A Learning Convolutional Neural Network Approach for Network Robustness Prediction

Yang Lou, *Member, IEEE*, Ruizi Wu, Junli Li, Lin Wang, *Senior Member, IEEE*, Xiang Li, *Senior Member, IEEE*, and Guanrong Chen, *Life Fellow, IEEE*

***Abstract*—Network robustness is critical for various societal and industrial networks against malicious attacks. In particular, connectivity robustness and controllability robustness reflect how well a networked system can maintain its connectedness and controllability against destructive attacks, which can be quantified by a sequence of values that record the remaining connectivity and controllability of the network after a sequence of node- or edge-removal attacks. Traditionally, robustness is determined by attack simulations, which are computationally very time-consuming or even practically infeasible for large-scale networks. In this article, an improved method for network robustness prediction is developed based on learning feature representation using the convolutional neural network (LFR-CNN). In this scheme, the higher-dimensional network data are compressed into lower-dimensional representations, which are then passed to a convolutional neural network to perform robustness prediction. Extensive experimental studies on both synthetic and real-world networks, both directed and undirected, demonstrate that: 1) the proposed LFR-CNN performs better than other two state-of-the-art prediction methods, with significantly smaller prediction errors; 2) LFR-CNN is insensitive to the variation of the input network size, which significantly extends its applicability; 3) although LFR-CNN needs more time to perform feature learning, it can achieve accurate prediction faster than attack simulations; and 4) LFR-CNN not only accurately predicts the network robustness, but also provides a good indicator for connectivity robustness, better than the classical spectral measures.**

***Index Terms*—Complex network, convolutional neural network (CNN), graph representation learning, prediction, robustness.**

I. Introduction

A COMPLEX network can be described by a graph consisting of large numbers of nodes and edges with complicated connection topologies. Many natural and engineering systems can be modeled as complex networks, and then analyzed by using graph theory and network analysis tools. The study of complex networks attracts increasing interest from research communities in various scientific and technological fields, including computer science, systems engineering, applied mathematics, statistical physics, biological sciences, and social sciences [1], [2], [3], [4].

In the pursuit of networked systems control for beneficial applications, the *network controllability* [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] is a fundamental issue, which refers to the ability of a network of interconnected dynamical systems in changing from any initial state to any desired state under feasible control input within finite time [17]. The *network connectivity*, on the other hand, is fundamentally important for a network to function, affecting particularly the network controllability [17] and synchronizability [19]. It is easy to see that good controllability requires good connectivity, but good connectivity does not necessarily guarantee good controllability [20]. In fact, network connectivity and controllability have very different characteristics and measures: the former is guaranteed by a sufficient number of edges, while the later further requires a proper organization of these edges.

Today, malicious attacks and random failures widely exist in engineering and technological facilities and processes, which degrade or even destroy certain network functions typically through destructing the network connectivity. Therefore, it is essential to strengthen the network connectivity against such attacks and failures [20], [21], [22], [23], [24], [25], [26]. In general, destructive attacks and failures take place in the forms of node- and edge-removals, which may cause significant degeneration of the network connectivity and controllability. In such situations, the abilities of a network to regain and maintain its connectivity and controllability against attacks or failures are of great concern, which is referred to

Manuscript received 20 June 2022; revised 28 August 2022; accepted 13 September 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62002249 and Grant 61873167; in part by the Foundation of Key Laboratory of System Control and Information Processing, Ministry of Education, China, under Grant Scip202103; in part by the Hong Kong Research Grants Council through GRF under Grant CityU11206320; and in part by the Fundamental Research Funds for the Central Universities. This article was recommended by Associate Editor H. Wang. *(Yang Lou and Ruizi Wu contributed equally to this work.) (Corresponding authors: Yang Lou; Lin Wang.)*

Yang Lou is with the College of Computer Science, Sichuan Normal University, Chengdu 610066, China, and also with the Department of Computing and Decision Sciences, Lingnan University, Hong Kong, China (e-mail: felix.lou@ieee.org).

Ruizi Wu and Junli Li are with the College of Computer Science, Sichuan Normal University, Chengdu 610066, China (e-mail: vridge@foxmail.com; lijunli@sicnu.edu.cn).

Lin Wang is with the Department of Automation and the Key Laboratory of System Control and Information Processing, Ministry of Education, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: wanglin@sjtu.edu.cn).

Xiang Li is with the Institute of Complex Networks and Intelligent Systems, Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University, Shanghai 201210, China, and also with the Department of Control Science and Engineering, Tongji University, Shanghai 200240, China (e-mail: lix2021@tongji.edu.cn).

Guanrong Chen is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China (e-mail: eegchen@cityu.edu.hk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2022.3207878.

Digital Object Identifier 10.1109/TCYB.2022.3207878

as the *connectivity robustness* and *controllability robustness*, respectively. The subject of connectivity and controllability robustness is fundamentally and practically important, which have been investigated in, and applied to the fields of nervous systems [27], wireless sensor networks [28], power grids [29], transportation networks [30], [31], and so on.

Connectivity robustness is commonly measured by using the change of the portion of nodes in the largest connected component (LCC) [23] that survives from a sequence of attacks. A network is deemed more robust against attacks if it can always maintain higher values of the fractions of LCC nodes throughout an attack process. The investigation and optimization of connectivity robustness using this measure emphasize on protecting the LCC. Given certain practical constraints, such as node degree preservation, connectivity robustness can be enhanced by edge rewiring, which imposes disturbances onto the network structure [32], [33], [34], [35], [36], [37], [38], [39]. After some edge rewiring operations, whether or not such disturbance enhances the robustness has to be evaluated, typically by using time-consuming attack simulations. As a remedy, several easy-to-access indicators, for example, assortativity [40] and spectral measures [36], [41], are adopted for estimating the network connectivity robustness. It is found that onion-like structured heterogeneous networks with positive assortativity coefficients are robust against attacks [42], [43], [44]. However, these measures have limited scopes of applications, and therefore, the time-consuming attack simulation remains as the main approach today.

Controllability robustness is generally measured using the change of the density of driver nodes, at which external control signals can be imposed as input. A network is deemed more robust against attacks if it can maintain a lower density of driver nodes throughout an attack process. The optimization of controllability robustness emphasizes on maintaining a low demand of additional driver nodes. Although controllability robustness can be enhanced by edge rewiring as in connectivity robustness enhancement, their objective functions are very different. In fact, on top of the connectedness, the way the nodes are connected makes a huge impact on the resulting network controllability. For example, it is observed that a power-law degree distribution does not necessarily imply weak controllability robustness; while multichain [45] and multiloop [46] structures significantly strengthen the controllability robustness. Moreover, it is empirically found that extreme homogeneity is necessary for the optimal topology that has the best controllability robustness against random node attacks [47]. Likewise, attack simulation is a main approach to measuring the network controllability robustness today, which however is even more time-consuming than measuring the network connectivity discussed above.

For both connectivity and controllability robustness enhancements, deep neural networks [48], [49], [50], [51] provide a useful tool for computation, optimization, and analysis. Successful deep learning applications on complex networks include network robustness prediction using convolutional neural networks (CNNs) [20], [52], [53], and critical node identification using deep reinforcement learning [25] and graph attention networks [26]. Main advantages of CNN-based approaches for robustness prediction include: 1) the method is straightforward, where the adjacency matrix of the network is treated as a gray-scale image, and then, the classification (if any) and regression tasks are the same as in conventional image processing and 2) the performance of CNN-based approach is stable and reliable: all types of networks are acceptable as input, which is also shift-invariant [54], namely, shuffling and transposing pixels of an image (while keeping the network topology unchanged) does not degrade the performance of the prediction [53].

However, the CNN-based approach cannot guarantee the prediction performance when the input size has significant changes from the training samples. Empirical studies suggest that it is tolerable to a ±3.0% size difference for real-world networks [53], and ±7.3% difference for synthetic networks [55]. In addition, since many complex networks are sparse, the gray-scale images converted from network adjacency matrices typically contain a large amount of useless information, where quite a lot of pixels can be removed or compressed.

To overcome the aforementioned issues, a learning feature representation-based CNN (LFR-CNN) approach is proposed in this article. The contributions of this work are summarized as follows.

1) LFR-CNN is proposed for precise network robustness prediction, which consists of an LFR module and a CNN. The LFR module not only compresses the higher-dimensional complex network data into lower-dimensional representations but also enhances the learning ability of the predictor.
2) The proposed LFR-CNN not only overcomes the issues of different input sizes but more importantly also achieves a balance in the structure complexity and the magnitude of the number of parameters for two state-of-the-art network robustness predictors [52], [56].
3) Extensive experimental studies demonstrate that the proposed LFR-CNN outperforms two state-of-the-art predictors [52], [56], and provides a good indicator for connectivity robustness, better than the classical spectral measures.

It is worth mentioning that although only connectivity robustness and controllability robustness are considered in this article, the proposed LFR-CNN can also predict other network robustness performances, such as biological robustness, communication robustness, synchronizability, and some other network functions.

The remainder of this article is organized as follows. Section II reviews the measures of network connectivity and controllability robustness against destructive node-removal attacks. Section III introduces the proposed LRF-CNN in detail. Section IV presents experimental results with analysis and comparison. Section V concludes the investigation.

II. Robustness of Complex Networks

The concepts and calculations of connectivity robustness and controllability robustness are introduced in this section, where connectivity robustness reflects how well a networked system can maintain its connectedness under a sequence of node-removal attacks, while controllability robustness reflects how well it can maintain its controllable state. In this article, only node-removal attacks are investigated, while edge-removal attacks can be studied in a similar manner.

A. Connectivity Robustness

An undirected network is connected if and only if for each pair of nodes there is a path between them. A directed network is *weakly connected* if it remains to be connected after all the directions are removed. Both *connectedness* and *weak connectedness* are employed as measures of the network connectivity in this article, for undirected and directed networks, respectively.

Under a sequence of node-removal attacks, the connectivity robustness is evaluated using the fraction of nodes in the LCC after each node-removal [23], as follows:

$$p(i) = \frac{N_{\text{LCC}}(i)}{N-i}, \quad i = 0, 1, \ldots, N-1 \tag{1}$$

where $p(i)$ and $N_{\text{LCC}}(i)$ are the fraction and number of nodes in LCC after totally i nodes have been removed, respectively; N is the number of nodes in the network before being attacked. When these values are plotted versus the fraction of removed nodes, a curve is obtained, called the *connectivity curve*.

B. Controllability Robustness

For a linear time-invariant networked system $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$, where $\mathbf{A}$ and $\mathbf{B}$ are constant matrices of compatible dimensions, and $\mathbf{x}$ and $\mathbf{u}$ are the state vector and control input, respectively. The system is *state controllable* if and only if the controllability matrix $[\mathbf{B}\ \mathbf{AB}\ \mathbf{A}^2\mathbf{B}\ \cdots \mathbf{A}^{N-1}\mathbf{B}]$ has a full row-rank, where N is the dimension of $\mathbf{A}$, which is also the size of the network in the present study. For a directed network, identifying the set of the minimum number of driver nodes N_D can be converted to searching for a maximum matching of the network [5]: $N_D = \max\{1, N - |E^*|\}$, where $|E^*|$ is the number of edges in the maximum matching E^*. For an undirected network, the minimum number of needed driver nodes can be calculated using the exact controllability formula [6]: $N_D = \max\{1, N - \text{rank}(\mathbf{A})\}$. Then, the network controllability robustness is calculated as follows:

$$q(i) = \frac{N_D(i)}{N-i}, \quad i = 0, 1, \ldots, N-1 \tag{2}$$

where $N_D(i)$ is the minimum number of driver nodes needed to retain the network controllability after a total of i nodes have been removed. When these values are plotted, a curve is obtained, called the *controllability curve*.

C. Error Measures

For either connectivity or controllability, consider three curves: $\mathbf{s}_t = [s_t(0), \ldots, s_t(N-1)]$ denotes the true curve obtained by attack simulations, and $\mathbf{s}_1 = [s_1(0), \ldots, s_1(N-1)]$ and $\mathbf{s}_2 = [s_2(0), \ldots, s_2(N-1)]$ denote two predicted curves, respectively. The difference between the true curve and a predicted curve is calculated by $\boldsymbol{\xi}_\alpha = |\mathbf{s}_t - \mathbf{s}_\alpha|$, where $\boldsymbol{\xi}_\alpha = [\xi_\alpha(0), \ldots, \xi_\alpha(N-1)]$ is the sequence of errors between the two curves, where $\xi_\alpha(i) = |s_t(i) - s_\alpha(i)|$, for $\alpha = 1$ or 2, and $i = 0, 1, \ldots, N-1$.

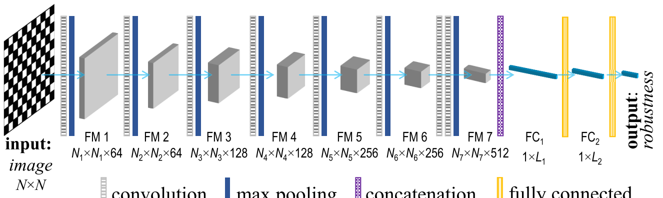


Fig. 1. CNN structure of PCR. The input is adjacency matrix; the output is an N-vector. For $N = 1000$, seven FM groups are generated with $N_i = \lceil N/2^{(i+1)} \rceil$, for $i = 1, 2, \ldots, 7$. Concatenation layer reshapes the data to a vector, from FM 7 to FC_1. $\text{FC}_1 = 512N_7^2$ and $\text{FC}_2 = 4096$ [52].

The *prediction error* $\bar{\xi}_\alpha$ is then calculated by

$$\bar{\xi}_\alpha = \frac{1}{N}\sum_{i=0}^{N-1} \xi_\alpha(i). \tag{3}$$

The vector $\boldsymbol{\xi}_\alpha$ can be used to visualize the prediction errors throughout the attack process. The scalar $\bar{\xi}_\alpha$ measures the *overall* prediction error, that is, $\bar{\xi}_1 < \bar{\xi}_2$ means that the predicted curve $\mathbf{s}_1$ obtains lower prediction error than $\mathbf{s}_2$.

For notational convenience, the integer index sequence $i = 0, 1, \ldots, N-1$, will be replaced by the fractional index sequence $\delta = 0, (1/N), \ldots, ((N-1)/N)$, thereby, equivalently replacing $p(i)$ and $q(i)$, with $p(\delta)$ and $q(\delta)$.

III. Performance Predictor

This section briefly reviews the predictor for controllability robustness (PCR) [52] that employs a VGG-structured CNN [57], and PATCHY-SAN [56] that consists of an LFR-based 1D-CNN. Pros and cons of these two approaches are discussed. Then, a structural LFR-CNN is designed by incorporating the LFR module and a simplified VGG-structured CNN. LFR-CNN has a parameter magnitude significantly greater than PATCHY-SAN, but less than PCR.

A. Convolutional Neural Network

PCR is a CNN-based framework for predicting the controllability robustness [52], which has also been applied to predict connectivity robustness [20]. The CNN structure of PCR is shown in Fig. 1. Network adjacency matrices are converted to gray-scale images and then used directly as input to CNN. Both classification and regression tasks can be performed using such an image-processing mechanism. Due to a sufficiently large source of training data that can be generated using various synthetic network models, tens of millions of internal parameters in a CNN can be properly trained.

The mean-squared error between the predicted curve $\hat{v}$ and true curve v is used as the loss function

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} ||\hat{v}(i) - v(i)|| \tag{4}$$

where $\hat{v}(i)$ represents the predicted connectivity or controllability value when a total number of i nodes has been removed from the network; $v(i)$ represents the corresponding true value obtained by attack simulation; $||\cdot||$ is the Euclidean norm. The training process aims at adjusting the internal parameters [48], with the objective of minimizing $\mathcal{L}$.

B. PATCHY-SAN

Complex network data have distinguished continuous and discrete attributes that are different from general image data. A group of recurrent neural networks, namely, the graph neural networks (GNNs) [58], [59], [60], are specifically designed for processing graph data. Specifically, lower-dimensional representations are generated from compacting higher-dimensional raw graph data, and then, classification and regression tasks are performed by processing the lower-dimensional representation data. PATCHY-SAN [56], as a successful GNN, processes graph data with selecting, assembling, and normalizing (SAN) operations, detailed as follows.

1) Node Sequence Selection: A fixed-length sequence of W nodes is selected from the N nodes in the network. Nodes are arranged in descending order according to certain important measure. Thus, for different networks, similar important nodes are arranged in similar ranks in the node sequence.

Node sequence selection is the process of sorting and identifying critical nodes. Each node is assigned a score via a labeling procedure, where node centrality measures, such as degree and betweenness are used to describe the node importance. Then, all the nodes are sorted in descending order of the labeling scores; the first W nodes are selected. A receptive fields of size g will be created for each node in the selected sequence. Each receptive field is constructed by *assembling* and *normalizing* as introduced in the following. If $N < W$, all-zero receptive fields are added for padding.

2) Neighborhood Assembly: A set of neighboring nodes is collected for each node in the selected sequence. A breadth-first search is used to collect the neighborhood field, namely, if there are not enough neighboring nodes collected in the current depth, then search in the one-step further neighborhoods, and so on, until at least g neighboring nodes are collected, or no more neighboring node to explore.

3) Normalization: The extracted neighborhood data are ranked to create the normalized receptive fields. The normalization process also imposes an order on the neighboring field for each selected node such that the unordered neighboring field is mapped into an embedding vector space in a linear order. The orders of nodes are determined by a labeling procedure using node centrality measures. In the resultant normalized vector, the root node is assigned as the first element, followed by the second to gth neighboring nodes. This normalization procedure leverages graph labeling on the neighboring nodes of the root node.

To this end, an N-node network is represented by a W-unit receptive field, where each receptive field is a $g \times h$ matrix, with h representing the number of attributes used for the neighboring nodes. Since generally $W \leq N$, $g \ll N$, and $h \ll N$, an $N \times N$ adjacency matrix is mapped to a compressed representation of size Wgh, which will be reshaped and then, passed to a 1D-CNN for further processing in PATCHY-SAN.

TABLE I
COMPARISON OF PCR, PATCHY-SAN, LFR-CNN IN TERMS OF REPRESENTATION, REPRESENTATION SIZE, NUMBER OF LAYERS, AND MAGNITUDE OF NUMBER OF PARAMETERS

	Converted Representation	Size	Feature Maps	Parameters
PCR	Gray-Scale Image	N^2	7(6)	2.4×10^7
PATCHY-SAN	LFR	Wgh	2	5.1×10^5
LFR-CNN	LFR	Wgh	3	6.0×10^6

Since this procedure generates learned feature representations for graph data, it is called an LFR module.

C. LFR-CNN

PCR is straightforward and fast, while PATCHY-SAN extracts topological features first. The input of PCR is the raw adjacency matrix. Since many real-world networks are sparse, which have much fewer edges than the possible maximum number of edges, the input adjacency matrix contains a lot of meaningless information that can be removed or compressed. PATCHY-SAN employs a shallow 1D-CNN structure. Empirically, if properly trained and used, deeper neural networks with more layers and parameters are prone to having better performances than those with fewer layers and parameters, especially for large-scale complex network data.

Table I shows that PCR converts an $N \times N$ adjacency matrix to a gray-scale image without compression, while for PATCHY-SAN an adjacency matrix is compressed to an LFR of size Wgh. The core components of PCR and PATCHY-SAN are a 2D-CNN and a 1D-CNN, respectively. A CNN with 7 feature map (FM) groups (or 6-FM for smaller-sized networks) in PCR requires training a total number of 2.4×10^7 internal parameters, while the 1D-CNN in PATCHY-SAN requires training 5.1×10^5 parameters.

In this article, LFR-CNN is proposed by installing a 2D-CNN (similar to PCR, but with shallower structure) following the LFR module. Compared to PCR and PATCHY-SAN, LFR-CNN has the following advantages.

1) A 2D-CNN can be more powerful than the 1D-CNN in PATCHY-SAN.
2) With LFR, the required number of FMs in 2D-CNN can be significantly reduced, and more importantly, the required number of FMs does not need to change for different network sizes.
3) LFR-CNN requires an intermediate number of training parameters, that is, 6.0×10^6. LFR-CNN achieves a balance in CNN structure and the magnitude of number of parameters between PCR and PATCHY-SAN.

The structures of PCR, PATCHY-SAN, and LFR-CNN are shown in Fig. 2. The input complex network data are adjacency matrix in this article, which could also be a Laplacian matrix or some other representations; the output is the predicted network robustness. The LFR module consists of selection, assembly, and normalization operations. Given the same LFR as the input, a 2D-CNN can capture more feature details than a 1D-CNN, therefore is more suitable to be applied to large-scale complex network data. The proposed LFR-CNN

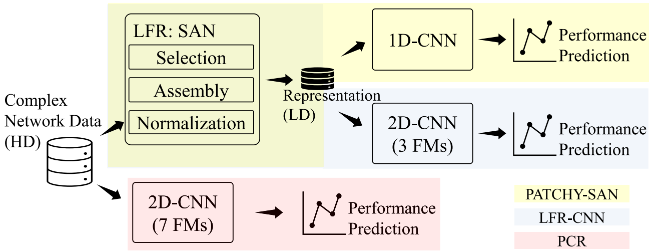


Fig. 2. General framework of PATCHY-SAN, LFR-CNN, and PCR. PATCHY-SAN and LFR-CNN share the common module of LFR performing the SAN tasks, which compresses higher-dimensional (HD) complex network data to be lower-dimensional (LD) representations. LFR-CNN and PCR share a similar 2D-CNN.

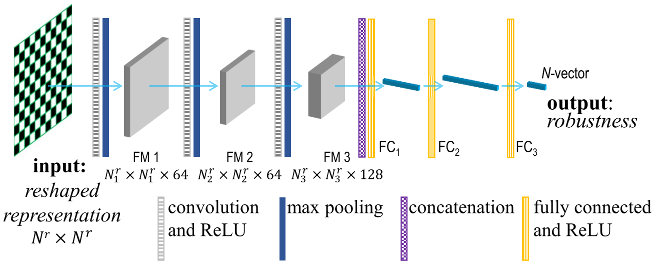


Fig. 3. 2D-CNN structure with 3 FM groups installed in LFR-CNN. $N_i^r = \lceil N^r/2^{(i+1)} \rceil$, for $i = 1, 2, 3$, where $N^r \times N^r$ is the size of the input reshaped representation. The concatenation layer reshapes the matrix to a vector from FM3 to FC$_1$.

TABLE II
PARAMETER SETTING OF THE 3-FM 2D-CNN INSTALLED IN LFR-CNN

Group	Layer	Kernel Size	Stride	Output Channel
Group 1	Conv7-64	7 × 7	1	64
	Max2	2 × 2	2	64
Group 2	Conv5-64	5 × 5	1	64
	Max2	2 × 2	2	64
Group 3	Conv3-128	3 × 3	1	128
	Max2	2 × 2	2	128

naturally combines PATCHY-SAN and PCR by incorporating their advantages.

Similarly to PCR, a VGG-structured [57] CNN is installed in LFR-CNN. For network sizes around $N = 1000$, PCR needs 7 FM groups to perform prediction. When the network size is reduced (e.g., $N = 500$), the number of FMs can be reduced (e.g., 6 FMs). In contrast, since raw graph data are compressed by the LFR module, the CNN in LFR-CNN is not necessary to be adjusted if the network size is not significantly changed. Specifically, as shown in the experimental studies, LFR-CNN is able to process different network sizes $N \in [350, 1300]$ using the same 3-FM CNN.

The detailed structure is illustrated in Fig. 3, and the parameters are summarized in Table II. Each group of FM1–FM3 contains a convolution layer, an ReLU performing the activation function $f(x) = \max(0, x)$ [61], and a max-pooling layer. The output of each hidden layer is summed up, rectified by the ReLU, and then transmitted to the next layer. To that end, max pooling layers will reduce the data dimension as input to the next layer. Then, two fully-connected layers are installed to map feature representations and reshape the regression output. The same loss function as in PCR is employed, as shown in (4). Source codes of this work are available for the public[1].

IV. EXPERIMENTAL STUDIES

A total of nine synthetic network models are simulated, including the Erdös–Rényi (ER) random-graph [62], Barabási–Albert (BA) scale-free [63], [64], generic scale-free (SF) [65], onion-like generic scale-free (OS) [23], Newman–Watts small-world (SW-NW) [66], Watts–Strogatz small-world (SW-WS) [67], q-snapback (QS) [46], random triangle (RT) [68], and random hexagon (RH) [68] networks.

Specifically, a BA network is generated according to the preferential attachment scheme [63], while an SF network is generated according to a set of predefined weights $w_i = (i + \theta)^{-\sigma}$, where $i = 1, 2, \ldots, N$, $\sigma \in [0, 1)$ and $\theta \ll N$. Two nodes i and j are randomly picked and connected with a probability proportional to their weights w_i and w_j, respectively. An OS network is generated based on an SF, with $2N$ rewiring operations toward assortativity maximization. The degree distributions of BA, SF, and OS all follow the power law.

Both SW-NW and SW-WS start from an N-node loop having K (= 2) connected nearest neighbors. The difference is that additional edges are added without removing any existing edges in SW-NW [66]; while rewiring operations are performed in SW-WS [67]. QS consists of a backbone chain and multiple snapback edges [46]. RT and RH consist of RTs and hexagons, respectively, [68]. To exactly control the number of generated edges to be M, uniformly-randomly adding or removing edges can be performed.

For each synthetic model, 1000 instances are randomly generated for training, thus, there are $1000 \times 9 = 9000$ training samples in total. In addition, two different sets of $100 \times 9 = 900$ samples are used for cross-validation and testing, respectively.

The size of each synthetic network instance is randomly determined in three different settings, namely: 1) set $N \in [350, 650]$ (with an average $\bar{N} = 500.5$) for the experiments of predicting connectivity and controllability robustness in Sections IV-A, IV-B, IV-C, IV-D, IV-G, and IV-H; 2) set $N \in [700, 1300]$ (with an average $\bar{N} = 999.8$) for the scalability investigation in Sections IV-B and IV-E; and 3) set $N \in [700, 900]$ (with an average $\bar{N} = 800.0$) for the study of the influence of information loss on the three comparative approaches in Section IV-F.

The average degrees are also assigned randomly. The ranges are set differently for various network models. For SW $\langle k \rangle \in [2.5, 5]$, for RH, $\langle k \rangle \in [2, 4]$, for RT, $\langle k \rangle \in [1.5, 3]$, while for other models, $\langle k \rangle \in [3, 6]$. This setting guarantees the initial connectedness of networks. The overall average degree of the training network is 4.33, while that of the testing network is 4.36, with data obtained by performing posterior statistics.

The proposed LFR-CNN is compared with PATCHY-SAN [56] and PCR [20], [52] in predicting the connectivity and controllability robustness for both synthetic and real-world

[1] https://fylou.github.io/sourcecode.html

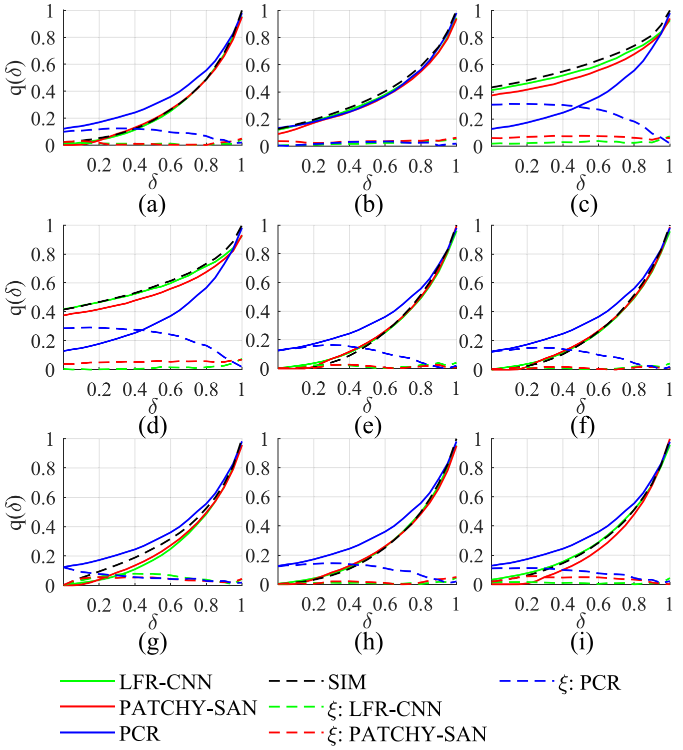


Fig. 4. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for controllability robustness of directed networks ($N \in [350, 650]$) under RA. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

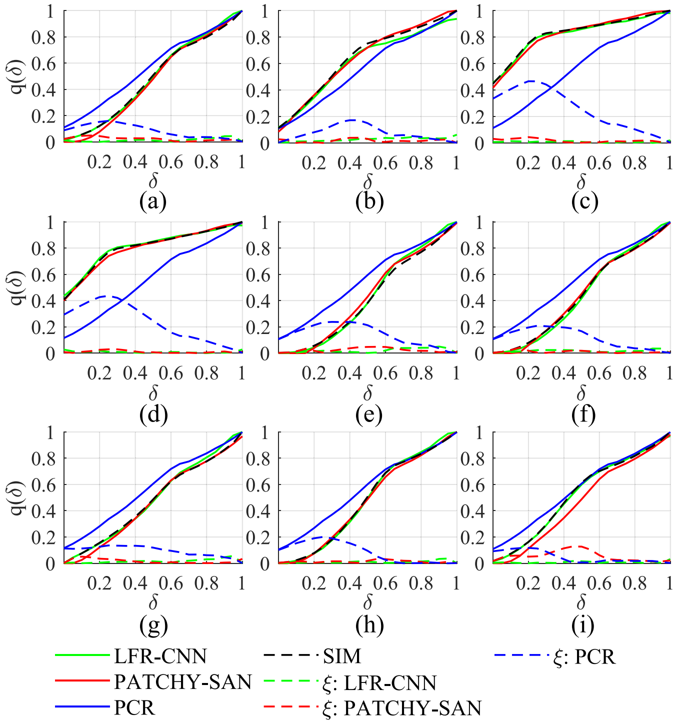


Fig. 5. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for controllability robustness of directed networks ($N \in [350, 650]$) under TB. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

networks under various node-removal attacks, including random attack (RA), targeted betweenness-based (TB) attack, and targeted degree-based (TD) attack. For PCR, a 6-FM CNN is used for $N < 700$ and a 7-FM structure is used for $N \geq 700$. For PATCHY-SAN and LFR-CNN, the structures remain the same for all networks with $N \in [350, 1300]$. For LFR, set the length of the selected node sequence to be $W = 500$ for $N < 700$, and $W = 1000$ for $N \geq 700$; the receptive field size $g = 10$; the number of attributes $h = 2$ (the two default attributes are node degree and clustering coefficient). All experiments are performed on a PC Intel Core i7-8750H CPU @ 2.20GHz, which has memory (RAM) 16 GB with running Windows 10 Home 64-bit Operating System.

A. Predicting Controllability Robustness for Directed Networks

Controllability robustness of directed networks under RA and TB are predicted using LFR-CNN, PCR, and PATCHY-SAN. The simulation results are shown in Figs. 4 and 5. A network controllability curve is denoted by $q(\delta)$, where δ represents the proportion of removed nodes. For each predictor, its predicted controllability curve and prediction error curve (dashed line) are plotted in the same color; "SIM" represents attack simulations. Each curve is averaged from 100 testing samples.

As shown in Figs. 4 and 5, PCR performs badly in prediction. This is due to the following two reasons: 1) both the training and testing data have a wide network size variation with $N \in [350, 650]$ and $\langle k \rangle \in [1.5, 6]$ and 2) there are nine different synthetic network models trained and tested. The fixed input size of PCR requires downsampling or upsampling, which may result in non-negligible information loss. Also, the straightforward processing mechanism makes PCR less capable of dealing with similar network models, for example, BA and SF, SW-NW, and SW-WS. As a result, PCR predicts most controllability curves in a similar pattern for different networks with different sizes and average degrees. In contrast, the LFR module in LFR-CNN and PATCHY-SAN facilitates knowledge extraction. Given $N \in [350, 650]$ and $W = 500$, for the network instances with $N \geq W$, only 500 most important nodes are selected for further processing, while the information loss caused by the unselected nodes is limited. If $N < W$, then, all nodes are selected and their corresponding receptive fields are generated. The LFR module improves the ability of knowledge extraction for different topologies with different sizes. In Figs. 4(c), (d), (i), and 5(i), it is visible that the green curves (LFR-CNN predictions) are closer to the black dotted curves (true simulation results) than the red curves (PATCHY-SAN predictions), meaning that LFR-CNN performs clearly better than PATCHY-SAN in prediction.

Table III summarizes the overall prediction errors of the three predictors in different experiments, with Kruskal–Wallis H-test [69] results. The overall errors of the results in Fig. 4 are shown in Table III(I), which shows that: 1) LFR-CNN performs significantly better than PCR for all networks and 2) LFR-CNN performs significantly better than PATCHY-SAN for ER, SF, OS, and RT, but significantly worse than PATCHY-SAN for SW-WS, QS, and RH. The overall errors of the results in Fig. 5 are shown in Table III(II), which shows that:

TABLE III
COMPARISON OF AVERAGE PREDICTION ERRORS AMONG LFR-CNN, PCR, AND PATCHY-SAN, WHERE $N \in [350, 650]$. THE SIGNS IN PARENTHESES DENOTE THE KRUSKAL–WALLIS H-TEST [69] RESULTS OF LFR-CNN VERSUS PCR AND LFR-CNN VERSUS PATCHY-SAN, RESPECTIVELY. A "+" SIGN DENOTES THAT LFR-CNN SIGNIFICANTLY OUTPERFORMS THE OTHER METHOD BY OBTAINING LOWER ERRORS; A "≈" SIGN DENOTES NO SIGNIFICANT DIFFERENCE BETWEEN TWO METHODS; AND A "−" SIGN DENOTES THAT LFR-CNN PERFORMS SIGNIFICANTLY WORSE THAN THE OTHER METHODS WITH GREATER ERRORS

Average Prediction Error ξ		ER	BA	SF	OS	SW-NW	SW-WS	QS	RH	RT
(I) Controllability Robustness of Directed Networks under RA	LFR-CNN	0.0450 (+,+)	0.0395 (+,≈)	0.0601 (+,+)	0.0567 (+,+)	0.0480 (+,≈)	0.0361 (+,−)	0.0375 (+,−)	0.0440 (+,−)	0.0474 (+,+)
	PCR	0.1280	0.1509	0.2689	0.2541	0.1139	0.1358	0.1301	0.1331	0.1360
	PATCHY-SAN	0.0313	0.0458	0.0732	0.0601	0.0450	0.0253	0.0272	0.0304	0.0541
(II) Controllability Robustness of Directed Networks under TB	LFR-CNN	0.02544 (+,+)	0.05219 (+,−)	0.04376 (+,≈)	0.04650 (+,≈)	0.02355 (+,+)	0.02445 (+,+)	0.02210 (+,≈)	0.02134 (+,+)	0.03641 (+,+)
	PCR	0.1369	0.1625	0.2704	0.2570	0.1374	0.1548	0.1384	0.1302	0.1300
	PATCHY-SAN	0.0354	0.0351	0.0391	0.0388	0.0273	0.0333	0.0238	0.0258	0.0614
(III) Connectivity Robustness of Undirected Networks under RA	LFR-CNN	0.0362 (+,+)	0.0665 (≈,≈)	0.0868 (+,≈)	0.0908 (+,≈)	0.0338 (+,+)	0.0365 (+,+)	0.0350 (+,+)	0.0406 (+,+)	0.0767 (≈,≈)
	PCR	0.0695	0.0767	0.1167	0.1219	0.0663	0.0863	0.0825	0.0728	0.0779
	PATCHY-SAN	0.0639	0.0692	0.0835	0.0803	0.0703	0.0670	0.0663	0.0590	0.0635
(IV) Connectivity Robustness of Undirected Networks under TD	LFR-CNN	0.0302 (+,+)	0.0334 (+,≈)	0.0215 (+,≈)	0.0262 (+,≈)	0.0279 (+,+)	0.0265 (+,+)	0.0254 (+,+)	0.0345 (+,+)	0.0563 (+,≈)
	PCR	0.1423	0.1680	0.2724	0.2792	0.1644	0.1520	0.1402	0.1351	0.1386
	PATCHY-SAN	0.0404	0.0420	0.0230	0.0282	0.0501	0.0446	0.0439	0.0408	0.0460

TABLE IV
BASIC INFORMATION OF REDDIT-MULTI REAL-WORLD NETWORKS. COMPARISON OF AVERAGE PREDICTION ERRORS AMONG LFR-CNN, PCR, AND PATCHY-SAN, FOR CONTROLLABILITY ROBUSTNESS, WHERE $N \in [419, 570]$. NUMBERS IN PARENTHESES DENOTE THE RANKS OF PREDICTORS IN ASCENDING ORDER OF PREDICTION ERRORS

	RW1	RW2	RW3	RW4	RW5	RW6	RW7	RW8	RW9
REDDIT-MULTI [70]	12K-16	12K-40	12K-41	12K-49	12K-81	12K-124	12K-129	5K-1	5K-2
N	499	510	538	551	499	522	570	419	428
$\langle k \rangle$	6.31	8.93	6.84	7.15	4.95	7.56	5.75	47.07	35.01
LFR-CNN	0.1082 (2)	0.0667 (1)	0.1035 (1)	0.1014 (2)	0.0856 (1)	0.1041 (1)	0.0824 (1)	0.1168 (1)	0.0875 (1)
PCR	0.0969 (1)	0.0938 (2)	0.1104 (2)	0.0949 (1)	0.1532 (2)	0.1224 (2)	0.1378 (2)	0.1866 (2)	0.1718 (2)
PATCHY-SAN	0.1503 (3)	0.1211 (3)	0.1497 (3)	0.1531 (3)	0.1733 (3)	0.1563 (3)	0.1679 (3)	0.3611 (3)	0.2636 (3)

1) LFR-CNN performs significantly better than PCR for all networks and 2) LFR-CNN performs significantly better than PATCHY-SAN for ER, SW-NW, SW-WS, RH, and RT, but significantly worse than PATCHY-SAN for BA only. All in all, LFR-CNN outperforms PCR for all networks; LFR-CNN outperforms PATCHY-SAN in nine comparisons, but is worse in four comparisons, while in the other five comparisons, two predictors have no significant differences.

B. *Predicting Robustness for Real-World Networks*

A total of nine real-world network instances are randomly selected from the Reddit multiset data [70]. Three predictors are used to predict the controllability robustness of real-world networks under RA. The basic information of these networks and the prediction errors obtained by the three predictors are summarized in Table IV. Ranks of predictors in ascending order are attached in parentheses following the prediction errors, where the average ranks of LFR-CNN, PCR, and PATCHY-SAN are 1.22, 1.78, and 3, respectively. This suggests that LFR-CNN and PCR have better generalizability than PATCHY-SAN for unknown real-world networks, although the overall prediction errors for all three predictors are relatively greater than that for synthetic networks. The predicted controllability curves are shown in Fig. 6, which demonstrate that LFR-CNN predicts the controllability curves closer to the simulation results than the other two predictors.

To further verify the effectiveness of LFR-CNN, six real-world networks with either significantly greater or smaller sizes are employed for both connectivity and controllability robustness prediction tests. The six networks, namely, DBLP,[2] MovieLens-User,[3] Grid Yeast,[4] C-elegance [71], PolBooks,[5] and Karate,[6] are denoted by RW10–RW15, respectively. Table V shows the basic information and the prediction errors obtained by the three predictors. For all the three predictors, the model trained using $N \in [700, 1300]$ synthetic instances (with CNN input size $W = 1000$) is used to predict the robustness curves for RW10–RW12, while the model trained using $N \in [350, 650]$ instances (with CNN input size $W = 500$) are used to predict the robustness curves for RW13–RW15.

For connectivity robustness, the average prediction ranks of LFR-CNN, PCR, and PATCHY-SAN are 1.33, 2.67, and 2.00, respectively; while for controllability robustness, the ranks are 1.83, 2.17, and 2.00, respectively. Fig. 7 shows the corresponding controllability and connectivity curves.

As can be seen from the comparisons of real-world networks including RW1–RW9 and RW13–RW15, when the network size is similar or less than the CNN input size (here $W = 500$), LFR-CNN and PATCHY-SAN consistently outperform PCR. However, when network size is significantly larger than the

[2] https://networkrepository.com/cit-DBLP.php

[3] https://networkrepository.com/rec-movielens-user-movies-10m.php

[4] https://networkrepository.com/bio-grid-yeast.php

[5] http://www.casos.cs.cmu.edu/computational_tools/datasets/external/polbooks/index11.php

[6] http://konect.cc/networks/ucidata-zachary

TABLE V
BASIC INFORMATION OF SIX REAL-WORLD NETWORKS. COMPARISON OF AVERAGE PREDICTION ERRORS AMONG LFR-CNN, PCR, AND PATCHY-SAN, FOR BOTH CONNECTIVITY AND CONTROLLABILITY ROBUSTNESS. NUMBERS IN PARENTHESES DENOTE THE RANKS OF PREDICTORS IN ASCENDING ORDER OF PREDICTION ERRORS

		RW10	RW11	RW12	RW13	RW14	RW15	Average Rank
	Real-world Networks	DBLP	MovieLens-User	Grid Yeast	C-elegance	Pol Books	Karate	
	N	12591	7602	6008	279	105	34	
	$\langle k \rangle$	3.95	7.30	52.25	7.86	8.40	4.59	
Connectivity Robustness	LFR-CNN	0.1191 (2)	0.0907 (1)	0.1179 (2)	0.0643 (1)	0.1055 (1)	0.0790 (1)	**1.33**
	Patchy-SAN	0.2414 (3)	0.2806 (3)	1.0336 (3)	0.0713 (2)	0.1072 (2)	0.1706 (3)	2.67
	PCR	0.0619 (1)	0.1297 (2)	0.1107 (1)	0.2514 (3)	0.1705 (3)	0.1269 (2)	2.00
Controllability Robustness	LFR-CNN	0.4304 (3)	0.6154 (2)	0.1883 (2)	0.0459 (1)	0.1457 (1)	0.1349 (2)	**1.83**
	Patchy-SAN	0.2001 (2)	0.6745 (3)	0.3746 (3)	0.0675 (2)	0.1892 (2)	0.0996 (1)	2.17
	PCR	0.0923 (1)	0.2665 (1)	0.1116 (1)	0.2653 (3)	0.5360 (3)	0.2342 (3)	2.00

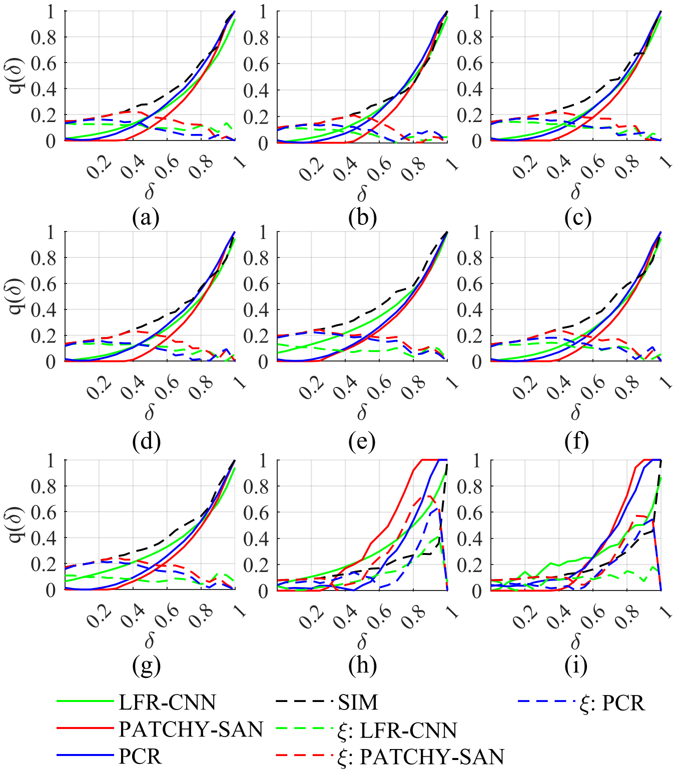


Fig. 6. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for controllability robustness of REDDIT-MULTI [70] real-world networks ($N \in [419, 570]$) under RA. (a) RW1. (b) RW2. (c) RW3. (d) RW4. (e) RW5. (f) RW6. (g) RW7. (h) RW8. (i) RW9.

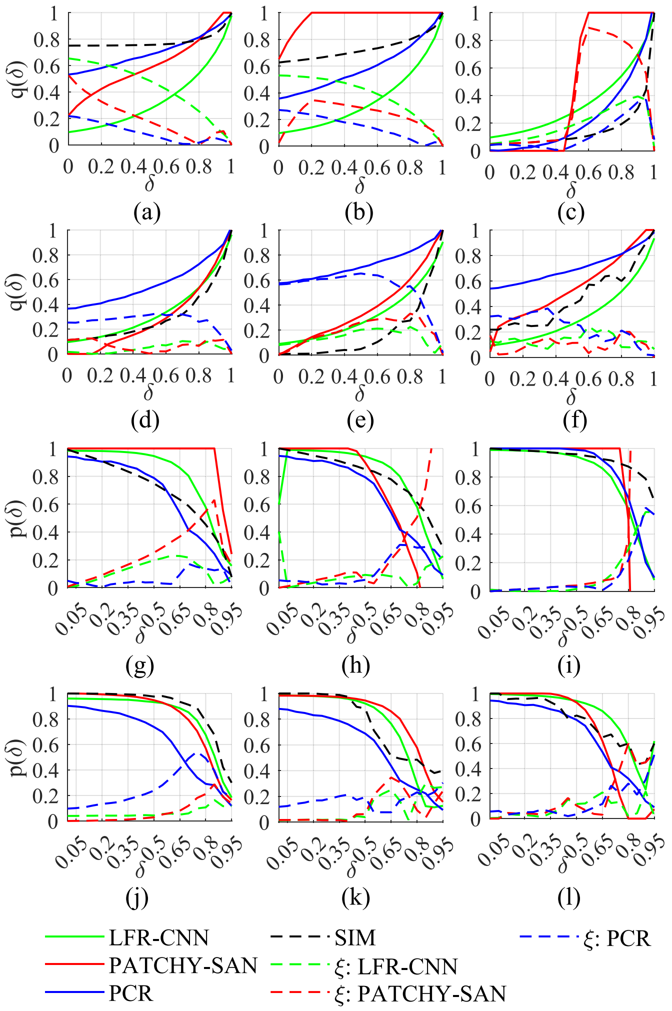


Fig. 7. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN for (a)–(f) controllability robustness, and (g)–(l) connectivity robustness, under RA.

CNN input size $W = 1000$, namely, for RW10–RW12, PCR outperforms LFR-CNN and PATCHY-SAN.

The LFR module deliberately selects W most important nodes from N nodes, while PCR uniformly–randomly picks W out of N nodes. If N is not significantly greater than W, the LFR module will benefit the prediction. However, when $N \gg W$, the selected W most important nodes become a very small and biased portion of all nodes. With the scale-free nature of many real-world networks, only the higher-degree nodes are analyzed in LFR, which distorts the nature of the original network, by giving an illusion that all nodes are of higher degrees in the network. In contrast, as for PCR, the uniformly–random sampling retains the original network feature better. Although subnets of scale-free networks are not scale-free [72], the independent random sampling without replacement process samples more lower-degree nodes but fewer higher-degree nodes. As can be seen from Fig. 7(a), the initial controllability of RW10 is about 0.7, indicating it is a sparse heterogeneous network that requires a high proportion driver nodes. PCR predicts the initial controllability of RW10 by 0.5, which also implies that RW10 is sparse and heterogeneous. However, LFR-CNN and PATCHY-SAN predict the

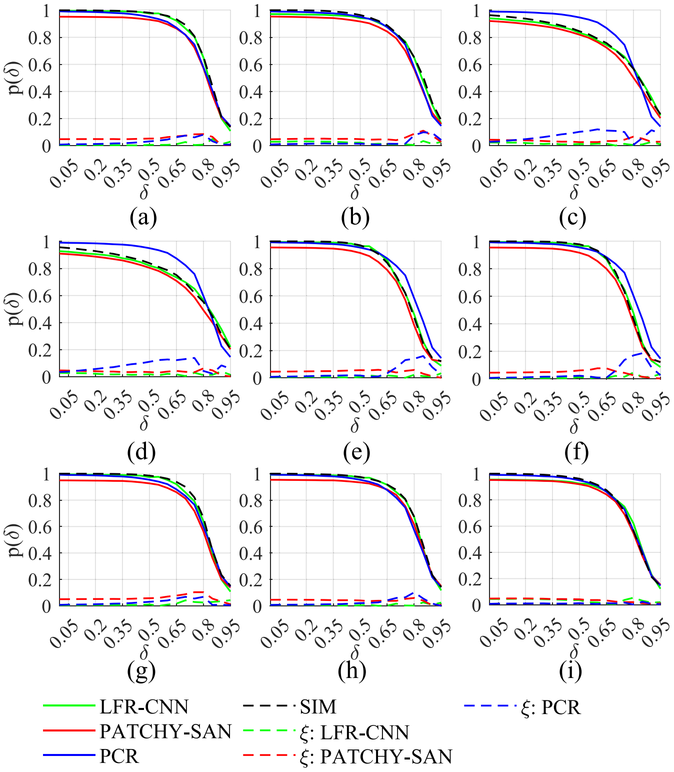


Fig. 8. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for connectivity robustness of undirected networks ($N \in [350, 650]$) under RA. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

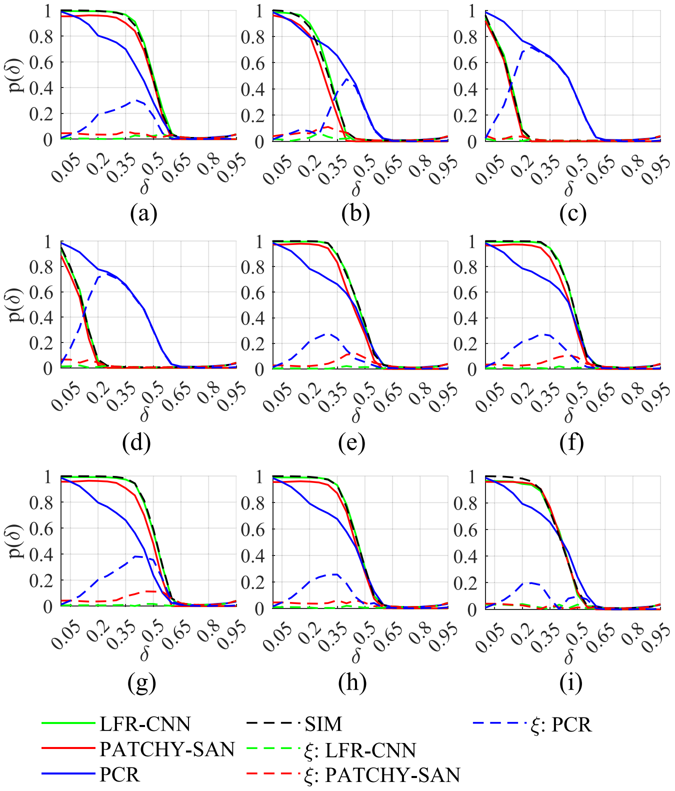


Fig. 9. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for connectivity robustness of undirected networks ($N \in [350, 650]$) under TD. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

initial controllability of RW10 by 0.1 and 0.2, which implies that RW10 is either dense, or homogeneous, or dense and homogeneous. Therefore, the following predictions are also misled. The performance degeneration is caused by the nature of the LFR module.

In general, neither $N \gg W$ nor $N \ll W$ is suggested in applications. Here, the simulation results shown in Table V and Fig. 7 demonstrate the performances in extreme situations.

C. Predicting Connectivity Robustness for Undirected Networks

CNN-based approaches are capable of dealing with *all* types of complex networks, including weighted and unweighted, directed and undirected, real-world, and synthetic networks [53]. Here, for brevity, a comparison of connectivity robustness predictions is performed only on undirected networks. The predicted connectivity curves under RA are shown in Fig. 8, for which the overall prediction errors are summarized in Table III(III). Fig. 8 shows that all the three predictors perform well (or fairly good) on predicting the connectivity curves, which are denoted by $p(\delta)$. Table III(III) shows that the prediction errors are mostly in a magnitude of 10^{-2}. The predicted curves under TD are shown in Fig. 9, for which the overall errors are summarized in Table III(IV). It is clear that PCR performs imprecisely well.

The data summarized in Table III(III) and (IV) suggest that LFR-CNN outperforms PCR and PATCHY-SAN in predicting 16 out of 18 and 10 out of 18 comparisons, respectively, while for the rest networks, LFR-CNN performs statistically equivalently well as PCR and PATCHY-SAN.

In a nutshell, LFR-CNN outperforms PCR in 34/36 cases, and outperforms PACTHY-SAN in 19/36 cases; PACTHY-SAN outperforms LFR-CNN in 4/36 cases, while PCR does not outperform LFR-CNN in any case; for the rest cases, no significant differences are detected. Specifically, as shown in all four sections in Table III, LFR-CNN generally outperforms both PCR and PATCHY-SAN for homogeneous networks (ER, SW-NW, SW-WS, RH, and RT), but outperforms PCR only for heterogeneous networks (BA, SF, and OS). The LFR module works consistently well for homogeneous networks. As for heterogeneous networks, as explored in Section IV-B, if there is a large difference between the network size and the CNN input size, the extracted lower-dimensional features may not be always effective.

D. Node Attributes as Receptive Fields

In the normalization step of LFR, the attributes of the selected neighborhood nodes are embedded in a receptive field. Here, different combinations of node attributes, including degree, clustering coefficient, and betweenness are compared. Table VI shows the prediction errors for: 1) controllability robustness and 2) connectivity robustness, among the three combinations. It is clear that the default setting using degree and clustering coefficient (*deg* & *cc*) outperforms the other two combinations.

TABLE VI
COMPARISON OF AVERAGE PREDICTION ERRORS OBTAINED USING DIFFERENT ATTRIBUTE COMBINATIONS IN LFR-CNN. THREE NODE ATTRIBUTES, INCLUDING DEGREE (*deg*), CLUSTERING COEFFICIENT (*cc*), AND BETWEENNESS (*bet*), COMPOSE THREE PAIRWISE COMBINATIONS

		ER	BA	SF	OS	SW-NW	SW-WS	QS	RH	RT
(I) Controllability Robustness of Directed Networks under RA	*deg* & *cc*	0.0432 (≈,+)	0.0357 (+,+)	0.0436 (≈,+)	0.0372 (+,+)	0.0581 (+,+)	0.0322 (≈,+)	0.0351 (≈,+)	0.0399 (≈,+)	0.0421 (+,+)
	deg & *bet*	0.0384	0.0562	0.0472	0.0556	0.0439	0.0321	0.0337	0.0394	0.0515
	bet & *cc*	0.0589	0.0865	0.1203	0.1179	0.0640	0.0543	0.0566	0.0571	0.0681
(II) Connectivity Robustness of Undirected Networks under RA	*deg* & *cc*	0.0293 (+,+)	0.0490 (≈,+)	0.0791 (≈,+)	0.0769 (+,+)	0.0287 (+,+)	0.0288 (+,+)	0.0287 (+,+)	0.0340 (+,+)	0.0461 (≈,+)
	deg & *bet*	0.0503	0.0494	0.0921	0.0937	0.0635	0.0562	0.0527	0.0508	0.0568
	bet & *cc*	0.1291	0.1434	0.1628	0.1632	0.1331	0.1339	0.1298	0.1325	0.1454

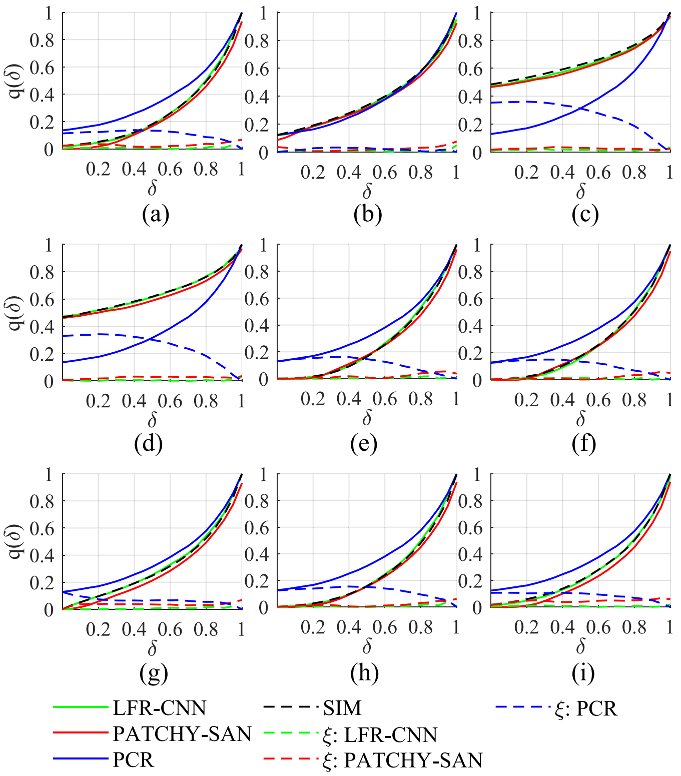


Fig. 10. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for controllability robustness of directed networks ($N \in [700, 1300]$) under RA. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

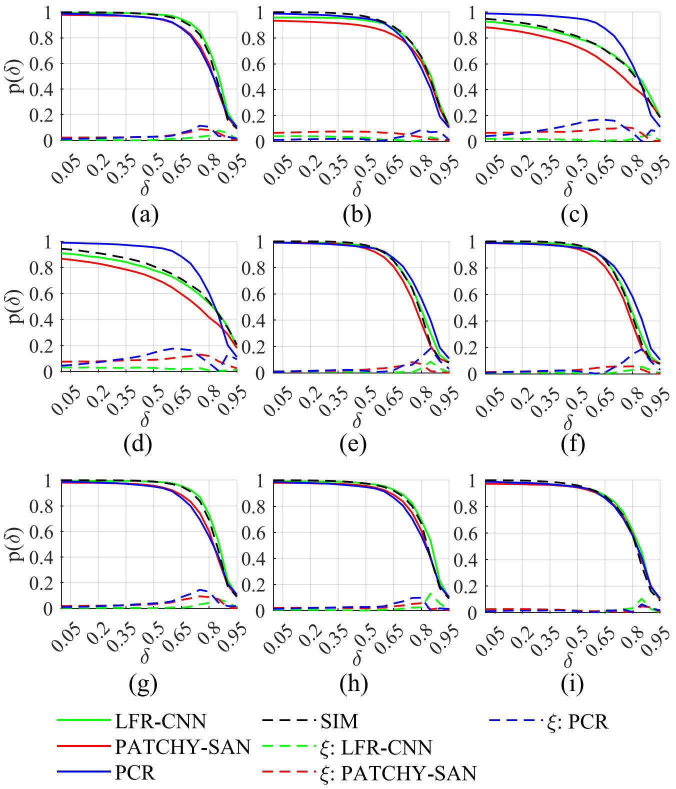


Fig. 11. [Color online] Comparison of prediction results using LFR-CNN, PCR, and PATCHY-SAN, for connectivity robustness of undirected networks ($N \in [700, 1300]$) under RA. (a) ER. (b) BA. (c) SF. (d) OS. (e) SW-NW. (f) SW-WS. (g) QS. (h) RH. (i) RT.

Note that features are embedded in local receptive fields, thus, local features *deg* and *cc* are more suitable, while global feature *bet* is not. However, this result does not diminish the importance of *bet* regarding both connectivity and controllability robustness, but suggests a global methodology to embed it.

E. Scalability of Network Size Variation

To further verify the scalability, the proposed LFR-CNN is compared with PCR and PATCHY-SAN on predicting a set of networks of sizes $N \in [700, 1300]$. Here, a 7-FM PCR is employed and $W = 1000$ is set for LFR-CNN and PATCHY-SAN. The predicted controllability and connectivity curves under RA are shown in Figs. 10 and 11, respectively. It is visible that LFR-CNN and PATCHY-SAN perform better than PCR in controllability robustness prediction. As for connectivity robustness, LFR-CNN performs visibly better than PATCHY-SAN and PCR in Fig. 11 (c) and (d).

The overall prediction errors are shown in Table VII. LFR-CNN outperforms PCR for 17 out of 18 cases, and outperforms PATCHY-SAN for 13 out of 18 cases; while for the rest comparisons, LFR-CNN performs statistically equivalently to PCR or PATCHY-SAN in prediction.

F. For Identical Network Size

The core prediction components in LFR-CNN, PCR, and PATCHY-SAN are 3-FM 2D-CNN, 7-FM 2D-CNN, and 1D-CNN, respectively. These CNN-based core components perform the regression task and predict the robustness performance for an input network. In PCR, the input data

TABLE VII
COMPARISON OF AVERAGE PREDICTION ERRORS AMONG LFR-CNN, PCR, AND PATCHY-SAN, WHERE $N \in [700, 1300]$. THE SIGNS IN PARENTHESES DENOTE THE KRUSKAL–WALLIS H-TEST [69] RESULTS OF LFR-CNN VERSUS PCR AND LFR-CNN VERSUS PATCHY-SAN, RESPECTIVELY. A "+" SIGN DENOTES THAT LFR-CNN SIGNIFICANTLY OUTPERFORMS THE OTHER METHOD BY OBTAINING LOWER ERRORS; A "≈" SIGN DENOTES NO SIGNIFICANT DIFFERENCE BETWEEN TWO METHODS; AND A "−" SIGN DENOTES THAT LFR-CNN PERFORMS SIGNIFICANTLY WORSE THAN THE OTHER METHODS WITH GREATER ERRORS

Average Prediction Error $\bar{\xi}$		ER	BA	SF	OS	SW-NW	SW-WS	QS	RH	RT
(I) Controllability Robustness of Directed Networks under RA	LFR-CNN	0.0191 (+,+)	0.0406 (+,≈)	0.0356 (+,≈)	0.0341 (+,≈)	0.0151 (+,+)	0.0171 (+,+)	0.0162 (+,+)	0.0177 (+,+)	0.0316 (+,+)
	PCR	0.1433	0.1408	0.2820	0.2706	0.1349	0.1282	0.1242	0.1395	0.1284
	PATCHY-SAN	0.0374	0.0387	0.0420	0.0448	0.0259	0.0240	0.0375	0.0268	0.0499
(II) Connectivity Robustness of Undirected Networks under RA	LFR-CNN	0.0266 (+,+)	0.0594 (≈,+)	0.0705 (+,+)	0.0790 (+,+)	0.0239 (+,+)	0.0297 (+,≈)	0.0271 (+,+)	0.0293 (+,+)	0.0424 (+,≈)
	PCR	0.0654	0.0744	0.1321	0.1348	0.0784	0.0861	0.0833	0.0741	0.0809
	PATCHY-SAN	0.0440	0.0757	0.0971	0.1070	0.0479	0.0444	0.0427	0.0357	0.0398

TABLE VIII
COMPARISON OF AVERAGE PREDICTION ERRORS AMONG LFR-CNN, PCR, AND PATCHY-SAN, WHERE $N = 800$

Average Prediction Error $\bar{\xi}$	ER	SF	QS	SW-NW
LFR-CNN	0.0189 (≈,+)	0.0750 (−,+)	0.0162 (≈,+)	0.0157 (≈,+)
PCR	0.0166	0.0194	0.0145	0.0141
PATCHY-SAN	0.0253	0.1074	0.0208	0.0263

TABLE IX
RUN TIME COMPARISON OF PCR, PATCHY-SAN, LFR-CNN, AND ATTACK SIMULATION (SIM)

Unit: Second	Controllability Robustness		Connectivity Robustness	
SIM	4.7902		1.3704	
PCR	0.0463		0.0477	
PATCHY-SAN	LFR 1.1312	1D-CNN 0.0034	LFR 1.1302	1D-CNN 0.0035
	1.1346		1.1337	
LFR-CNN	LFR 1.1320	CNN 0.0051	LFR 1.1300	CNN 0.0049
	1.1371		1.1349	

to CNN are adjacency matrices, while for LFR-CNN and PATCHY-SAN, the LFR module will convert the raw adjacency matrices to lower-dimensional representations before inputting them to the respective CNNs. Specifically, suppose that H is the input size of the prediction component of LFR-CNN, PCR, or PATCHY-SAN, and given an input adjacency matrix of size $N \times N$ ($N \neq H$). Upsampling or downsampling is necessary to resize the input for PCR, where the original adjacency information may be significantly modified. In contrast, for LFR-CNN and PATCHY-SAN, the $N \times N$ matrix is represented by a sequence of W receptive fields, namely, the information of W most important nodes is input, while if $N > W$, some less important information will be discarded. Therefore, if a network size disagrees with the input size of a predictor, information loss is generally more severe in PCR than in LFR-CNN and PATCHY-SAN.

Table VIII shows the prediction errors when all the network sizes are equal to the input size of CNNs, for both training and testing data, namely, $H = N = W = 800$, with $\langle k \rangle \in [1.5, 6]$. In this case, neither upsampling nor downsampling is required for PCR. All three predictors perform quite well, with very low prediction errors. LFR-CNN outperforms PATCHY-SAN for all four networks, and PCR outperforms LFR-CNN for SF network. This suggests that PCR is fragile to the variation of network size. This verifies that LFR makes the prediction performance more robust against network size variation.

G. Run Time Comparison

Table IX shows the run time comparison of PCR, PATCHY-SAN, LFR-CNN, and attack simulation, for both controllability and connectivity robustness predictions. The network size is $N \in [350, 650]$; the data are averaged from 100 independent runs. As shown in Table IX, the simulation time for controllability robustness is longer than that for connectivity robustness, while for the three predictors, there is no significant difference between the prediction time of controllability and connectivity robustness. It is also notable that PCR is significantly faster than attack simulation, PATCHY-SAN, and LFR-CNN. Running the LFR module is time-consuming, while running the CNN in either PATCHY-SAN or LFR-CNN is faster than PCR due to a simpler structure used.

Overall, compared to attack simulation, LFR-CNN is able to predict relatively precise controllability and connectivity curves, by saving about 76% and 17% computational time, respectively. In addition, run time for attack simulation increases faster than CNN-based approaches, for example, with $N \in [700, 1300]$, the run time for controllability robustness attack simulation is 41.62 s, while it is only 3.67 s for LFR-CNN.

H. Compared to Spectral Measures

Spectral measures are widely used for estimating network connectivity robustness of undirected networks [36]. Table X shows the estimated connectivity robustness ranks of different networks, using three CNN-based predictors and six spectral measures, including algebraic connectivity (AC), effective resistance (EF), natural connectivity (NC), spectral gap (SG), spectral radius (SR), and spanning tree count (ST). Undirected networks with $N \in [350, 650]$ and $\langle k \rangle \in [1.5, 6]$ are used for comparison. Prediction results are unified by the predicted rank errors of network robustness, calculated by $\xi_r = |\hat{rl} - rl|$, where $\hat{rl}$ represents a predicted rank-list and rl is the true rank-list by simulation. For example, given $\hat{rl} = [5, 3, 1, 4, 2]$ and $rl = [2, 3, 1, 5, 4]$, the rank error is $\xi_r = |\hat{rl} - rl| =$

TABLE X
PREDICTION RANK ERRORS OF THE SIX SPECTRAL MEASURES, PCR, PATCHY-SAN, AND LFR-CNN. BOLD NUMBERS INDICATE THE BEST PERFORMING PREDICTION MEASURES

Average Rank Error	ER	BA	SF	OS	QS	SW-NW	SW-WS	RH	RT	Overall	Rank
AC	34.7	35.2	35.4	39.7	34.2	30.4	31.5	35.6	32.8	34.4	8
EF	**30.6**	37.3	35.5	39.7	32.8	33.6	34.4	32.2	36.0	34.7	9
NC	31.8	33.6	34.2	33.7	30.9	27.9	34.0	32.1	32.7	32.3	4
SG	31.3	33.2	31.0	34.2	34.0	29.1	32.6	33.0	35.8	32.7	6
SR	33.5	30.9	**29.9**	33.3	33.8	31.7	30.3	34.6	33.2	32.4	5
ST	37.6	32.1	32.4	**30.7**	34.0	**27.2**	33.0	32.0	**29.0**	32.0	3
PCR	33.4	35.1	35.5	33.5	37.9	31.0	34.3	32.8	33.2	34.1	7
PATCHY-SAN	35.4	**28.7**	30.6	31.1	32.5	28.4	30.1	**30.3**	29.6	**30.7**	**1**
LFR-CNN	33.5	36.7	31.7	31.6	**30.3**	28.3	**29.2**	30.6	29.4	31.3	2

[3, 0, 0, 1, 2] and the average rank error is $\bar{\xi}_r = 1.2$. As shown in Table X, PATCHY-SAN and LFR-CNN obtain the best two average rank errors, while PCR does not perform well due to a larger variation of network size and average degree.

V. CONCLUSION

In this article, an LFR-CNN is proposed for network robustness performance prediction, including both connectivity robustness and controllability robustness. Conventionally, network robustness is evaluated by time-consuming attack simulations, from which a sequence of network connectivity or controllability values are collected and used to measure the remaining network after a sequence of destructive attacks (here, node-removal attacks). LFR-CNN is designed to gain a balance between PCR and PATCHY-SAN, in terms of both input size and internal parameters. The LFR module not only compresses the raw higher-dimensional adjacency matrix to a lower-dimensional representation but also extends the capability of LFR-CNN to process complex network data with a wide-ranged variation of network size and average degree parameters.

Extensive numerical experiments are performed using both synthetic and real-world networks, including directed and undirected networks, and then analyzed and compared, which reveal clearly the pros and cons of several typical and comparable schemes and measures. Specifically, the good performance of LFR-CNN in predicting both connectivity robustness and controllability robustness is verified by comparing with other two state-of-the-art network robustness predictors, PCR and PATCHY-SAN. It is found that LFR-CNN is much less sensitive than PCR to the network size variation. Although LFR-CNN requires a relatively long run time for feature learning, it can still achieve accurate prediction faster than the conventional attack simulations. Meanwhile, LFR-CNN not only accurately predicts the connectivity and controllability robustness curves of various complex networks under different types of attacks, but also serves as an excellent indicator for the connectivity robustness, better than spectral measures.

The present study, after all, makes the current investigation of network connectivity and controllability robustness more subtle and complete. Yet, it should be noted that the correlation between controllability robustness and spectral measures has not been investigated, leaving a good but challenging topic for future research pursuits.

REFERENCES

[1] A.-L. Barabási, *Network Science*. Cambridge, U.K.: Cambridge Univ. Press, 2016.

[2] M. E. Newman, *Networks: An Introduction*. Oxford, U.K.: Oxford Univ. Press, 2010.

[3] G. Chen, X. Wang, and X. Li, *Fundamentals of Complex Networks: Models, Structures and Dynamics*, 2nd ed. Hoboken, NJ, USA: Wiley, 2014.

[4] G. Chen and Y. Lou, *Naming Game: Models, Simulations and Analysis*. Cham, Switzerland: Springer, 2019.

[5] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Controllability of complex networks,” *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[6] Z. Z. Yuan, C. Zhao, Z. R. Di, W.-X. Wang, and Y.-C. Lai, “Exact controllability of complex networks,” *Nat. Commun.*, vol. 4, p. 2447, Sep. 2013.

[7] M. Pósfai, Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, “Effect of correlations on network controllability,” *Sci. Rep.*, vol. 3, p. 1067, Jan. 2013.

[8] G. Menichetti, L. Dall’Asta, and G. Bianconi, “Network controllability is determined by the density of low in-degree and out-degree nodes,” *Phys. Rev. Lett.*, vol. 113, no. 7, 2014, Art. no. 78701.

[9] Y. Pan and X. Li, “Structural controllability and controlling centrality of temporal networks,” *PLoS One*, vol. 9, no. 4, 2014, Art. no. e94998.

[10] A. E. Motter, “Networkcontrology,” *Chaos Interdiscipl. J. Nonlinear Sci.*, vol. 25, no. 9, 2015, Art. no. 97621.

[11] L. Wang, X. Wang, G. Chen, and W. K. S. Tang, “Controllability of networked MIMO systems,” *Automatica*, vol. 69, pp. 405–409, Jul. 2016.

[12] Y.-Y. Liu and A.-L. Barabási, “Control principles of complex systems,” *Rev. Mod. Phys.*, vol. 88, no. 3, 2016, Art. no. 35006.

[13] L. Wang, X. Wang, and G. Chen, “Controllability of networked higher-dimensional systems with one-dimensional communication,” *Roy. Soc. Philos. Trans. A*, vol. 375, no. 2088, 2017, Art. no. 20160215.

[14] L.-Z. Wang, Y.-Z. Chen, W.-X. Wang, and Y.-C. Lai, “Physical controllability of complex networks,” *Sci. Rep.*, vol. 7, Jan. 2017, Art. no. 40198.

[15] B. Hou, X. Li, and G. Chen, “The roles of input matrix and nodal dynamics in network controllability,” *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 4, pp. 1764–1774, Dec. 2018.

[16] Y. Zhang and T. Zhou, “Controllability analysis for a networked dynamic system with autonomous subsystems,” *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3408–3415, Jul. 2017.

[17] L. Xiang, F. Chen, W. Ren, and G. Chen, “Advances in network controllability,” *IEEE Circuits Syst. Mag.*, vol. 19, no. 2, pp. 8–32, 2nd Quart., 2019.

[18] J.-N. Wu, X. Li, and G. Chen, “Controllability of deep-coupling dynamical networks,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 5211–5222, Dec. 2020.

[19] G. Chen, “Searching for best network topologies with optimal synchronizability: A brief review,” *IEEE/CAA J. Autom. Sinica*, vol. 9, no. 4, pp. 573–577, Apr. 2022.

[20] Y. Lou, R. Wu, J. Li, L. Wang, and G. Chen, “A convolutional neural network approach to predicting network connectedness robustness,” *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 4, pp. 3209–3219, Oct.–Dec. 2021.

[21] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack vulnerability of complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 65, no. 5, 2002, Art. no. 56109.

[22] B. Shargel, H. Sayama, I. R. Epstein, and Y. Bar-Yam, "Optimization of robustness and connectivity in complex networks," *Phys. Rev. Lett.*, vol. 90, no. 6, 2003, Art. no. 68701.

[23] C. M. Schneider, A. A. Moreira, J. S. Andrade, S. Havlin, and H. J. Herrmann, "Mitigation of malicious attacks on networks," *Proc. Nat. Acad. Sci.*, vol. 108, no. 10, pp. 3838–3841, 2011.

[24] A. Bashan, Y. Berezin, S. V. Buldyrev, and S. Havlin, "The extreme vulnerability of interdependent spatially embedded networks," *Nat. Phys.*, vol. 9, pp. 667–672, Aug. 2013.

[25] C. Fan, L. Zeng, Y. Sun, and Y.-Y. Liu, "Finding key players in complex networks through deep reinforcement learning," *Nat. Mach. Intell.*, vol. 2, pp. 317–324, Jun. 2020.

[26] M. Grassia, M. De Domenico, and G. Mangioni, "Machine learning dismantling and early-warning signals of disintegration in complex systems," *Nat. Commun.*, vol. 12, p. 5190, Aug. 2021.

[27] G. Yan *et al.*, "Network control principles predict neuron function in the *CaenorhabditisElegans* Connectome," *Nature*, vol. 550, no. 7677, pp. 519–523, 2017.

[28] T. Qiu, J. Liu, W. Si, and D. O. Wu, "Robustness optimization scheme with multi-population co-evolution for scale-free wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1028–1042, Jun. 2019.

[29] Z. Chen, J. Wu, Y. Xia, and X. Zhang, "Robustness of interdependent power grids and communication networks: A complex network perspective," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 1, pp. 115–119, Jan. 2018.

[30] C. Yang, J. Mao, X. Qian, and P. Wei, "Designing robust air transportation networks via minimizing total effective resistance," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2353–2366, Jun. 2019.

[31] Q. Cai, S. Alam, H. Ang, and V. Duong, "A Braess's paradox inspired method for enhancing the robustness of air traffic networks," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, 2020, pp. 798–805.

[32] A. Zeng and W. Liu, "Enhancing network robustness against malicious attacks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 85, no. 6, 2012, Art. no. 66130.

[33] V. H. Louzada, F. Daolio, H. J. Herrmann, and M. Tomassini, "Smart rewiring for network robustness," *J. Complex Netw.*, vol. 1, no. 2, pp. 150–159, 2013.

[34] C. M. Schneider, N. Yazdani, N. A. Araújo, S. Havlin, and H. J. Herrmann, "Towards designing robust coupled networks," *Sci. Rep.*, vol. 3, no. 1, pp. 1–7, 2013.

[35] L. Bai, Y.-D. Xiao, L.-L. Hou, and S.-Y. Lao, "Smart rewiring: Improving network robustness faster," *Chin. Phys. Lett.*, vol. 32, no. 7, 2015, Art. no. 78901.

[36] H. Chan and L. Akoglu, "Optimizing network robustness by edge rewiring: A general framework," *Data Min. Knowl. Discov.*, vol. 30, no. 5, pp. 1395–1425, 2016.

[37] Y. Lou, S. Xie, and G. Chen, "Searching better rewiring strategies and objective functions for stronger controllability robustness," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 6, pp. 2112–2116, Jun. 2021.

[38] S. Wang, J. Liu, and Y. Jin, "A computationally efficient evolutionary algorithm for multiobjective network robustness optimization," *IEEE Trans. Evol. Comput.*, vol. 25, no. 3, pp. 419–432, Jun. 2021.

[39] L. Ma *et al.*, "Enhancing robustness and resilience of multiplex networks against node-community cascading failures," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 52, no. 6, pp. 3808–3821, Jun. 2022.

[40] M. E. Newman, "Mixing patterns in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 67, no. 2, 2003, Art. no. 26126.

[41] N. Perra and S. Fortunato, "Spectral centrality measures in complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 78, no. 3, 2008, Art. no. 36107.

[42] Z.-X. Wu and P. Holme, "Onion structure and network robustness," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, no. 2, 2011, Art. no. 26106.

[43] T. Tanizawa, S. Havlin, and H. E. Stanley, "Robustness of onion-like correlated networks against targeted attacks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 85, no. 4, 2012, Art. no. 46109.

[44] Y. Hayashi and N. Uchiyama, "Onion-like networks are both robust and resilient," *Sci. Rep.*, vol. 8, Jul. 2018, Art. no. 11241.

[45] X.-Y. Yan, W.-X. Wang, G.-R. Chen, and D.-H. Shi, "Multiplex congruence network of natural numbers," *Sci. Rep.*, vol. 6, Mar. 2016, Art. no. 23714.

[46] Y. Lou, L. Wang, and G. Chen, "Toward stronger robustness of network controllability: A snapback network model," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 9, pp. 2983–2991, Sep. 2018.

[47] Y. Lou, L. Wang, K.-F. Tsang, and G. Chen, "Towards optimal robustness of network controllability: An empirical necessary condition," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 9, pp. 3163–3174, Sep. 2020.

[48] H. Iiduka, "Appropriate learning rates of adaptive learning rate optimization algorithms for training deep neural networks," *IEEE Trans. Cybern.*, early access, Sep. 8, 2021, doi: 10.1109/TCYB.2021.3107415.

[49] B. Xiao *et al.*, "PAM-DenseNet: A deep convolutional neural network for computer-aided COVID-19 diagnosis," *IEEE Trans. Cybern.*, early access, Aug. 24, 2021, doi: 10.1109/TCYB.2020.3042837.

[50] J. Sun, W. Zheng, Q. Zhang, and Z. Xu, "Graph neural network encoding for community detection in attribute networks," *IEEE Trans. Cybern.*, vol. 52, no. 8, pp. 7791–7804, Aug. 2022.

[51] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.

[52] Y. Lou, Y. He, L. Wang, and G. Chen, "Predicting network controllability robustness: A convolutional neural network approach," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 4052–4063, May 2022.

[53] Y. Lou, Y. He, L. Wang, K. F. Tsang, and G. Chen, "Knowledge-based prediction of network controllability robustness," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 16, 2021, doi: 10.1109/TNNLS.2021.3071367.

[54] R. Zhang, "Making convolutional networks shift-invariant again," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7324–7334.

[55] C. Wu, Y. Lou, R. Wu, W. Liu, and J. Li, "CNN-based prediction of network robustness with missing edges," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2022, pp. 1–8.

[56] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2014–2023.

[57] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[58] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[59] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 1025–1035.

[60] W. L. Hamilton, *Graph Representation Learning* (Synthesis Lectures on Artificial Intelligence and Machine Learning), vol. 14. Cham, Switzerland: Springer, 2020.

[61] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2011, pp. 315–323.

[62] P. Erdös and A. Rényi, "On the strength of connectedness of a random graph," *Acta Mathematica Hungarica*, vol. 12, nos. 1–2, pp. 261–267, 1964.

[63] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[64] A.-L. Barabási, "Scale-free networks: A decade and beyond," *Science*, vol. 325, no. 5939, pp. 412–413, 2009.

[65] K.-I. Goh, B. Kahng, and D. Kim, "Universal behavior of load distribution in scale-free networks," *Phys. Rev. Lett.*, vol. 87, no. 27, 2001, Art. no. 278701.

[66] M. E. Newman and D. J. Watts, "Renormalization group analysis of the small-world network model," *Phys. Lett. A*, vol. 263, nos. 4–6, pp. 341–346, 1999.

[67] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

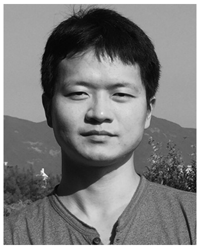
[68] G. Chen, Y. Lou, and L. Wang, "A comparative study on controllability robustness of complex networks," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 66, no. 5, pp. 828–832, May 2019.

[69] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *J. Amer. Stat. Assoc.*, vol. 47, no. 260, pp. 583–621, 1952.

[70] P. Yanardag and S. Vishwanathan, "Deep graph kernels," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (KDD)*, 2015, pp. 1365–1374.

[71] R. Rossi and N. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 4292–4293.

[72] M. P. Stumpf, C. Wiuf, and R. M. May, "Subnets of scale-free networks are not scale-free: Sampling properties of networks," *Proc. Nat. Acad. Sci.*, vol. 102, no. 12, pp. 4221–4224, 2005.



Yang Lou (Member, IEEE) received the Ph.D. degree from the Department of Electrical Engineering, City University of Hong Kong, Hong Kong, China, in 2017.

He was a Research Assistant Professor with the Department of Computing and Decision Sciences, Lingnan University, Hong Kong, China, from 2021 to 2022, and a Postdoctoral Research Fellow with the Centre for Chaos and Complex Networks and the Department of Electrical Engineering, City University of Hong Kong, from 2017 to 2021. He is an Associate Professor with the College of Computer Science, Sichuan Normal University, Chengdu, China, and has been appointed as a Specially Appointed Researcher with the Graduate School of Information Science and Technology, Osaka University, Suita, Japan. His current research interests include machine learning, network science, and optimization.

Ruizi Wu received the B.E. degree from Southwest Jiaotong University, Chengdu, China, in 2020. He is currently pursuing the master’s degree with the School of Computer Science, Sichuan Normal University, Chengdu.

His research interests include complex networks, evolutionary computation, and machine learning.

Junli Li received the Ph.D. degree from the Department of Mathematics, Zhejiang University, Hangzhou, Zhejiang, China, in 2002.

He is currently a Professor with the College of Computer Science, Sichuan Normal University, Chengdu, Sichuan, China. His research interests include complex networks, evolutionary computation, and machine learning.

Lin Wang (Senior Member, IEEE) received the B.S. and M.S. degrees from the School of Mathematical Sciences, Shandong Normal University, Jinan, China, in 2003 and 2006, respectively, and the Ph.D. degree in operations research and control theory from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2009.

She is currently a Professor with the Department of Automation, Shanghai Jiao Tong University, Shanghai, China. Her current research interests include multiagent systems, adaptive complex networks, and coordination of multiple manipulators.

Xiang Li (Senior Member, IEEE) received the B.S. and Ph.D. degrees in control theory and control engineering from Nankai University, Tianjin, China, in 1997 and 2002, respectively.

From 2002 to 2004, he was a Postdoctoral Research Fellow with the City University of Hong Kong, Hong Kong. From 2005 to 2006, he was a Humboldt Research Fellow with Int. University Bremen, Bremen, Germany. From 2004 to 2007, he was an Associate Professor with Shanghai Jiao Tong University, Shanghai, China. From 2008 to 2021, he was a Full Professor/Distinguished Professor with Fudan University, Shanghai. He is currently a Distinguished Professor with Tongji University, Shanghai, and the Founding Director of the Institute of Complex Networks and Intelligent Systems, Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University. His main research interests cover network science and intelligent systems with control theory and applications.

Dr. Li received the IEEE Guillemin-Cauer Best Paper Award from the IEEE Circuits and Systems Society in 2005, the Shanghai Natural Science Award (First Class) in 2008, the National Science Foundation for Distinguished Young Scholar of China in 2014, the National Natural Science Award of China (Second Class) in 2015, the Ten Thousand Talent Program of China in 2017, the TCCT CHEN Han-Fu Award of Chinese Automation Association in 2019, and the Excellent Editor Award of the IEEE TRANSACTIONS NETWORK SCIENCE AND ENGINEERING in 2021, among other awards and honors. He serves as an Associate Editor for *Journal of Complex Networks*, *Research*, and the IEEE Circuits and Systems Society Newsletter, and an Associate Editor from 2018 to 2021 and an Area Editor since 2022 for the IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING.

Guanrong Chen (Life Fellow, IEEE) received the M.Sc. degree in computer science from Sun Yat-sen University, Guangzhou, China, in 1981, and the Ph.D. degree in applied mathematics from Texas A&M University at College Station, College Station, TX, USA, in 1987.

Since 2000, he has been a Chair Professor and the Founding Director of the Centre for Complexity and Complex Networks, City University of Hong Kong, Hong Kong, and is currently the Hong Kong Shun Hing Education and Charity Fund Chair Professor in Engineering. His research interests are in the fields of complex networks, nonlinear dynamics, and control systems.

Prof. Chen was awarded the 2011 Euler Gold Medal from Russia, and conferred Honorary Doctor Degrees by the Saint Petersburg State University, Russia, in 2011 and by the University of Normandy, France, in 2014. He has been a member of the Academy of Europe since 2014 and a Fellow of The World Academy of Sciences since 2015. He has been a Highly Cited Researcher in Engineering continuously for some ten years according to Clarivate Web of Science.