdots \\ f_{N1}(t_k) & \cdots & f_{NN-1}(t_k) \end{pmatrix}, \tag{12}$$

where $\boldsymbol{F}'(t_k)$ is $N \times N-1$-dimensional, which consists of all the k^{th} row vectors of $\boldsymbol{F}_n$, $n = 1, 2, \ldots, N$. Further, we transform $\boldsymbol{F}'(t_k)$ as follows,

$$TF\left(\boldsymbol{F}'(t_k)\right) = \begin{pmatrix} \boldsymbol{F}_1(t_k) & 0 & \cdots & 0 \\ 0 & \boldsymbol{F}_2(t_k) & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \boldsymbol{F}_N(t_k) \end{pmatrix}, \tag{13}$$

where $TF(\boldsymbol{F}'(t_k))$ is an $N \times (N-1)N$-dimensional matrix, $k = 1, 2, \ldots, K$. Then, we define a global possible payoff matrix by combining all the transformed matrices, expressed as follows,

$$\boldsymbol{F}' = \begin{pmatrix} TF\left(\boldsymbol{F}'(t_1)\right) \\ TF\left(\boldsymbol{F}'(t_2)\right) \\ \cdots \\ TF\left(\boldsymbol{F}'(t_K)\right) \end{pmatrix}, \tag{14}$$

where $\boldsymbol{F}'$ is a $KN \times (N-1)N$-dimensional matrix.

Similarly, we define a new gain vector for each t_k round, expressed as follows,

$$\boldsymbol{G}'(t_k) = (g_1(t_k), g_2(t_k), \ldots, g_N(t_k))^T. \tag{15}$$

A global payoff vector is defined by combing all the vectors $\boldsymbol{G}'(t_k)$, $k = 1, 2, \ldots, K$, expressed as follows,

$$\boldsymbol{G}' = \left(\boldsymbol{G}'^T(t_1), \boldsymbol{G}'^T(t_2), \ldots, \boldsymbol{G}'^T(t_K)\right)^T, \tag{16}$$

where $\boldsymbol{G}'$ is a KN-dimensional vector.

Accordingly, the series of optimization models defined in Eq. (11) are transformed into a single optimization model.

$$\begin{aligned} &\min \quad ||\boldsymbol{A}||_1 \\ &s.t. \quad \boldsymbol{G}' = \boldsymbol{F}'\boldsymbol{A} \end{aligned} \tag{17}$$

It is noticed that the dimension of $\boldsymbol{A}$ contains some redundant variables, which not only increase time complexity but also incur data conflict.

Hence, in the following, we devise a transformation matrix to remove these redundant variables. Specifically, we first define a

new connection vector $\boldsymbol{A}'$ as follows,

$$\begin{aligned} \boldsymbol{A}' &= \left(\boldsymbol{A}'_1, \boldsymbol{A}'_2, \ldots, \boldsymbol{A}'_n, \ldots, \boldsymbol{A}'_N\right), \\ \boldsymbol{A}'_n &= (a_{n,n+1}, a_{n,n+2}, \ldots, a_{n,N}), \end{aligned} \tag{18}$$

where the matrix $\boldsymbol{A}'$ is a $\frac{N(N-1)}{2}$-dimensional vector. The functionality of the devised transformation matrix, denoted by $\boldsymbol{C}$, is to build a connection between $\boldsymbol{A}'$ and $\boldsymbol{A}$, i.e., $\boldsymbol{A} = \boldsymbol{C}\boldsymbol{A}'$. Particularly, $\boldsymbol{C}$ is a $N(N-1) \times \frac{N(N-1)}{2}$-dimensional matrix. Before that, we first formulate an $N \times N$ symmetric matrix, denoted by $\boldsymbol{B} = (b_{ij})$, where the upper triangular element b_{ij}, $j > i$, is computed as follows.

$$b_{ij} = \begin{cases} 1 & if\ i = 1, j = 2 \\ b_{i-1,N} + 1 & if\ i > 1, j = i + 1 \\ b_{i,i+1} + j - i - 1 & if\ j > i + 1 \\ 0, & \text{otherwise} \end{cases}. \tag{19}$$

A vector $\boldsymbol{B}'$ is computed by removing the diagonal elements from the matrix $\boldsymbol{B}$ and vectoring the rest elements, which is formulated as follows,

$$\boldsymbol{B}' = (b_{12}, \ldots, b_{1N}, b_{21}, \ldots, b_{N1}, \ldots, b_{N,N-1}), \tag{20}$$

where the maximum value of elements is equal to $N(N-1)/2$. By utilizing the vector $\boldsymbol{B}'$, each element $c_{ij} \in \boldsymbol{C}$, is computed as follows.

$$c_{ij} = \begin{cases} 1, & if\ j == b'_i, b'_i \in \boldsymbol{B}', \\ 0, & \text{otherwise}. \end{cases} \tag{21}$$

Here $\forall i \in [1, N(N-1)]$, $\forall j \in [1, N(N-1)/2]$. Particularly, c_{ij} indicates whether to transform the j^{th} element in $\boldsymbol{A}'$ to the i^{th} element in $\boldsymbol{A}$ or not.

By replacing $\boldsymbol{A}$ with $\boldsymbol{A}'$, then the global convex optimization model in Eq. (17) is formally presented as follows.

$$\begin{aligned} &\min \quad ||\boldsymbol{A}'||_1 \\ &s.t. \quad \boldsymbol{G}' = \boldsymbol{F}'\boldsymbol{C}\boldsymbol{A}' \end{aligned} \tag{22}$$

Thus, the network reconstruction problem can be simply solved by applying convex optimization tools to achieve the optimal solution $\boldsymbol{A}^*$of Eq. (22).

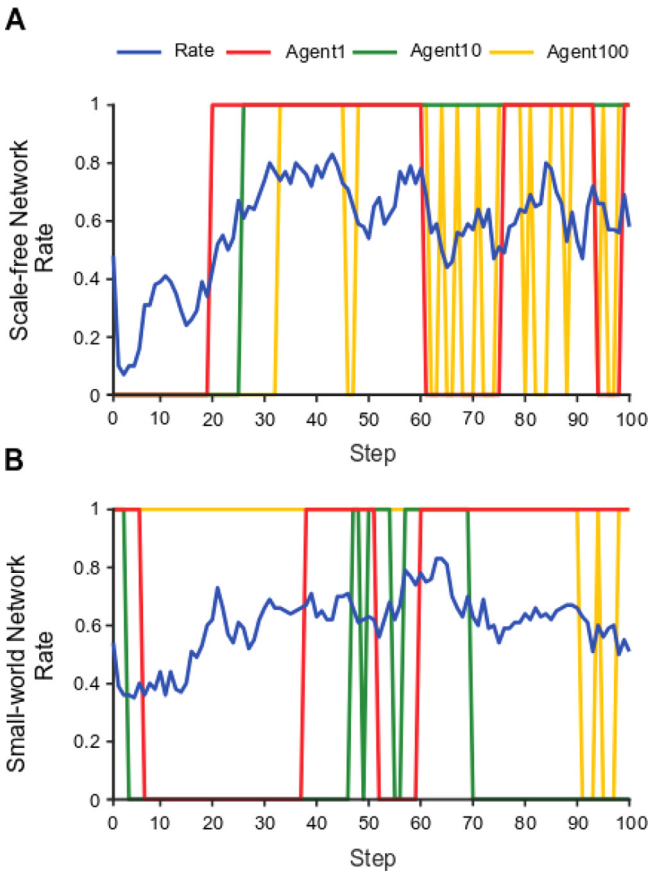

Fig. 1. The time series of the strategies of different agents in a scale-free network (A) and a small-world network (B). Red, green, yellow line denotes three individual agents and the blue line is the average cooperation rate of all 100 agents. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4. Simulation results

In this section, we first verify the effectiveness of the proposed method on two classical types of complex networks: a scale-free network [22] and a small world network [23]. Then we test the robustness of our method by taking into account the measurement noise.

4.1. Reconstruct complex networks without noise

We first generate a scale-free network of 100 nodes and a small-world network of 100 nodes respectively according to the methods in Santos and Pacheco [22], Watts and Strogatz [23], and then record the adjacency matrices, agent strategies and fitness during the evolutionary process. Here the adjacency matrices are the true value used for evaluating the accuracy of the proposed method and agent strategies and fitness are used for reconstructing the network. Fig. 1 illustrates the time series of the strategies for three typical agents, as well as the average cooperation rate of all 100 agents. It can be observed that the network hasn't reached an equilibrium state for the 100 time step. Accordingly, we can use the recorded statistics to reconstruct the network.

Since the values in the adjacency matrix are all continuous values, we adopt a fixed threshold 0.1 to determine whether the link exists. This threshold is also used in paper [16]. Specifically, we assume that the link exists when a_{ij} is greater than 0.1, otherwise it does not exist. We introduce two evaluation metrics, the success rates of existent links (SREL) and nonexistent links (SRNL), to quantify the performance of our method. Here SREL is defined as the ratio of the number of successfully inferred connections to all the existing connections, and SRNL is defined as the ratio of the number of successfully inferred non-connections to all non-connections in the adjacency matrices. We compare our method with the compressive sensing method (CST) [7,10] and the conflict-based method (CBM) [16]. The CST method performs convex optimization to reconstruct the network node by node, whereas the CBM method incorporates latent constraints to enhance inference of network structure. The results are shown in Figs. 2 and 3. Here the x-coordinate is the data ratio. It's defined as the ratio of the number of measurements to the number of agents, which indicates how much data we need to reconstruct the network. It can be seen that the proposed method (red circle) is much better than CST (green triangular) and CBM (blue square). When the data ratio is 0.1, our method can achieve good performance. SREL = 0.99 and SRNL = 0.987 for a scale-free network, and SREL = 0.997 and SRNL = 0.981 for a small-world network.

We also evaluate the performance of our method with the precision-recall (PR) curve and the receiver operating characteristic (ROC) curve [24]. Specifically, PR curves summarize the trade-off between the true positive rate and the positive predictive value for a model by choosing different probability thresholds, whereas ROC curves summarize the trade-off between the true positive rate and false positive rate for a model by choosing different thresholds. The area under the PR curve (AUPR) and the area under the ROC curve (AUROC) are used as aggregate indicators of the algorithm performance. Note that the larger AUPR pushes the curve

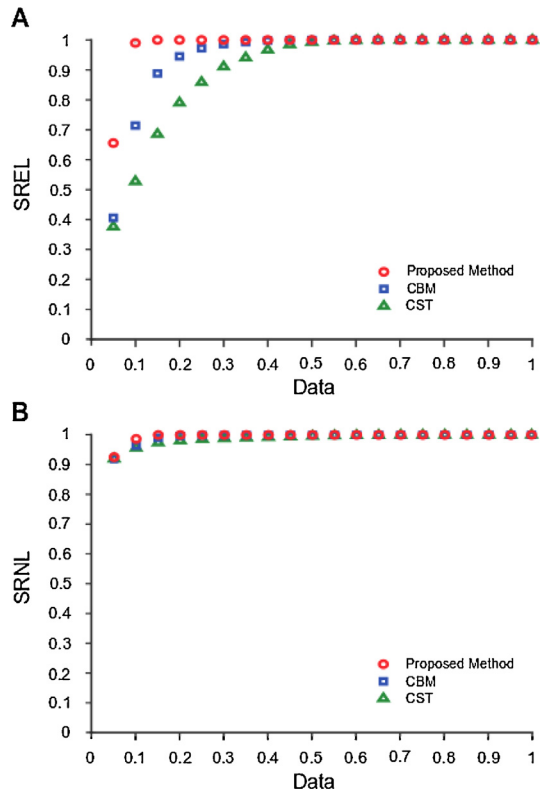

Fig. 2. SREL (A) and SRNL (B) curves measuring the performance of our methods in a scale-free network. Network size $N = 100$. Each data point is obtained by averaging over 20 network realizations.

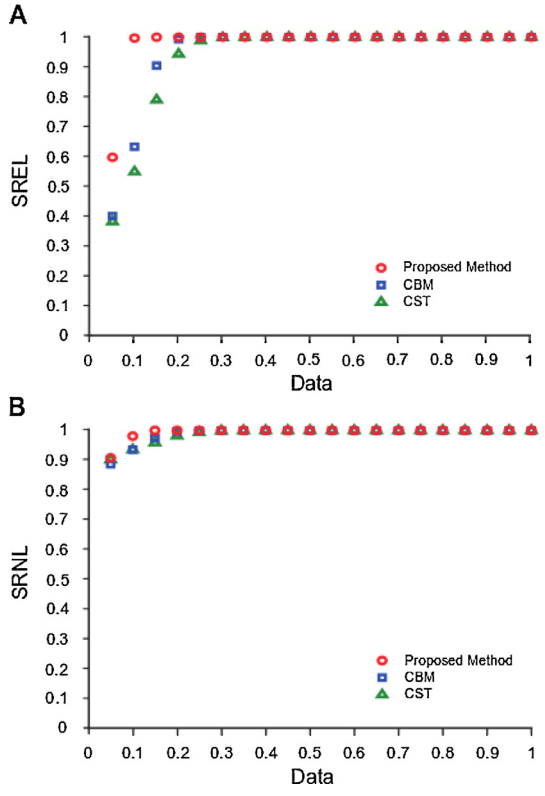

Fig. 3. SREL (A) and SRNL (B) curves measuring the performance of our methods in a small-world network. Network size $N = 100$. Each data point is obtained by averaging over 20 network realizations.

Table 1
AUPR measuring the performance of three different methods in a scale-free network and a small-world network.

	Scale free network		Small-world network	
	$R = 0.1$	$R = 0.2$	$R = 0.1$	$R = 0.2$
CST	0.33	0.9625	0.4276	0.7771
CBM	0.4011	0.995	0.5896	0.9682
Our method	**0.9974**	**1**	**0.9975**	**0.9999**

Table 2
AUROC measuring the performance of three different methods in a scale-free network and a small-world network.

	Scale free network		Small-world network	
	$R = 0.1$	$= 0.2$	$R = 0.1$	$R = 0.2$
CST	0.8105	0.9918	0.8205	0.9271
CBM	0.8255	0.9974	0.8808	0.9578
Our method	**0.9999**	**1**	**0.9999**	**1**

towards the upper-right-hand corner and the larger AUROC moves the curve towards the upper-left-hand corner. We calculate the recall, precision, FPR and TPR at 3000 thresholds of equal interval between -1 and 2. Based on this, PR and ROC curves are drawn and the corresponding AUPR and AUROC are calculated. Figs. 4 and 5 compare the PR and ROC curves of our method with CST and CBM. It is found the PR curve and the ROC curve of our method (red line) are much higher than those of CST (green line) and CBM (blue line), which indicates our method can achieve better infer precision of the existent and non-existent connections than CST and CBM under varying predefined threshold. One also can get the same conclusion from AUPR and AURoC in Tables 1 and 2. As shown, our method is far more accurate and efficient than CST [7,10] and CBM [16], which is reflected in an increase of AUPR by 148.67% for a scale-free network and 69.18% for a small-world network, and an increase of AUROC by 21.13% for a scale-free network and 13.52% for a small-world network compared with CBM, even when the measured data is insufficient (the data rate is 0.1). Particularly, AUPR and AUROC of our method has already achieved 1 in both scale-free and small-world networks when the data rate is only 0.2. Therefore, this set of simulation results demonstrate that the proposed method is a very powerful means of reconstructing the network structure.

Figs. 6 and 7 illustrate the original reconstructed values of each possible edge in a scale-free network and a small-world network respectively. For enough data, e.g., $R = 0.4$, there is a vast and clear gap between existent connections and non-existent connections for both methods, which indicates that both two methods can accurately infer the connection states. For insufficient data, e.g., $R = 0.1$, the links are difficult to identify with CST due to the mixture of reconstructed values whereas there is still a clear gap of reconstructed values with our method, ensuring accurate reconstruction.

In order to analyze how the date rate affect the accuracy, here we plot the PR curve and ROC curve with data rate 0.05 (red line) and 0.1 (blue line). The results are shown in Fig. 8, one can see that the reconstruction accuracy increase with the increase of data rate. When the data rate equals 0.1, both AUPR and AUROC achieve 1. Thus our method only needs small amount of data to reconstruct the network.

4.2. Reconstruct complex networks with noise

In practice, the observed measurements of the network are often contaminated with noise. We test the robustness of the proposed method against varying amplitudes of noise. Specifically, the gaussian noise is directly added to the payoff vector of node (i.e.,

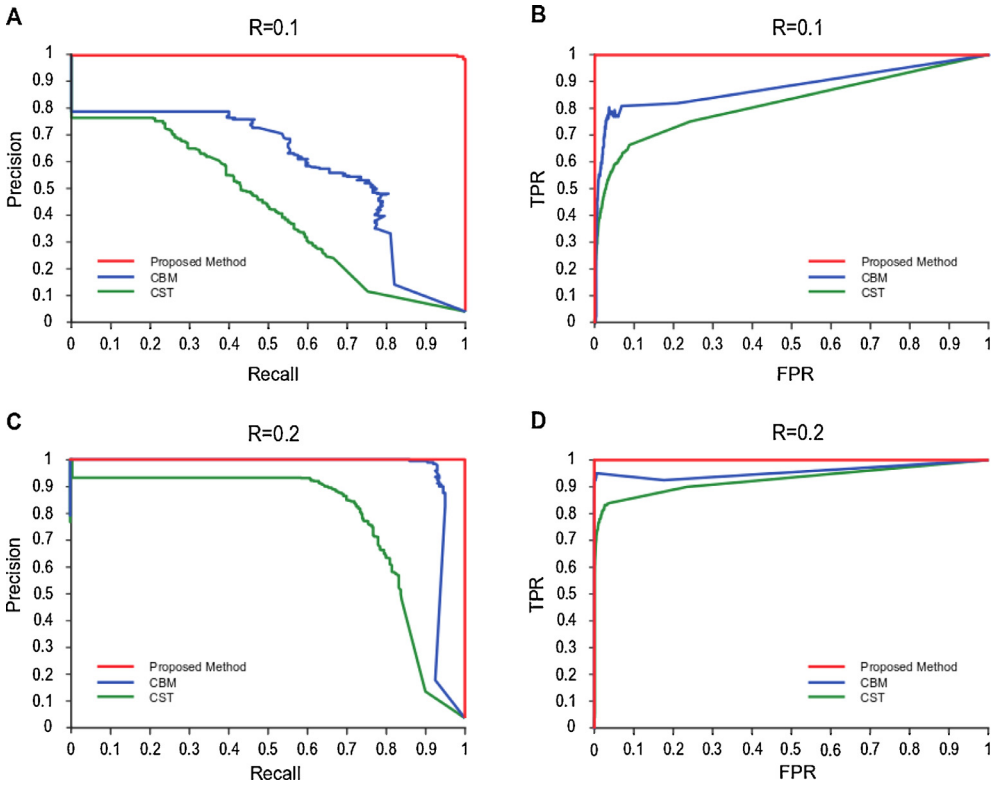

Fig. 4. PR curve (A) and ROC curve (B) measuring the performance of our methods in a scale-free network. Network size $N = 100$. Date rate $R = 0.1$. (C) and (D) are the same as (A) and (B), but with the data rate $R = 0.2$.

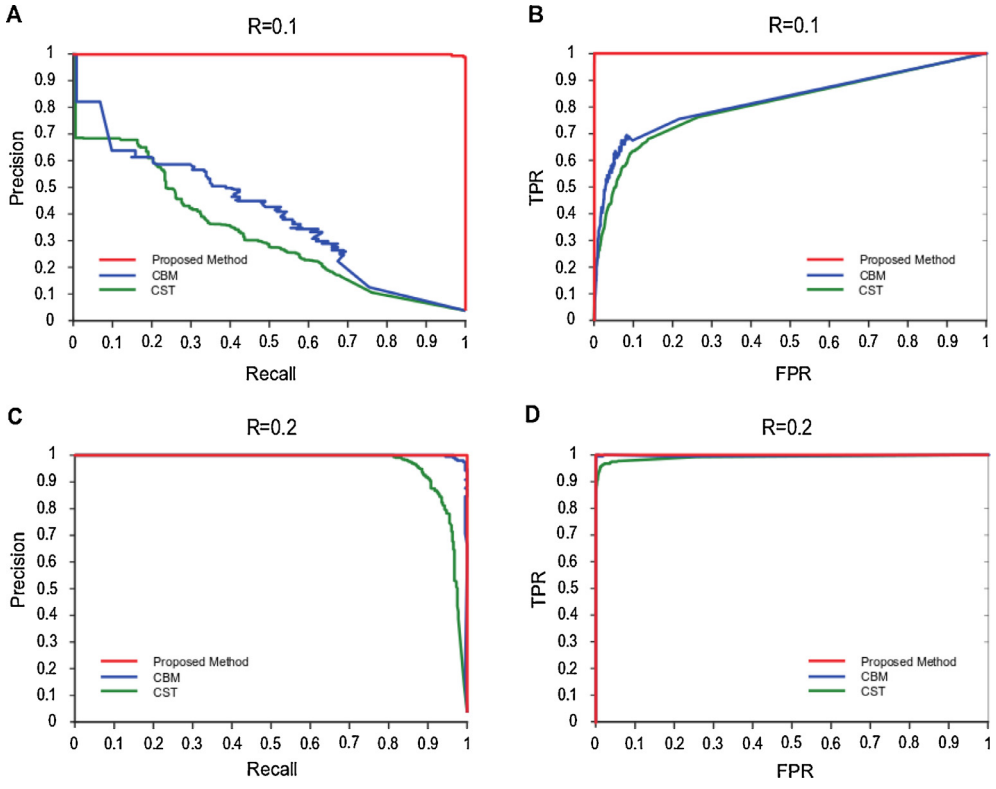

Fig. 5. PR curve (A) and ROC curve (B) measuring the performance of our methods in a small-world network. Network size $N = 100$. Date rate $R = 0.1$. (C) and (D) are the same as (A) and (B), but with the data rate $R = 0.2$.

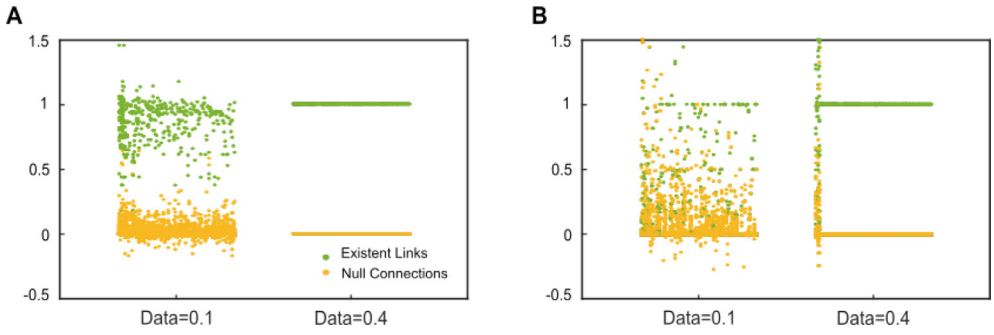

Fig. 6. The reconstructed values of elements in $\boldsymbol{A}'$ with the proposed method (A) and CST (B) for a scale-free network.

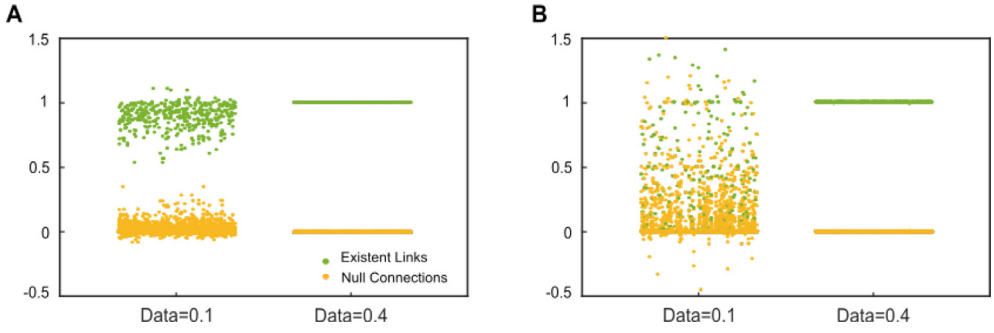

Fig. 7. The reconstructed values of elements in $\boldsymbol{A}'$ with the proposed method (A) and CST (B) for a small-world network.

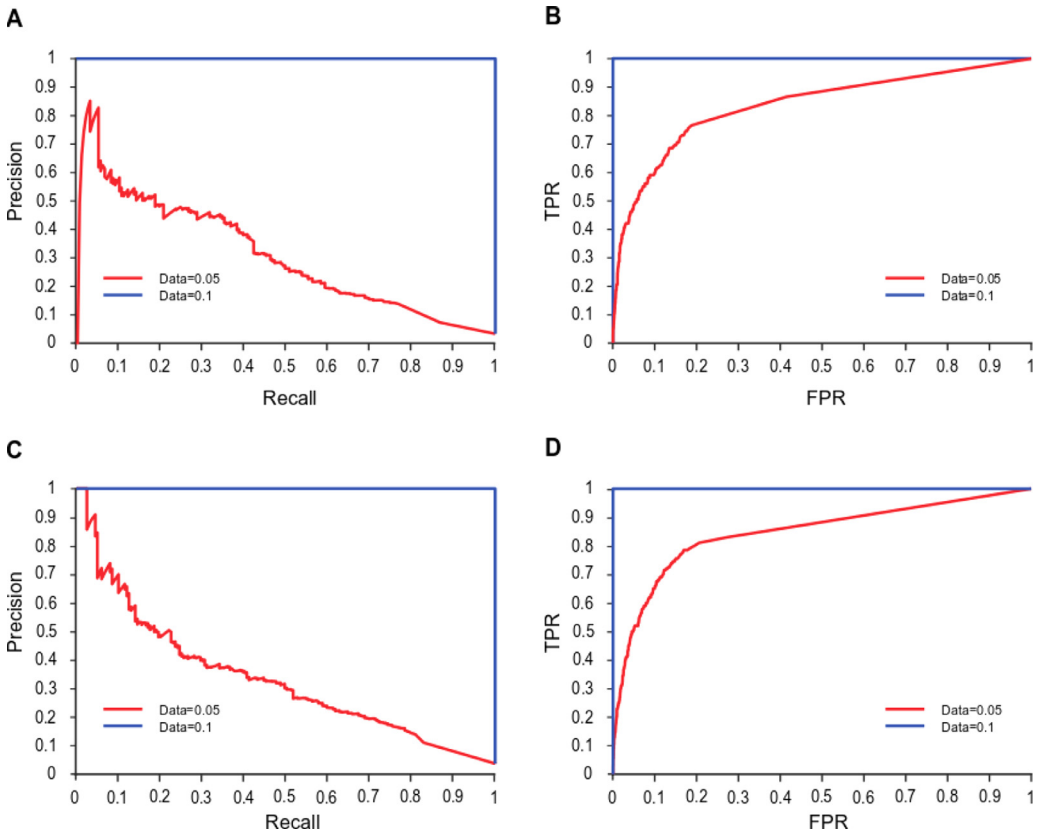

Fig. 8. PR and ROC curves for structure identification performance in a scale-free network (A-B) and a small-world network (C-D).

G_n), where the noise amplitude increases from 0.5 to 9 with interval 0.5. Fig. 11 shows average ratio of absolute noise to gain of node under scale-free and small-world networks. According to Fig. 11, the noise ratio increases from 12.7% to 253.0% in scale-free networks. As shown in Figs. 9 and 10, the SREL of all the methods decreases dramatically at first and then slows down when the noise amplitude exceeds 4. It is because the average ratio of absolute noise to gain is higher than 50% with noise amplitude of 4, which indicates a high-level noise and randomizes the gain value of each node. It is verified that the SREL of all the methods are less than 40%. However, our method still achieves the best SREL. Further, it is noted that our method achieves relative lower SRNL

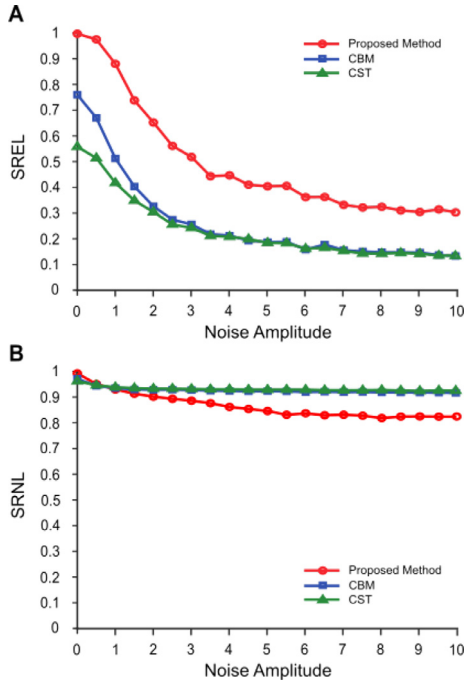

Fig. 9. Effect of noise amplitude on SREL (A) and SRNL (B) in a scale-free network. Here we used the network from Fig. 2. The noise is directly added to the payoff value of node. Each data point is obtained by averaging over 20 network realizations.

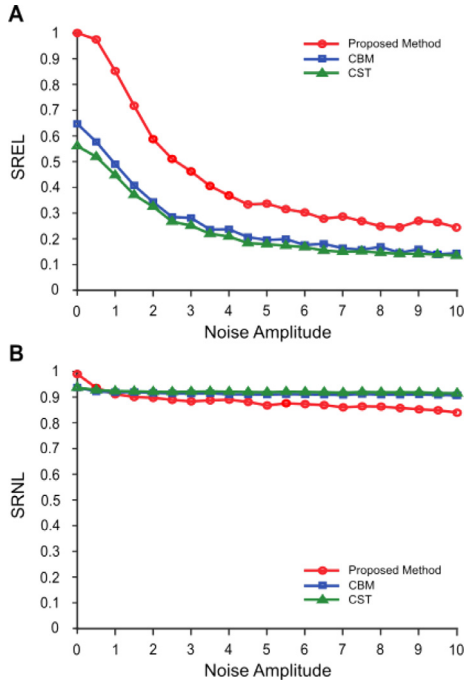

Fig. 10. Effect of noise amplitude on SREL (A) and SRNL (B) in a small-world network. Here we used the network from Fig. 3. The noise is directly added to the payoff value of node. Each data point is obtained by averaging over 20 network realizations.

than other two methods but the performance gap is close. Particularly, the SRNL of three methods still achieves higher than 0.8. It is because that most links are non-existent due to the sparsity of the networks.

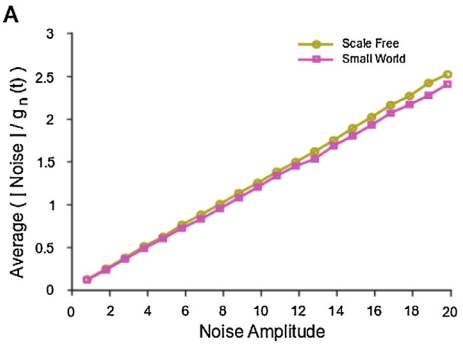

Fig. 11. The ratio of absolute noise to payoff value of node under scale-free and small-world networks. Here we used the network from Figs. 2 and 3. The noise is directly added to the payoff value of node. Each data point is obtained by averaging over 20 network realizations.

5. Conclusion

In this paper, we proposed a new convex optimization framework of compressed sensing-based network reconstruction in complex networks with insufficient measured data. To be specific, a global convex optimization model is formulated to derive the connection states of all nodes in the network, which eliminates data conflict and improves system performance. We conduct the experiments under two classical complex networks, i.e., the scale-free and small world. The comprehensive results demonstrate the accuracy and robustness of the proposed methods compared with two competitive algorithms under limited measured data and varying amplitudes of noise.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Zihua Hang: Investigation, Conceptualization, Methodology, Software. **Penglin Dai:** Methodology, Writing - original draft, Writing - review & editing, Supervision. **Shanshan Jia:** Software, Validation, Data curation. **Zhaofei Yu:** Methodology, Writing - original draft, Writing - review & editing.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grants nos.61802319, 61806011); the National Post-Doctoral Program for Innovative Talents under grant BX20180005; China Postdoctoral Science Foundation funded project (Nos. 2019M660245 and 2020T130547); Sichuan Science and Technology Program (No. 2020YJ0272).

References

[1] Zhu L, Wang Y. Rumor spreading model with noise interference in complex social networks. Phys A 2017;469:750–60.
[2] Dai P, Liu K, Wu X, Yu Z, Xing H, Lee VCS. Cooperative temporal data dissemination in SDN-based heterogeneous vehicular networks. IEEE Internet Things J 2019;6(1):72–83.
[3] Xia D, Ke H, Zuo M, Ye J, Min Q, Wang Z. Researches on modeling learners' collaboration network in virtual learning community. In: 2018 9th International conference on information technology in medicine and education (ITME); 2018. p. 596–600.
[4] Wu L, Li M, Wang J, Wu F-X. The minimum steering node set of complex networks and its applications to biomolecular networks. IET Syst Biol 2016;10(7):116–23.

[5] Pandey PK, Badarla V. Reconstruction of network topology using status–time-series data. Phys A 2018;490:573–83.
[6] Shen Z, Wang W-X, Fan Y, Di Z, Lai Y-C. Reconstructing propagation networks with natural diversity and identifying hidden sources. Nat Commun 2014;5.
[7] Wang W-X, Lai Y-C, Grebogi C, Ye J. Network reconstruction based on evolutionary-game data via compressive sensing. Phys Rev X 2011;1(2):021021.
[8] Barranca VJ, Zhou D, Cai D. Compressive sensing reconstruction of feed-forward connectivity in pulse-coupled nonlinear networks. Phys Rev E 2016;93(6).
[9] Barranca VJ, Zhou D. Compressive sensing inference of neuronal network connectivity in balanced neuronal dynamics. Front Neurosci 2019;13:1101.
[10] Han X, Shen Z, Wang W-X, Di Z. Robust reconstruction of complex networks from sparse data. Phys Rev Lett 2015;114(2):028701.
[11] Huang K, Wang Z, Jusup M. Incorporating latent constraints to enhance inference of network structure. IEEE Trans Netw Sci Eng 2020;7(1):466–75.
[12] Wang W-X, Yang R, Lai Y-C, Kovanis V, Grebogi C. Predicting catastrophes in nonlinear dynamical systems by compressive sensing. Phys Rev Lett 2011;106:154101.
[13] Jeong W, Yu U. Prisoner's dilemma game on complex networks with a death process: effects of minimum requirements and immigration. Phys A 2019;517:47–52.
[14] Wang W-X, Lai Y-C, Grebogi C, Ye J. Network reconstruction based on evolutionary-game data via compressive sensing. Phys Rev X 2011;1:021021.
[15] Han X, Shen Z, Wang W-X, Di Z. Robust reconstruction of complex networks from sparse data. Phys Rev Lett 2015;114(2).
[16] Ma L, Han X, Shen Z, Wang W-X, Di Z. Efficient reconstruction of heterogeneous networks from time series via compressed sensing. PLoS One 2015;10(11).
[17] Nowak M, May R. Evolutionary games and spatical chaos. Nature 1992;359(6398):826–9.
[18] Santos FC, Santos MD, Pacheco JM. Social diversity promotes the emergence of cooperation in public goods games. Nature 2008;454(7201):213–U49.
[19] Santos F, Pacheco J. Scale-free networks provide a unifying framework for the emergence of cooperation. Phys Rev Lett 2005;95:098104.
[20] Allen-Zhu Z, Gelashvili R, Razenshteyn I. Restricted isometry property for general p-norms. IEEE Trans Inf Theory 2016;62(10):5839–54.
[21] Candes EJ, Tao T. Near-optimal signal recovery from random projections: universal encoding strategies? IEEE Trans Inf Theory 2006;52(12):5406–25.
[22] Santos FC, Pacheco JM. Scale-free networks provide a unifying framework for the emergence of cooperation. Phys Rev Lett 2005;95(9):098104.
[23] Watts DJ, Strogatz SH. Collective dynamics of 'small-world' networks. Nature 1998;393(6684):440.
[24] Davis J, Goadrich M. The relationship between precision-recall and ROC curves. In: Proceedings of the 23rd international conference on machine learning. ACM; 2006. p. 233–40.