Link prediction in complex networks via modularity-based belief propagation*

Darong Lai(赖大荣)[1,2,†], Xin Shu(舒欣)[3], and Christine Nardini[4,5]

[1] School of Computer Science and Engineering, Southeast University, Nanjing 211189, China
[2] Key Laboratory of Computer Network and Information Integration of Ministry of Education, Southeast University, Nanjing 210096, China
[3] College of Information Science and Technology, Nanjing Agricultural University, Nanjing 210095, China
[4] CNR-IAC "Mauro Picone" Via dei Taurini 19, 00185 Roma, Italy
[5] Personalgenomics, Via delle Grazie, Verona, Italy

(Received 26 August 2016; revised manuscript received 25 November 2016; published online 13 February 2017)

Link prediction aims at detecting missing, spurious or evolving links in a network, based on the topological information and/or nodes' attributes of the network. Under the assumption that the likelihood of the existence of a link between two nodes can be captured by nodes' similarity, several methods have been proposed to compute similarity directly or indirectly, with information on node degree. However, correctly predicting links is also crucial in revealing the link formation mechanisms and thus in providing more accurate modeling for networks. We here propose a novel method to predict links by incorporating stochastic-block-model link generating mechanisms with node degree. The proposed method first recovers the underlying block structure of a network by modularity-based belief propagation, and based on the recovered block structural information it models the link likelihood between two nodes to match the degree sequence of the network. Experiments on a set of real-world networks and synthetic networks generated by stochastic block model show that our proposed method is effective in detecting missing, spurious or evolving links of networks that can be well modeled by a stochastic block model. This approach efficiently complements the toolbox for complex network analysis, offering a novel tool to model links in stochastic block model networks that are fundamental in the modeling of real world complex networks.

Keywords: link prediction, complex network, belief propagation, modularity

PACS: 89.75.Hc, 02.50.Tt, 89.20.Ff **DOI:** 10.1088/1674-1056/26/3/038902

1. Introduction

Networks are powerful tools to describe numerous complex social, biological, and information systems, where nodes are units of the system and links between nodes indicate their interactions or relations.[1] For example, an urban bus traffic system can be modeled by different types of networks for further analysis.[2] The analysis of the functions and dynamics of complex systems can then be conducted through network analysis, from network structure to network dynamics.[3–6] Due to the limited knowledge of a complex real system represented by a network, data collected are always partial or blended with noise. For example, 80% of the molecular interactions in yeast[7] and more than 99% in humans[8,9] are still unknown. Similarly, informant inaccuracy[10] or sampling biases[11] in social sciences often lead to social networks constructed with missing data or noise. In addition, laboratory experiments or manual judgments to validate links are often costly. Fortunately, such costs, unacceptable for blind checking of all possible interactions, can be sharply reduced by focusing on the most likely interactions predicted by efficient computational algorithm on the basis of known (observed) interactions.

Based on observed links and/or attributes of nodes in a network, link prediction attempts to estimate the likelihood of the existence of a link between two nodes. Reliable link prediction algorithms can be used to identify missing links, to predict evolving links that may appear in the future, to filter out spurious links in a noisy network, and more. Similarity-based link prediction is a formalism that associates the likelihood of two disconnected nodes to be connected to their similarity: more similar nodes are supposed to have higher likelihoods of being connected. If nodes' attributes can be extracted efficiently, link prediction can be cast in the frame of calculating similarities between pairs of nodes in the Euclidean metric space.[12] Without involving nodes' attributes but with the sole network structure, Fouss *et al.* found that a node in a network could be expressed as a vector in a Euclidean space that preserves Euclidean commute-time distance, and that the similarity between two nodes was the inner product of nodes' corresponding vectors.[13] Actually, nodes' similarities are not necessarily calculated by first mapping nodes into vectors when only the topological structure of a network is available. Topological similarity measures, designed to capture the similarity between two nodes via the sole network structure, give rise to a broad variety of structural similarity based methods for link prediction in complex networks,[14] such as methods based on counting common neighbors.[15–17]

*Project supported by the National Natural Science Foundation of China (Grants No. 61202262), the Natural Science Foundation of Jiangsu Province, China (Grants No. BK2012328), and the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grants No. 20120092120034).

†Corresponding author. E-mail: darong.lai@gmail.com

© 2017 Chinese Physical Society and IOP Publishing Ltd

Methods not relying on structural similarity include maximal likelihood methods[18–20] by finding the best fit graph(s) (with pre-specified local structures such as blocks, motifs, hierarchy, and so on) to the observed networks and then scoring pairs of nodes by likelihoods, and probabilistic model methods[21–23] that use parameterized probabilistic models to derive the probability of the existence of a link. Although various link prediction methods are continuing to be designed, in this paper we focus on the structural similarity based link prediction methods, for their intuitiveness, driving interest in the community.

These methods can be roughly classified into three major classes:[14] global, local, and intermediate. The global similarity methods use the whole network topological information to compute nodes similarity. For example, the Katz index counts all paths connecting two nodes with more weights on shorter paths.[24] Other global similarity methods include random walk with restart[14] and average commute time index,[13] to name a few. Global methods need information of the whole network structure and output results with higher performance and computational complexity compared with local and intermediate approaches. Local similarity methods use only local structure information to define node similarity. For example, common neighbor (CN) counts the number of common neighbors between two nodes that are more likely to have a link if they share more common neighbors.[15] With different normalizations, the extension of CN includes the Salton[25] and Jaccard index. Building on the mechanism that generates scale-free networks without growth,[26] the preferential attachment index (PA) computes the probability of a link between two nodes proportional to their degrees. Compared to CN and its variants, PA does not require information on the neighbors of each node and thus reduces the computational complexity. With information on neighbors and their degrees, the Adamic–Adar (AA)[16] and resource allocation (RA) indices[17] compute structural similarity by assigning less-connected nodes more weights. AA and RA indices are similar in essence but weight differently large degree nodes. Finally, the intermediate approaches between global and local include local path index[27] and its variants, which are methods based on local random walks[14] and give a good tradeoff between performance and complexity. More details on various state-of-the-art link prediction methods can be found summarized in an excellent survey.[14]

An important feature of the link prediction method lies in their performances being associated to the mechanism(s) underlying networks' formation.[14,18,19,28–31] If the principle of a link prediction method is consistent to the network formation mechanism, the method performs well providing accurate predictions. In contrast, if a link prediction method cannot capture well the underlying link formation mechanism, it will often not give satisfactory predictions. For example, the hierarchical random graph based link prediction framework with the assumption of hierarchical organization of networks tends to have high prediction accuracy if the observed network organizes hierarchically.[18] Similarly, if the observed network is organized into blocks[32,33] or distinct roles, the framework based on the stochastic block model (SBM)[34] will provide accurate link predictions.[19] Inspired by this fact, Wang *et al.* viewed link addition in network evolution as a kind of link prediction algorithm and employed likelihood analysis to judge which mechanism is better among a given series of network formation mechanisms for explaining the evolution of a network,[28] while Zhang *et al.* applied the likelihood analysis and link prediction methods to quantitatively measure how multiple mechanisms contribute to a network formation.[29] In directed network cases, Zhang *et al.* found that a non-observed link would be more likely to exist if the link generates more potential-definable subgraphs (such as Bi-fan structure consisting of 4 nodes and 4 directed links) deduced by the potential theory with clustering and homophily mechanisms.[30]

When nodes in a network are partitioned into blocks and the probability that two nodes are connected depends only on the blocks they belong to, SBM is well suited for capturing the network organization mechanisms. Actually, SBM is a powerful model for generating a wide variety of networks, which can capture ubiquitous and fundamental properties of real world networks, overarching the more specific models presented above: modular or block structure, core-periphery, hierarchical, multipartite structure, and so on. SBM generates networks using as the link formation mechanism the probability of the existence of a link between two nodes that depends only on the node's block memberships. However, nodes in a group/block are not always homogenous but often heterogeneous, which is a feature found in most empirical network data.[1,3,35] Therefore, we deem reasonable to model the process of real-world network formation by considering nodes' specific characteristics. According to this observation, we propose to compute the structural similarity between two nodes by considering information on both block structures and node degrees. The method applies a modularity-based belief propagation algorithm[36] to recover the underlying block structure of a network and then calculates the connecting probability distribution over blocks. We name this method the block model preferential attachment index (BMPA).

In this paper, we first describe the problem of link prediction in a network and introduce several state-of-the-art similarity based methods as baselines. Then we propose our novel method for link prediction and review the modularity-based belief propagation algorithm for block structure recovering. Testing and comparisons between BMPA and baseline methods on a set of real-world networks and synthetic networks generated by SBM are finally shown.

2. Problem setting and baseline methods

We focus on link prediction in an undirected and unweighted network $G(V, E)$ with no self-connections nor multiple links, where V is the set of nodes and E the set of links. For missing link prediction, the set of links E is randomly divided into two sets: the training set E^{T} used for similarity calculation, and the testing set E^{P} for performance testing. Links in E^{P} in this case are not allowed to be included in E^{T}, that is, $E^{\mathrm{T}} \cap E^{\mathrm{P}} = \emptyset$ and $E^{\mathrm{T}} \cup E^{\mathrm{P}} = E$. Link set partition is here constrained to keep the induced network $G\ (V,\ E^{\mathrm{T}})$ connected. For spurious link prediction, extra links are added randomly to E. As a results, all added random links form the testing set E^{P}, and the training set in this case is $E^{\mathrm{T}} = E \cup E^{\mathrm{P}}$. The aim of spurious link prediction is to filter out from $G\ (V,\ E^{\mathrm{T}})$ the connected pairs that are also in the testing set E^{P}, while the aim of missing link prediction is to uncover unconnected pairs in $G\ (V,\ E^{\mathrm{T}})$ that are otherwise connected pairs in E^{P}. Evolving links predict links that may appear in the future in a network changing over time. To mimic this discovery, the ability of predicting evolving links can be revealed by the ability of filtering out randomly added links given that the observed network is the original network $G(V, E)$, which is different from that of missing or spurious link prediction. In such a case, the training set becomes $E^{\mathrm{T}} = E$, and E^{P} is the set of randomly added links.

To quantify the ability of methods to predict links, we use the area under the receiver operating characteristic curve (AUC).[37] Similarity based link prediction associates node similarity to the likelihood of the link existence. By computing similarities of all pairs in $U - E^{\mathrm{T}}$ (here U is the set of all possible node pairs in a network), i.e., the likelihoods of link existence of non-observed links in $U - E^{\mathrm{T}}(U - E \cup E^{\mathrm{P}})$, link prediction methods provide a ranked list of the likelihoods for missing link prediction. With this ranked list, the AUC value of a method in missing link prediction is the probability that a randomly picked missing link in E^{P} has a higher rank than a randomly picked nonexistent link in $U - E$. To ease the calculation, we here adopted the AUC computing method[14,17] of calculating the score of each non-observed link (similarity between its two ends) instead of giving the ordered list. At each step, a missing link in E^{P} and a nonexistent link in $U - E$ are randomly picked and their scores are compared. If among K independent comparisons, there are K_1 randomly picked missing links with a higher score and K_2 links with the same score, the AUC value is

$$\mathrm{AUC} = \frac{K_1 + 0.5K_2}{K}. \tag{1}$$

AUC varies generally between 0.5 and 1, with AUC = 0.5 indicating random guesses and AUC = 1 indicating the perfect ability of the method to predict missing links.

Similarly, the AUC value of a method for spurious or evolving link prediction can be calculated by comparing the scores between a randomly picked spurious link in E^{P} and a randomly picked link in set E. In spurious or evolving link prediction, K_1 in Eq. (1) indicates the number of randomly picked spurious links in E^{P} with a score lower than the one produced by a randomly picked link in E, and K_2 is the times that they have the same score. Therefore, the AUC value for spurious or evolving link prediction can be interpreted as the probability that a randomly picked spurious link in E^{P} has a lower rank than a randomly picked existent link in E.

For a given node pair i and j, link prediction methods compute the similarity score S_{ij} between the nodes by referring to the structural information in the training set. Four well-known local similarity based link prediction methods are used as a baseline for performance comparison. They are chosen for their effectiveness in link prediction and for the direct or indirect usage of node degree information. The global methods are not chosen due to their high computational complexity and intermediate methods such as local path index are avoided for the complexity of tuning parameters.

With d_i the degree of node i and $N(i)$ the set of its neighbors, the CN method [15] computes the similarity score between nodes i and j by counting their shared neighbors

$$S_{ij}^{\mathrm{CN}} = |N(i) \cap N(j)|, \tag{2}$$

where $|\cdot|$ represents the number of elements in a set, and $N(i) \cap N(j)$ the set of common neighbors of nodes i and j.

The PA method[26] assumes that the probability of a link connecting nodes i and j is proportional to their degrees d_i and d_j, and computes the score as

$$S_{ij}^{\mathrm{PA}} = d_i \times d_j. \tag{3}$$

Compared to CN, PA does not require the information on nodes' neighborhood, and thus has lower computational complexity.

Different from CN that gives equal weights to each neighbor of a node, the AA method [16] assigns more weights to less-connected neighbors and computes the similarity score as

$$S_{ij}^{\mathrm{AA}} = \sum_{z \in N(i) \cap N(j)} \frac{1}{\log d_z}. \tag{4}$$

The resource allocation dynamics motivated method, RA,[17] also uses unequal weights when counting common neighbors as AA, but penalizes the high-degree neighbors more heavily. The similarity score computed by RA is

$$S_{ij}^{\mathrm{RA}} = \sum_{z \in N(i) \cap N(j)} \frac{1}{d_z}. \tag{5}$$

3. BMPA similarity and modularity-based belief propagation

Nodes in a network are often organized into groups (or blocks) according to distinct mechanisms. For example, people often establish social relations with others according to

their age, gender, professions, ethnicity, and so on. As indicated above, when predicting the existence of links in such kinds of networks, it is crucial to reflect this underlying link formation mechanism in the prediction method. SBM can well capture such a ubiquitous and fundamental property. In SBM, the likelihood or probability that two nodes are connected depends only on the groups they belong to. Therefore, in the frame of SBM, all nodes in one group have a common probability to connect to all nodes in another group. We therefore simply called the probability that two nodes are connected as the probability that the groups they belong to are connected, to reflect such a characteristic of SBM that all nodes in the same group are treated equally. However, in real-world networks, all nodes in a group are not always homogenous. They have their own characteristics or roles, simply and directly characterized by node degree. Thus, to well capture nodes' similarity or the likelihood of a link existence, we combine these two factors.

Denote $\boldsymbol{A}$ as the adjacency matrix of a network, where $A_{ij} = 1$ if there exists a link between nodes i and j, otherwise$A_{ij} = 0$. The degree d_i is the number of links associated with node i, i.e., $d_i = \sum_j A_{ij}$. Suppose a network has q groups. Let m be the number of links of the network and $g_i = r$ $(1 \le r \le q)$ the group membership of node i. The number of links between groups r and s is $m_{rs} = \sum_{rs} A_{ij}\delta_{g_i,r}\delta_{g_j,s}$ (or twice the number of links if $r = s$) and the number of links associated with group r is $k_r = \sum_i d_i \delta_{g_i,r}$. Here $\delta_{g_i,r} = 1$ if $g_i = r$ and otherwise $\delta_{g_i,r} = 0$. The probability that two groups are connected (also known as the link distribution over groups, which is the probability that two nodes in these two groups are connected without considering the nodes' own characteristics but only group memberships) is defined as

$$p_{rs} = \frac{m_{rs}}{k_r k_s}. \tag{6}$$

When combining the connected probability of node groups with node degree, the new similarity index between nodes i and j, BMPA, is defined as

$$S_{ij} = p_{rs} d_i d_j = \frac{m_{rs}}{k_r k_s} d_i d_j. \tag{7}$$

The basic idea of BMPA is similar to that of the degree-corrected block model that takes into account degree heterogeneity to partition networks.[38] However, the connecting probability is modeled in a degree–block model using a Poisson distribution with S_{ij} as its mean, while in BMPA S_{ij} is used as similarity measure for ranking the likelihood of the existence of a link.

To compute node similarity with Eq. (7), group/block structure must be recovered first. A recently proposed modularity-based belief propagation (Modbp) approach[36] for finding the significant partition of a network can be adopted. Modbp is very efficient and works by iterative updates of two equations to find a significant partition of a network. If $v_t^{i\to j}$ denotes the belief or a message sent from node i to its neighbor node j, which is an estimate of the marginal probability that node i belongs to community t $(g_i = t)$ based on the interactions between node i and its neighbor nodes $k \ne j$, and $v_t^{(i)}$ is the marginal probability that node i belongs to community t based on the interactions between node i and all its neighbors, the two equations to update the beliefs are

$$v_t^{(i\to j)} = \frac{1}{Z_{i\to k}} \exp\Bigg[-\frac{\beta d_i}{2m}\theta_t + \sum_{k\in\partial i\setminus j} \log(1 + v_t^{(k\to i)}(\exp(\beta) - 1))\Bigg], \tag{8}$$

$$v_t^{(i)} = \frac{1}{Z_i} \exp\Bigg[-\frac{\beta d_i}{2m}\theta_t + \sum_{k\in\partial i} \log(1 + v_t^{(k\to i)} \times (\exp(\beta) - 1))\Bigg]. \tag{9}$$

Here $\theta_t = \sum_{k=1}^{n} d_k v_t^{(i)}$ is an external field acting on nodes in group t, and ∂i is the set of neighbors of node i. $\frac{1}{Z_{i\to k}}$ and $\frac{1}{Z_i}$ are normalized terms for Eqs. (8) and (9), respectively. Given initial guesses of $v_t^{(i\to j)}$ and $v_t^{(i)}$, and a fixed number of communities q, Modbp iteratively updates these two equations until convergence. Then node i is assigned to community t if $v_t^{(i)}$ is the maximum of the marginal probabilities for node i. If there is more than one maximum, ties are broken randomly. It should be pointed out that Modbp can also be applied to uncover the group structure of a given network if the number of groups in the network is not known in advance. By monitoring the fluctuations of so called retrieval modularity,[36] Modbp automatically determines the optimal number of groups in a network. Recent work on a Modbp variant reinforces this potential of finding the correct number of groups in a network with unknown group structure.[39]

4. Experiments

In this section, we test BMPA for link prediction on six real-world networks and synthetic networks generated by stochastic block model, and compare it with four high performance state-of-the-art link prediction methods.[14]

4.1. Results on real-world networks

Comparisons between BMPA and other state-of-the-art methods are performed on six real-world networks. The six networks are chosen for their usefulness in revealing different performance behaviors among BMPA and the four baseline methods as popular benchmarks in this type of research, thus also allowing an easy comparison with current literature. These networks are: (i) zcNet, a well-known social network of a karate club at an American university observed by Zachary in the early 1970s,[40] with club members as nodes and their social interactions as links; (ii) pblgNet, a network

of political blogs whose nodes are blogs about American politics and links the web hyperlinks between them;[41] (iii) dphNet, a social network of bottlenose dolphins living in Doubtful Sound in New Zealand,[42] whose nodes are dolphins and the links between nodes indicate that the dolphin pairs were seen together more often than expected by chance; (iv) sfiNet, a collaboration network of scientists at the Santa Fe Institute indicating coauthorship between these scientists;[32] (v) FWNet, a network of a food web in Florida Bay during the wet season,[43] representing carbon flow between species in the Bay; (vi) Yeast, a protein–protein interaction network in budding yeast[44] whose nodes are proteins and links their interactions. If networks are directed in their original form, their arcs are converted into undirected links. For disconnected networks, their largest connected components are used. Table 1 summarizes the basic topological structure information of these networks. The last row of Table 1 reports the number of groups in these networks, known in advance or determined by Modbp[36] or its variant.[39]

Table 1. The basic topological structure information of the six testing real-world networks. Here n is the number of nodes, m is the number of links, c represents the average degree, and q is the number of groups used in BMPA.

networks	zcNet	pblgNet	dphNet	sfiNet	FWNet	Yeast
n	34	1222	62	118	128	2224
m	78	16714	159	200	2075	6609
c	4.59	27.36	5.13	3.39	32.42	5.94
q	2	2	4	8	3	13

The experiments were conducted by evaluating the AUC performance on missing, evolving, and spurious link predictions. We formed the training and testing link sets of a network by randomly removing or adding a fraction of the total links in the network. The fraction of links removed was determined to keep all networks connected and at least half of the total links in the networks were removed if possible, while the maximal fraction of links added was equal to 1.

Figure 1 shows the comparisons between BMPA and other baseline methods on zcNet and pblgNet. For missing link prediction on zcNet, BMPA achieved the best performance provided that the fraction of removed links was not larger than 0.3. When the fraction exceeded 0.3, BMPA was still better than RA, AA, and CN, but worse than PA (Fig. 1(a)). A similar phenomenon occurred when predicting spurious links on zcNet if the fraction of links was not greater than 0.3 (Fig. 1(c)). When predicting evolving links on zcNet, BMPA was again the best choice (Fig. 1(b)). The AUC values of all methods except PA decreased with the fraction of links removed, while those of PA increased (Fig. 1(a)). In this case, preferential attachment dominated the formation of links in these modified networks. As a consequence, until there was a meaningful block structure in the network, taking into account both block connecting mechanism and node degree would improve performance, otherwise it became misleading. This was confirmed again from the results of link prediction on pblgNet. Both BMPA and PA performed well but BMPA was better than PA (the bottom panel of Fig. 1). Although RA, AA, and CN also behaved as well as BMPA and PA on evolving and spurious link prediction, they performed worse when predicting missing links on pblgNet.

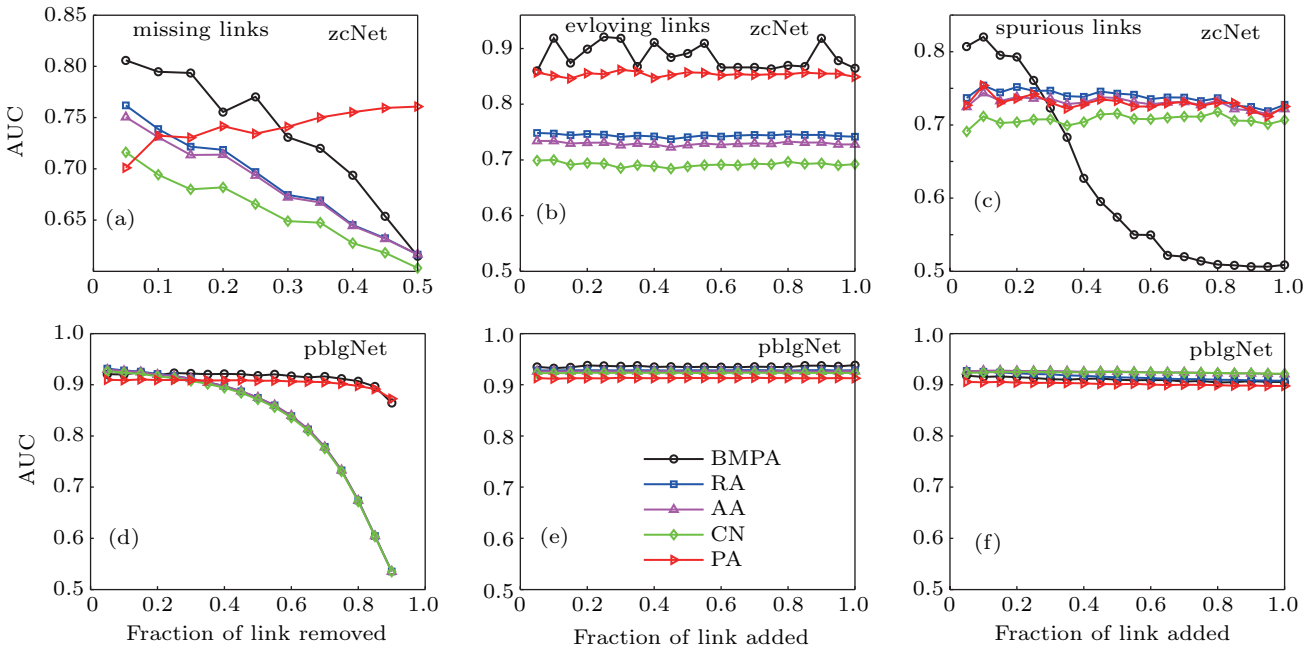


Fig. 1. (color online) Comparisons on the performance of link prediction in zcNet ((a)–(c)) and pblgNet ((d)–(f)), where BMPA was compared with RA, AA, CN, and PA. For missing link prediction, a fraction of links from the network were randomly removed but the network was kept connected, while for evolving or spurious link prediction a fraction of links were randomly added on the network but with different observed networks. Each point is averaged by repeating the experiments 100 times.

As shown in Fig. 2, for missing link prediction on dphNet and sfiNet, the performance of BMPA was comparable to RA, AA, and CN, and all are much better than PA, meaning that the preferential attachment mechanism could not well explain globally (PA) the underlying link formation process while it could locally (BMPA). For evolving link prediction on these two networks, BMPA performed best and PA performed worst (the middle panel of Fig. 2). These results confirm the fact that preferential attachment could not explain globally the underlying process of link formation in these two networks. If there was a clear block structure in a network, integrating information from the block structure would more accurately explain the underlying mechanism of link formation. This is why BMPA performed worse than RA, AA, and CN in both missing and spurious link prediction on these two networks but performed much better in evolving link prediction, since the observed network remained the original network in the case of evolving link prediction and the block structure was not destroyed. Although the preferential attachment mechanism from PA could not well explain globally the link formation in these networks, it could play a role at the local level if networks had block structure, i.e., preferential attachment mechanism from BMPA could well explain locally the link formation in this case.

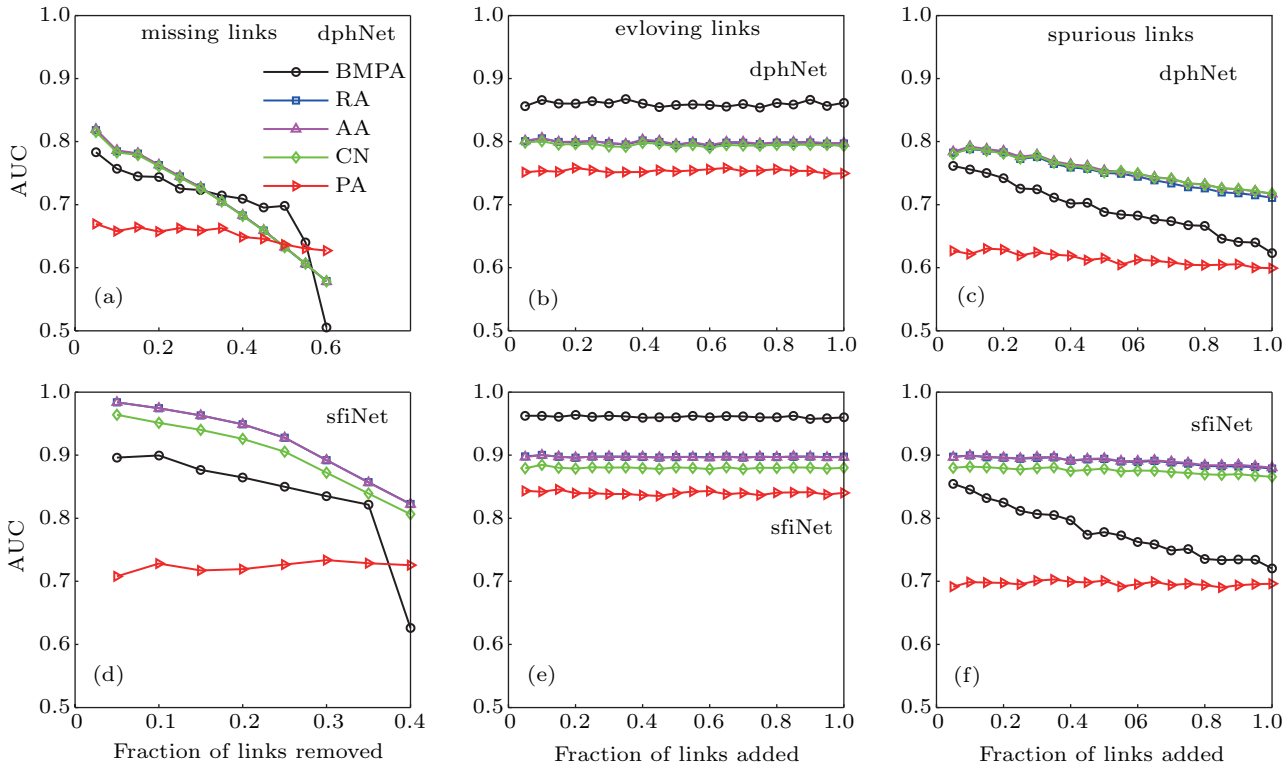


Fig. 2. (color online) Comparisons on the performance of link prediction in dphNet ((a)–(c)) and sfiNet ((d)–(f)), where BMPA was compared with RA, AA, CN, and PA. Each point is averaged by repeating the experiments 100 times.

The experiments above would suggest that PA is not the best choice for link prediction in networks. To confirm the association between good performance of link prediction and the appropriate link formation model, we further tested two additional networks: FWNet and Yeast. The results are shown in Fig. 3. It can be seen that for all of the missing, evolving, and spurious link prediction, PA was the best choice. Even if BMPA performed better than RA, AA, and CN on these two networks in most cases, the link formation process of these two networks appears to be well-explained globally by preferential attachment.

Complementarily, we conducted experiments on a series of synthetic networks with determined internal structure, to confirm the expectation that BMPA would perform very well in this case (Section 4.2).

As a final experiment on the real-world networks, we compared BMPA with a classical likelihood method for link reliability estimation[19] due to the shared SBM model used in the modeling network. Since the link reliability method (LRM) needs to sample a large number of possible partitions, it is a highly time-consuming method. In contrast, BMPA uses the consensus of several good partitions of a network to derive the link distribution over groups, and is thus scalable and efficient. Although it would be desirable to offer the comparison of BMPA with LRM together with other baseline methods in a single figure, the high computational costs of LRM makes it prohibitive to produce results on pblgNet and Yeast in the form adopted in Figs. 1 and 3 due to the large number of nodes in these two networks. As a consequence, we separate the comparison with LRM from the ones with the four other baseline

methods and provide here the comparison with LRM in tabular form, as routinely done in most of the literature of link prediction in complex networks,[14,31] for the sake of readability on BMPA results. Such a scheme for comparison is also reasonable, as it split the comparisons into two intuitive types: the ones with similarity based methods and the ones sharing SBM modeling networks.

Table 2 shows quantitatively the comparisons between BMPA and LRM in predicting missing links on the six aforementioned real-world networks, while Table 3 gives the results on predicting spurious links. As seen from Table 2, the performance of BMPA in missing link prediction is much better than LRM in all cases except when sfiNet randomly removed 10% of the total number of links. When predicting spurious links, BMPA performs better than LRM on all networks except on sfiNet and FWNet (see Table 3). A similar comparison can be made in the case of evolving link prediction (data not shown for conciseness). Although LRM has high computational complexity and performs poorly in these cases, it has an advantage over the structural similarity based methods as it can predict missing links when nodes are in separated components of a network. Such a property makes it quite suitable for predicting missing links in cases where the large portion of links is unknown as usually encountered in biological networks.

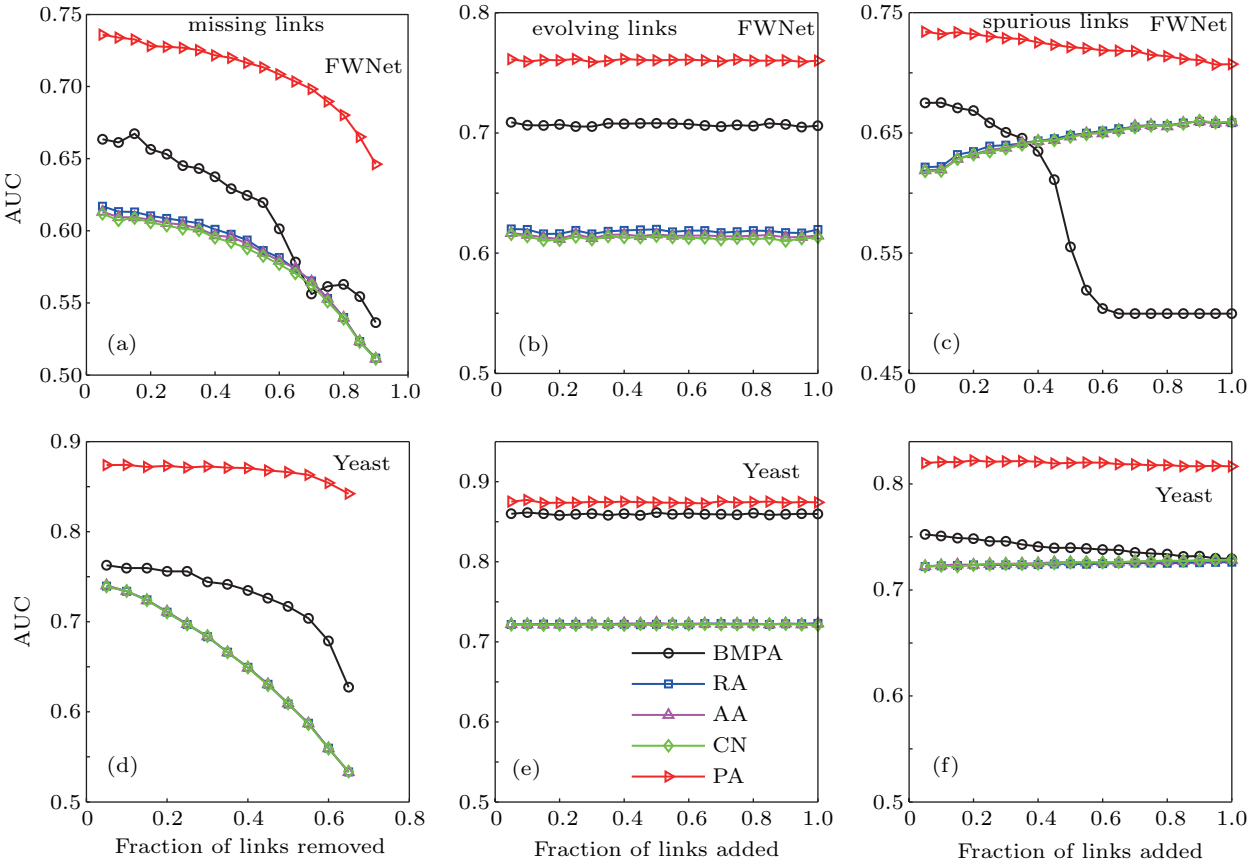


Fig. 3. (color online) Comparisons on the performance of link prediction in FWNet ((a)–(c)) and Yeast ((d)–(f)), where BMPA was compared with RA, AA, CN, and PA. Each point is averaged by repeating the experiments 100 times.

Table 2. Average accuracy of missing link prediction on real-world networks by BMPA and LRM. 10% and 20% of the total number of links were randomly removed. Numbers in bold are the maxima in each case and numbers in parenthesis are standard deviations of the AUC. Each of the results by LRM on the four smaller networks was averaged on 30 repetitions, while on pblgNet and Yeast the repetitions were reduced to 10 due to the high computational complexity of LRM.

Networks	10% links removed		20% links removed	
	LRM	BMPA	LRM	BMPA
zcNet	0.771 (0.090)	**0.797** (0.077)	0.697 (0.091)	**0.754** (0.082)
pblgNet	0.615 (0.007)	**0.920** (0.149)	0.623 (0.012)	**0.919** (0.113)
dphNet	0.532 (0.061)	**0.757** (0.066)	0.516 (0.053)	**0.744** (0.055)
sfiNet	**0.908** (0.059)	0.899 (0.042)	0.848 (0.079)	**0.865** (0.037)
FWNet	0.601 (0.019)	**0.661** (0.028)	0.604 (0.017)	**0.657** (0.025)
Yeast	0.595 (0.004)	**0.760** (0.014)	0.560 (0.009)	**0.756** (0.010)

Table 3. Average accuracy of spurious link prediction on real-world networks by BMPA and LRM. 10% and 20% of the total number of links were randomly added. Numbers in bold are the maxima in each case and numbers in parenthesis are standard deviations of the AUC. Each of the results by LRM on the four smaller networks was averaged on 30 repetitions, while on pblgNet and Yeast the repetitions were reduced to 10 due to the high computational complexity of LRM.

Networks \ Methods	10% links added		20% links added	
	LRM	BMPA	LRM	BMPA
zcNet	0.794 (0.062)	**0.820** (0.079)	0.786 (0.062)	**0.793** (0.055)
pblgNet	0.646 (0.047)	**0.915** (0.045)	0.675 (0.043)	**0.914** (0.045)
dphNet	0.458 (0.073)	**0.756** (0.057)	0.488 (0.061)	**0.742** (0.048)
sfiNet	**0.903** (0.036)	0.845 (0.044)	**0.878** (0.027)	0.825 (0.036)
FWNet	**0.789** (0.062)	0.675 (0.017)	**0.843** (0.033)	0.669 (0.020)
Yeast	0.576 (0.012)	**0.751** (0.010)	0.589 (0.011)	**0.748** (0.010)

Due to the predicting accuracy shown in this section and the high computational complexity of LRM, in the following section, we compared the performance of BMPA only with classical structural similarity based methods.

4.2. Results on synthetic networks

Synthetic networks in this section were generated by the stochastic block model. Given q groups of nodes with equal size, each pair of nodes i and j is connected with probability $p_{g_i g_j} = p_{\text{in}}$ if they are in the same group ($g_i = g_j$) and with probability $p_{g_i g_j} = p_{\text{out}}$ if they are in different groups ($g_i \neq g_j$). We here let $p_{\text{in}} > p_{\text{out}}$ and define $\varepsilon = p_{\text{out}}/p_{\text{in}}$. Therefore, the average degree c of a network generated is $c = (np_{\text{in}} + (q-1)p_{\text{out}})/q$. If $\varepsilon < (\sqrt{c}-1)/(\sqrt{c}-1+q)$, the network will have a detectable block structure,[36] otherwise no block structure can be recovered.

We generate three sets of synthetic networks with 1×10^3 nodes and average degrees c set to 3, 6, and 10. All generated networks have $q = 10$ groups. Consequently, the critical value ε^* $(= (\sqrt{c}-1)/(\sqrt{c}-1+q))$ of the three sets of networks are respectively 0.0682, 0.1266, and 0.1778. To generate networks with detectable block structure, we set ε to 0.05, 0.10, and 0.15, respectively.

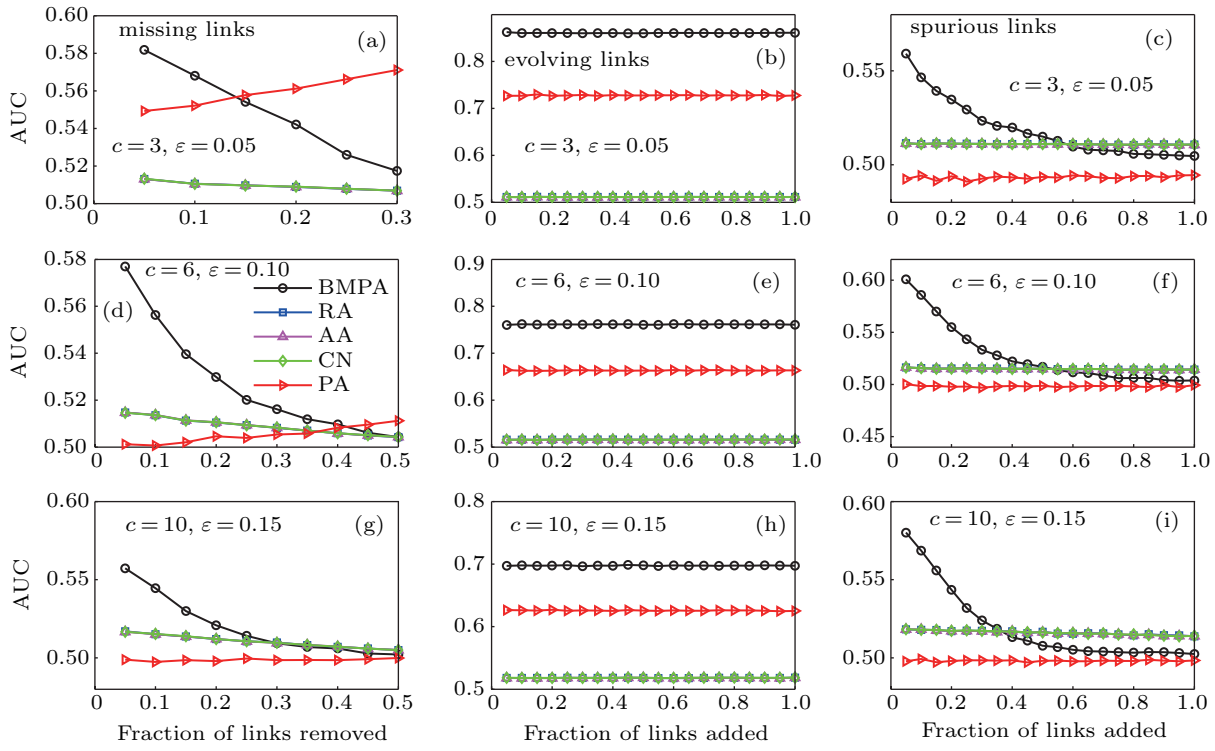


Fig. 4. (color online) Comparisons on the performance of link prediction in synthetic networks generated by the stochastic block model with 1×10^3 nodes, the number of blocks $q = 10$, and different average degrees c. BMPA was compared with RA, AA, CN, and PA. Performance of link prediction on networks with $c = 3$ ((a)–(c)), $c = 6$ ((d)–(f)), and $c = 10$ ((g)–(i)). For missing link prediction (left panel), a fraction of links were randomly removed from the network but the networks were kept connected, while for evolving and spurious link prediction (middle and right panel) a fraction of links were randomly added on the networks. Each point is averaged over 10 network instances by repeating the experiments 30 times.

Figure 4 shows the link prediction results of BMPA and baseline methods on these sets of networks. It can be clearly seen that BMPA performs much better on predicting missing, evolving, and spurious links of these SBM-type networks, which means that the underlying link formation mechanism of these networks is better captured by BMPA. This is con-

firmed on networks with different values of average degree c. In contrast, on all missing, evolving, and spurious link prediction, the AUC scores for RA, AA, and CN all approach 0.5. When more and more links are randomly removed or added, the internal block structure is destroyed thus decreasing the values of the corresponding AUC. In fact, randomly removing links in missing link prediction makes a network sparse, while randomly adding links in spurious link prediction makes a network dense, thus modifying the original block structure. For missing link prediction, it is interesting to see that PA performs better than BMPA and thus better than the other three baseline methods in sparser networks when more and more links are removed (Figs. 4(a) and 4(d)), while for spurious link prediction the phenomenon disappears. A similar situation occurred when predicting the missing link on zcNet (Fig. 1(a)). We hence speculate that in a very sparse network when there is no statistically meaningful block structure, preferential attachment will be more suitable for explaining the underlying link formation mechanism of the network. When there is a statistically meaningful block structure, the performance of BMPA is the best among all these methods, as shown for evolving link prediction in the middle panel of Fig. 4.

For missing or spurious link prediction, when more and more links were randomly removed or added, the block structure was gradually destroyed and thus the performance decreased with the fraction of links removed or added. While for evolving link prediction, the block structure was kept unchanged since randomly added links were not included in the training set of links. As a result, BMPA keeps performing at best with the fraction of added links in evolving link prediction. The reason is confirmed in Fig. 5, which shows how close the block partitions of the networks with randomly removed or added links are to the original network partition, quantified by normalized mutual information (NMI).[45] The higher the value of NMI is, the closer the partitions are. Clearly, the NMI value is larger when only a few links are removed or added (Fig. 5(a) or 5(b)). When more and more links are removed or added, the partitions on the remainder of the networks have lower consistency with the partition on the original network, leading to the AUC of BMPA decreases. In contrast, in the case of evolving link prediction, when links are randomly added, the partitions on the networks are the partition on the original network with an approximately horizontal line approaching to 1, as shown in Fig. 5(c). As proposed above, the reason in this case is that the internal block structure is unchanged.

Finally, we explored the robustness of BMPA with respect to the original network partition (block detection), as this is inferred and not given information. Block detection is achieved with Modbp[34] chosen for its computational efficiency and statistically significant block finding property, although other approaches could be used. When the real number of blocks in a network is not known a priori, Modbp or its variant are able to automatically compute a narrow range of potential partition around the true value, and outputs the consensus among the significant partitions produced for this narrow range of values. We here show that performances of BMPA are robust to the partition obtained by using Modbp.

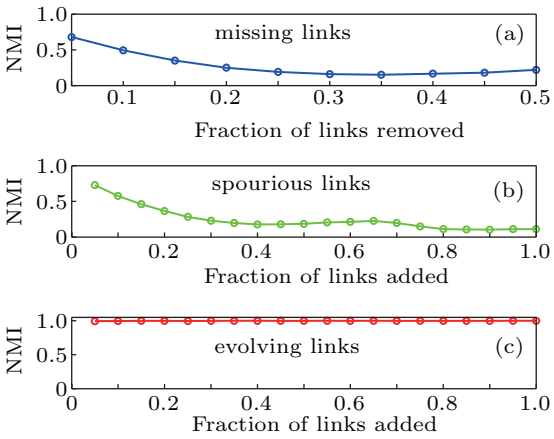


Fig. 5. (color online) The consistency between the block partitions obtained from networks with randomly removed or added links and the partition obtained from the original network. The consistency was measured by NMI with the fraction of links removed or added, and tested on networks with $c = 10$ and $q = 10$. (a) The consistency for randomly removing links in missing link prediction. (b) The consistency for randomly adding links in spurious link prediction. (c) The consistency for randomly adding links in evolving link prediction. Each point is averaged over 10 network instances by repeating the experiments 30 times.

Figure 6 shows the results of link prediction on networks with $c = 10$ when the networks were first partitioned into $q = 8$, 9, 11, or 12 blocks by Modbp. We found that when the number are around the real number ($q = 10$), the performance of BMPA on link prediction kept consistent with the one when networks were partitioned into $q = 10$ blocks, and the prediction is much better than the performance of RA and other baseline methods. Such robustness is due to the consistency of the partitions obtained by Modbp, confirmed from Fig. 7, where the NMI values between partitions obtained with different q around 10 and the partition of 10 blocks is high enough, provided the internal block structure is not destroyed, see Fig. 6(c). The approximately equal values of AUC (Fig. 6(c)) and NMI close to 1 (Fig. 7(c)) at q equal to 10, 11, and 12 indicate approximately perfect matches of partitions. Thanks to the nature of Modbp for significant block detection, when the number of blocks used for Modbp is larger than the real one, the partition obtained will keep highly consistent with real block structure since the extra groups in the partition consist of only a few nodes. Therefore, the consistency of partition from Modbp guarantees the robustness of BMPA to predict either missing, spurious, or evolving links in a network.

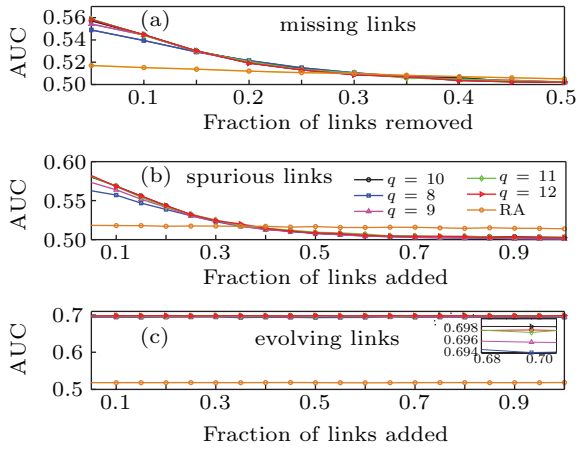


Fig. 6. (color online) The performance of link prediction on networks with $c = 10$ when different numbers of blocks were used in BMPA. (a) The performance of BMPA on missing link prediction. (b) The performance of BMPA on spurious link prediction. (c) The performance of BMPA on evolving link prediction, and the inset shows the different performance with different values of q. The number q of blocks used are the real number of blocks in the networks ($q = 10$) and numbers around the real one ($q = 8$, 9, 11, and 12). Each point is averaged over 10 network instances by repeating the experiments 30 times.

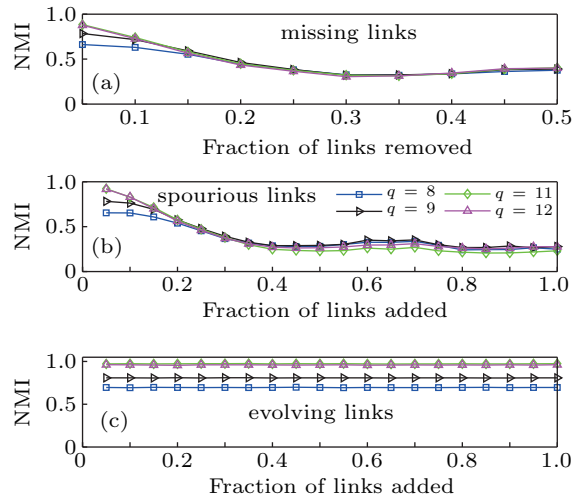


Fig. 7. (color online) The consistency between partitions obtained with different values of q and the partition with $q = 10$ blocks. The consistency is measured by NMI with the fraction of links removed or added, and the values of q around 10 ($q = 8$, 9, 11, and 12) are used for comparison. (a) The consistency of partitions in missing link prediction when links were randomly removed. (b) The consistency of partitions in spurious link prediction when links were randomly added. (c) The consistency of partitions in evolving link prediction when links were randomly added. Each point is averaged over 10 network instances by repeating the experiments 30 times.

5. Conclusion

In this paper we proposed a new method for link prediction in networks. It is based on a novel link likelihood computation index, BMPA. BMPA is designed to predict links in stochastic block model networks, ubiquitous and appropriate models of real-world complex networks, and assumes that the likelihood of two nodes being connected depends not only on the characteristics of nodes but also their block membership. Therefore, BMPA adopts an efficient significant block detection algorithm, Modbp, to first obtain the information of the underlying block structure of a network and then derives the probability of link distribution between any two blocks. Combining information of node degree with block connecting probability, BMPA connects the similarity between two nodes to the likelihood of the existence of a link between them. Comparisons with other state-of-the-art link prediction methods on various real-world networks and sets of synthetic networks generated by SBM confirm the good performance of BMPA in missing, evolving, and spurious link prediction in an SBM and assimilated real-world network.

When networks are generated by SBM, BMPA that captures the underlying link formation mechanism identifies the missing, evolving, and spurious links with high accuracy, improving on state-of-the-art SBM prediction methods. Such BMPA complements the toolkit of instruments needed to predict links when networks are appropriately modeled by stochastic block models.

Although in the current paper Modbp is adopted to recover blocks that are statistically significant, making BMPA robust to the fluctuations of block structure, it is not the only method to recover the block structure in a network. As a consequence, various block structure finding methods[46] can be integrated into the link prediction framework proposed here. Exploration of how different block structure finding methods influence prediction results when integrated into the framework proposed here is the object of future work.

References

[1] Newman M E J 2010 *Networks: An Introduction*, 1st edn. (Oxford: Oxford University Press) p. 1
[2] Ren T, Wang Y F, Liu M M and Xu Y J 2016 *Chin. Phys. B* **25** 020101
[3] Boccaletti S, Latora V, Moreno Y, Chavez M and Hwang D U 2006 *Phys. Rep.* **424** 175
[4] Ruan Y R, Lao S Y, Xiao Y D, Wang J D and Bai L 2016 *Chin. Phys. Lett.* **33** 028901
[5] Duan J M, Shang M S, Cai S M and Zhang Y X 2015 *Acta Phys. Sin.* **64** 200501 (in Chinese)
[6] Xu X L, Liu C P and He D R 2016 *Chin. Phys. Lett.* **33** 048901
[7] Yu H, Braun P, Yildirim M A et al. 2008 *Science* **322** 104
[8] Stumpf M P H, Thorne T, de Silva E, Stewart R, An H J, Lappe M and Wiuf C 2008 *Proc. Natl. Acad. Sci. USA* **105** 6959
[9] Amaral L A N 2008 *Proc. Natl. Acad. Sci. USA* **105** 6795
[10] Butts C 2003 *Soc. Netw.* **25** 103
[11] Kossinets G 2006 *Soc. Netw.* **28** 247
[12] Schifanella R, Barrat A, Cattuto C, Markines B and Menczer F 2010 *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, February 3–6, 2010, New York, USA, p. 271
[13] Fouss F, Pirotte A, Renders J M and Saerens M 2007 *IEEE T. Knowl. Data En.* **19** 355
[14] Lü L and Zhou T 2011 *Physica A* **390** 1150
[15] Newman M E J 2001 *Phys. Rev. E* **64** 025102
[16] Adamic L A and Adar E 2003 *Soc. Netw.* **25** 211
[17] Zhou T, Lü L and Zhang Y C 2009 *Eur. Phys. J. B* **71** 623
[18] Clauset A, Moore C and Newman M E J 2008 *Nature* **453** 98
[19] Guimera R and Sales-Pardo M 2009 *Proc. Natl. Acad. Sci. USA* **106** 22073
[20] Pan L, Zhou T, Lü L and Hu C K 2016 *Sci. Rep.* **6** 22955
[21] Taskar B, Wong M F, Abbeel P and Koller D 2004 *Proceedings of Neural Information Processing Systems*, December 8–13, 2003, Vancouver and Whistler, British Columbia, Canada, p. 659

[22] Yu K, Chu W, Yu S, Tresp V and Xu Z 2007 *Proceedings of Neural Information Processing Systems*, December 4–7, 2006, Vancouver, British Columbia, Canada, p. 1657
[23] Li Y J, Yin C, Yu H and Liu Z 2016 *Acta Phys. Sin.* **65** 200501 (in Chinese)
[24] Katz L 1953 *Psychmetrika* **18** 39
[25] Salton G and McGill M J 1983 *Introduction to Modern Information Retrieval* (Auckland: McGraw-Hill) p. 1
[26] Xie Y B, Zhou T and Wang B H 2008 *Physica A* **387** 1683
[27] Lü L, Jin C and Zhou T 2009 *Phys. Rev. E* **83** 046122
[28] Wang W Q, Zhang Q M and Zhou T 2012 *Europhys. Lett.* **98** 28004
[29] Zhang Q M, Xu X K, Zhu Y X and Zhou T 2015 *Sci. Rep.* **5** 10350
[30] Zhang Q M, Lü L, Wang W Q and Zhou T 2013 *PloS One* **8** e55437
[31] Lü L, Pan L, Zhou T and Zhang Y C 2015 *Proc. Natl. Acad. Sci. USA* **112** 2345
[32] Girvan M and Newman M E J 2002 *Proc. Natl Acad. Sci. USA* **99** 7821
[33] Shen Y, Ren G, Liu Y and Xu J L 2016 *Chin. Phys. B* **25** 068901
[34] Nowicki K and Snijders T A B 2001 *J. Am. Stat. Assoc.* **96** 1077
[35] Karrer B and Newman M E J 2011 *Phys. Rev. E* **83** 016107
[36] Zhang P and Moore C 2014 *Proc. Natl. Acad. Sci. USA* **111** 18144
[37] Hanely J A and McNeil B J 1982 *Radiology* **143** 29
[38] Karrer B and Newman M E J 2011 *Phys. Rev. E* **83** 016107
[39] Lai D, Shu X and Nardini C 2016 *J. Stat. Mech.* P053301
[40] Zachary W W 1977 *J. Anthropol. Res.* **33** 452
[41] Adamic L A and Glance N 2005 *Proceedings of the 3rd International Workshop on Link Discovery*, August 21–24, 2005, Chicago, USA, p. 36
[42] Lusseau D, Schneider K, Boisseau O J, Haase P, Slooten E and Dawson S M 2003 *Behav. Ecol. Sociobiol.* **54** 396
[43] Ulanowicz R E, Bondavalli C and Egnotovich M S 1998 Technical Report Ref. No. [UMCES] CBL 98-123
[44] Sun S, Ling L, Zhang N, Li G and Chen R 2003 *Nucleic Acids Res.* **31** 2443
[45] Danon L, Diaz-Guilera A, Duch J and Arenas A 2005 *J. Stat. Mech.* P09008
[46] Fortunato S 2010 *Phys. Rep.* **486** 75