Walk-modularity and community structure in networks

DAVID MEHRLE, AMY STROSSER and ANTHONY HARKIN

Network Science / Volume 3 / Issue 03 / September 2015, pp 348 - 360
DOI: 10.1017/nws.2015.20, Published online: 29 July 2015

Network Science **3** (3): 348–360, 2015.
doi:10.1017/nws.2015.20

Walk-modularity and community structure in networks

DAVID MEHRLE
Department of Mathematics, Cornell University, Ithaca, NY, USA
(e-mail: `dfm223@cornell.edu`*)*

AMY STROSSER
Department of Mathematics and Statistics, Villanova University, Villanova, PA, USA
(e-mail: `astrosse@villanova.edu`*)*

ANTHONY HARKIN
School of Mathematical Sciences, Rochester Institute of Technology, Rochester, NY, USA
(e-mail: `harkin@rit.edu`*)*

Abstract

Modularity maximization has been one of the most widely used approaches in the last decade for discovering community structure in networks of practical interest in biology, computing, social science, statistical mechanics, and more. Modularity is a quality function that measures the difference between the number of edges found within clusters minus the number of edges one would statistically expect to find based on some equivalent random graph model. We explore a natural generalization of modularity based on the difference between the actual and expected number of walks within clusters, which we refer to as walk-modularity. Walk-modularity can be expressed in matrix form, and community detection can be performed by finding the leading eigenvector of the walk-modularity matrix. We demonstrate community detection on both synthetic and real-world networks and find that walk-modularity maximization returns significantly improved results compared to traditional modularity maximization.

Keywords: *community detection, modularity*

1 Introduction

The problem of detecting community structure within networks has received considerable attention over the last decade, largely due to the rapidly increasing accessibility of large, real-world network data sets in many fields, including biology, computing, social sciences, statistical mechanics, and many more. A community, or cluster, may intuitively be thought of as a collection of nodes more densely connected to each other than with nodes outside the cluster. There are several theoretical and practical challenges in trying to detect community structure. For example, in real-world networks one usually does not know in advance how many separate communities are contained within the network or how many nodes comprise each of the communities.

Several popular and effective algorithms for community detection in networks rely on the concept of modularity in order to measure the quality of a partition

(Newman & Girvan, 2004; Newman, 2006a,b; Good et al., 2010; Porter et al., 2009; Reichardt & Bornholdt, 2006). The sensible observation motivating the modularity measure is that separate communities should have significantly fewer edges running between them than one would expect to find based on random chance. Equivalently, the number of edges connecting nodes within a community should be far greater than statistically expected relative to a particular random graph model. The modularity function is therefore defined as the difference between the edge density within groups of nodes minus the expected edge density of an equivalent null model network with the edges placed at random. Large positive values of modularity indicate the presence of a good partition of a network into communities, and many community detection algorithms are based on efficiently searching for partitions which maximize the modularity function (Newman, 2004; Clauset et al., 2004; Duch & Arenas, 2005; Blondel et al., 2008).

Insightful generalizations of modularity have been proposed according to different motifs, such as bipartite modularity (Barber, 2007), cycle modularity, and path modularity (Arenas et al., 2008). These generalizations are designed in order to uncover modules having certain structural properties, and are sensitive to the null models that are employed. In this paper, we explore the generalization of modularity based on walks. A walk on a network begins at a starting vertex and continues going along edges, which may be repeated, until it reaches a specified last vertex. The number of edges traversed is called the length of the walk. The modularity generalization is that the number of walks of a specified length connecting nodes within a community should be far greater than expected by random chance. Large, positive values of the walk-modularity function will indicate groups of nodes that are very tightly knit together by walks of a particular length. For walk length equal to one, the walk-modularity function reduces to the standard modularity measure, which just considers edges, and we refer to it as edge-modularity.

Network communicability is a concept that was introduced in order to account for many real-world situations where communication between a pair of nodes can occur along all possible routes between the nodes, and not just the shortest-path route (Estrada & Hatano, 2008, 2009; Estrada, 2011a,b; Estrada et al., 2012). In this concept, a weighted sum of all possible walks is used where shorter routes of communication are given larger weight than longer routes. When a factorial penalization is used for the weights, network communicability can be expressed as $G = e^A$, where $\mathbf{A}$ is the adjacency matrix of the network. Community detection can be performed based on the communicability matrix and the intuitive notion that nodes in a community should communicate better among themselves than with nodes outside the community. This notion is similar to the idea that motivates network modularity, except that modularity is a quality function that relies on comparing the network to some particular random null model, one which may or may not always be appropriate.

In the seminal paper of Newman (2006a), the modularity function was cast into a matrix formulation, yielding the so-called modularity matrix of a network. The modularity function can be maximized by computing leading eigenvectors of the modularity matrix, and high quality network partitions are obtained. For any specified walk length, we can define the walk-modularity matrix and can use it to find a good partition of a network into communities. The method of optimal modularity has several desirable features which have led to its widespread popularity. In particular, neither the number or sizes of the communities has to be specified in advance. These desirable features of the method of optimal modularity are retained when using the walk-modularity generalization.

2 Walk-modularity

Two nodes in a network are connected by a length ℓ walk if you can start at one node and traverse ℓ consecutive edges, possibly repeated, to reach the other node. A generalized notion of modularity can be based on the idea that a community will have a statistically unexpected number of walks between its nodes. Walk-modularity is therefore expressed as

$$Q_\ell = \text{(number of walks of length } \ell \text{ within communities)} - \text{(expected number of such walks)}.$$

Partitions of a network with large positive values of Q_ℓ will indicate groups of nodes that are very tightly knit together. Suppose a network contains n nodes and has an adjacency matrix denoted by $\mathbf{A}$. It is a well-known fact that powers of the adjacency matrix can be used to count the numbers of walks between pairs of nodes in a graph. The i, j entry of the matrix A^ℓ is the number of walks of length ℓ between vertices i and j.

In order to calculate the expected number of walks between nodes, we need to postulate an equivalent randomized network, or null model, with which to compare to the real network. We will employ the same null model described in the paper of Newman (2006a). In that work, an equivalent null model was considered where edges are placed entirely at random, subject to the constraint that the expected degree of each vertex in the null model is the same as the actual degree of the corresponding vertex in the real network. Let k_i be the degree of vertex i and let m be the total number of edges in the network. The expected number of edges between vertices i and j is then given by

$$P_{ij} = \frac{k_i k_j}{2m}.$$

Powers of the expected adjacency matrix, P, will count the expected number of walks in the null model. The i, j entry of the matrix P^ℓ is the expected number of walks of length ℓ between vertices i and j.

Define g_i to be the community to which vertex i belongs. The walk-modularity, for walks of length ℓ, can be written

$$Q_\ell = \frac{1}{2m_\ell} \sum_{i,j} \left[(A^\ell)_{ij} - (P^\ell)_{ij} \right] \delta(g_i, g_j) \,, \tag{1}$$

where $\delta(r, s) = 1$ if $r = s$ or zero otherwise. Walk-modularity only compares the actual number of walks versus the expected number of walks for vertices within the same community. We choose to normalize walk-modularity by

$$m_\ell = \frac{1}{2} \sum_{i,j} (A^\ell)_{ij} \,.$$

Note that for $\ell = 1$, the walk-modularity Q_1 reduces to the standard definition of modularity given in Newman & Girvan (2004), which we refer to as edge-modularity.

The walk length, ℓ, turns out to be a natural parameter in the clustering algorithm that provides the ability to control how tightly knit a community of nodes should be.

3 Spectral optimization of walk-modularity

To find community structure in a network, we wish to maximize the walk-modularity over all possible partitions of the network. A large number of community detection techniques have been based on maximizing edge-modularity, although the decision problem of finding a clustering with modularity exceeding a given value is **NP**-complete (Brandes et al., 2008). Clustering algorithms attempting to maximize modularity therefore focus on heuristics and approximations to approach the optimum value. There are many heuristic algorithms for maximizing edge-modularity, using techniques such as linear programming, vector programming, greedy agglomeration approaches, extremal optimization, and spectral optimization (Newman, 2004; Clauset et al., 2004; Duch & Arenas, 2005; Blondel et al., 2008; Brandes et al., 2008; Agarwal & Kempe, 2008). In what follows, we employ the spectral optimization procedure from Newman (2006a) adapted to work for walk-modularity, which proved to be the simplest optimization technique to implement while giving high quality results.

Closely paralleling the discussion in Newman (2006a,b), we first consider the problem of partitioning the network into just two groups of nodes. Define a length n index vector $\mathbf{s}$ such that

$$s_i = \begin{cases} +1 & \text{if vertex } i \text{ belongs to group 1,} \\ -1 & \text{if vertex } i \text{ belongs to group 2.} \end{cases}$$

Noting that the quantity $\delta(g_i, g_j) = \frac{1}{2}(s_i s_j + 1)$ is 1 if nodes i and j are in the same group, and 0 otherwise, we can write the walk-modularity as

$$\begin{aligned} Q_\ell &= \frac{1}{4m_\ell} \sum_{i,j} \left[(A^\ell)_{ij} - (P^\ell)_{ij}\right](s_i s_j + 1) \\ &= \frac{1}{4m_\ell} \sum_{i,j} \left[(A^\ell)_{ij} - (P^\ell)_{ij}\right] s_i s_j + \frac{1}{4m_\ell} \sum_{i,j} \left[(A^\ell)_{ij} - (P^\ell)_{ij}\right]. \end{aligned}$$

If we define the walk-modularity matrix as

$$\mathbf{B} = \mathbf{A}^\ell - \mathbf{P}^\ell$$

then

$$Q_\ell = \frac{1}{4m_\ell} \mathbf{s}^T \mathbf{B} \mathbf{s} + \frac{1}{4m_\ell} \sum_{i,j} B_{ij.} \qquad (2)$$

Since large, positive values of walk-modularity indicate the presence of a good division of the network, the goal is to obtain the index vector $\mathbf{s}$ which maximizes the value of Q_ℓ. This amounts to optimizing $\mathbf{s}^T\mathbf{B}\mathbf{s}$ since the second term in (2) does not depend on the choice of $\mathbf{s}$. Let $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$ be the eigenvalues of $\mathbf{B}$, with associated orthonormal eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$. Expressing $\mathbf{s}$ as a linear combination of the eigenvectors $\mathbf{s} = \sum_{i=1}^{n} a_i \mathbf{u}_i$ with $a_i = \mathbf{u}_i^T \mathbf{s}$, we obtain

$$\mathbf{s}^T \mathbf{B} \mathbf{s} = \left(\sum_{i=1}^{n} a_i \mathbf{u}_i^T \right) B \left(\sum_{i=1}^{n} a_i \mathbf{u}_i \right) = \sum_{i=1}^{n} a_i^2 \lambda_i.$$

Hence, the task of choosing $\mathbf{s}$ to maximize Q_ℓ is equivalent to choosing positive values a_i^2 to place as much weight as possible on the term in the sum with the largest positive eigenvalue, namely λ_1. Since, $a_1 = \mathbf{u}_1^T \mathbf{s}$, we choose $\mathbf{s}$ to be as close to parallel with $\mathbf{u}_1$ as possible. Thus, as in Newman (2006a), the choice of $\mathbf{s}$ is

$$s_i = \begin{cases} +1 & \text{if } u_i^{(1)} \geqslant 0 \\ -1 & \text{if } u_i^{(1)} < 0 \end{cases}, \tag{3}$$

where $u_i^{(1)}$ is the i-th entry of $\mathbf{u}_1$. Therefore, nodes are placed into either group 1 or group 2 depending on the sign of their corresponding entry in the leading eigenvector of the walk-modularity matrix.

This modularity maximization approach relies on finding the leading eigenvector of the walk-modularity matrix $\mathbf{B}$. An efficient way to accomplish this is to use power iteration, which requires repeatedly multiplying a vector $\mathbf{u}$ by the walk-modularity matrix. The product of $\mathbf{B}$ and a vector $\mathbf{u}$ may be written as

$$\mathbf{B}\mathbf{u} = \mathbf{A}^\ell \mathbf{u} - \mathbf{P}^\ell \mathbf{u} = \mathbf{A}(\mathbf{A}(\cdots(\mathbf{A}\mathbf{u}))) - \mathbf{P}(\mathbf{P}(\cdots(\mathbf{P}\mathbf{u}))).$$

Performing the multiplication in this manner, we see that $\mathbf{A}^\ell \mathbf{u}$ can be accomplished in $O(\ell m)$. The multiplication $\mathbf{P}^\ell \mathbf{u}$ can be done efficiently by recalling that $\mathbf{P}$ is defined as $\mathbf{k}\mathbf{k}^\mathbf{T}/2m$. The inner product $\mathbf{k}^T \mathbf{u}$ takes $O(n)$ time to evaluate, so evaluating $\mathbf{P}^\ell \mathbf{u}$ takes $O(\ell n)$ time to compute. In the power method, typically $O(n)$ iterations are needed. Therefore, the overall complexity is $O(\ell n^2)$ for sparse networks, and $O(\ell n^3)$ for dense networks.

In order to divide a network into more than just two communities, we will employ a recursive approach where we keep dividing groups in two until we find indivisible communities. In order to decide whether a particular group should be further divided, we must examine the change in walk-modularity that would result. Proceeding again as in Newman (2006a), we consider a group g with n_g nodes, and calculate the change in walk-modularity that would result from further division of the group into two pieces,

$$\begin{aligned} \Delta Q_\ell &= \frac{1}{2m_l} \left[\frac{1}{2} \sum_{i,j \in g} B_{ij}(s_i s_j + 1) - \sum_{i,j \in g} B_{ij} \right] \\ &= \frac{1}{4m_l} \left[\sum_{i,j \in g} B_{ij} s_i s_j - \sum_{i,j \in g} B_{ij} \right] \\ &= \frac{1}{4m_l} \sum_{i,j \in g} \left[B_{ij} - \delta_{ij} \sum_{k \in g} B_{ik} \right] s_i s_j \\ &= \frac{1}{4m_l} \mathbf{s}^T \mathbf{B}^{(g)} \mathbf{s}. \end{aligned} \tag{4}$$

In a recursive approach for dividing a network into multiple communities, maximizing the contribution to walk-modularity from subdivision of communities can be approached using the same leading eigenvector method as before, but using the matrix $\mathbf{B}^{(g)}$ at each step. Furthermore, a recursive community detection algorithm should refuse to make any subdivisions for which the change in walk-modularity is negative, which can be determined by explicitly calculating the value of ΔQ_ℓ at each step. Communities for which $\Delta Q_\ell \leqslant 0$ are called indivisible. In this manner, the spectral approach can be applied to divide a network into multiple indivisible communities without the need to specify in advance the number or sizes of the communities.

4 Examples

To demonstrate that walk-modularity is a very effective generalization of edge-modularity for finding communities, we applied the spectral optimization algorithm described in the previous section to both computer generated test cases and real-world networks. It is very important to emphasize that, because our focus is conducting a consistent comparison between edge-modularity and walk-modularity, only the leading eigenvector algorithm is used for the tests in Sections 4.1 and 4.2. No other enhancements for modularity maximization were included in this work. The only parameter which varies is the walk length ℓ. The synthetic test networks in Section 4.1 are randomly generated using either a variation on the standard Erdős-Rényi model of random networks or generated following the benchmark test described in Lancichinetti et al. (2008), which has previously been used to benchmark community-detection algorithms (Agarwal & Kempe, 2008; Berry et al., 2011; Fortunato, 2009; Newman, 2011). The real-world test networks examined in Section 4.2 are culled from the data in Lusseau et al. (2003); Zachary (1977).

4.1 Synthetic networks

Walk-modularity significantly outperforms edge-modularity on the synthetic networks we examine in this section. The first synthetic test shown is a modified Erdős-Rényi random network with $n = 500$ nodes with a K_{20} complete subgraph connected to it (Figure 1). The probability of an edge between two nodes in the random network seen in Figure 1 is 0.10. The probability of an edge connecting a node in the K_{20} subgraph to a node in the random network is 0.5. The average degree of nodes in the complete subgraph are roughly the same as the average degree of nodes in the random network. To consider a node as either misplaced or placed correctly, the two communities were defined to be the embedded K_{20} and the Erdős-Rényi random network. If the community detection algorithm placed a node from the complete subgraph in the same community as the rest of the network, or vice versa, the node is counted as misplaced. In other words, we are trying to compare how well the algorithm can find an embedded complete subgraph. In this test, we find that walk-modularity significantly outperforms edge-modularity in that simply increasing the length of walks considered from $\ell = 1$ to $\ell = 3$ results in a decrease from 50 misplaced nodes to merely 1 misplaced node. When $\ell = 4$, walk-modularity correctly places every single node into the two communities.

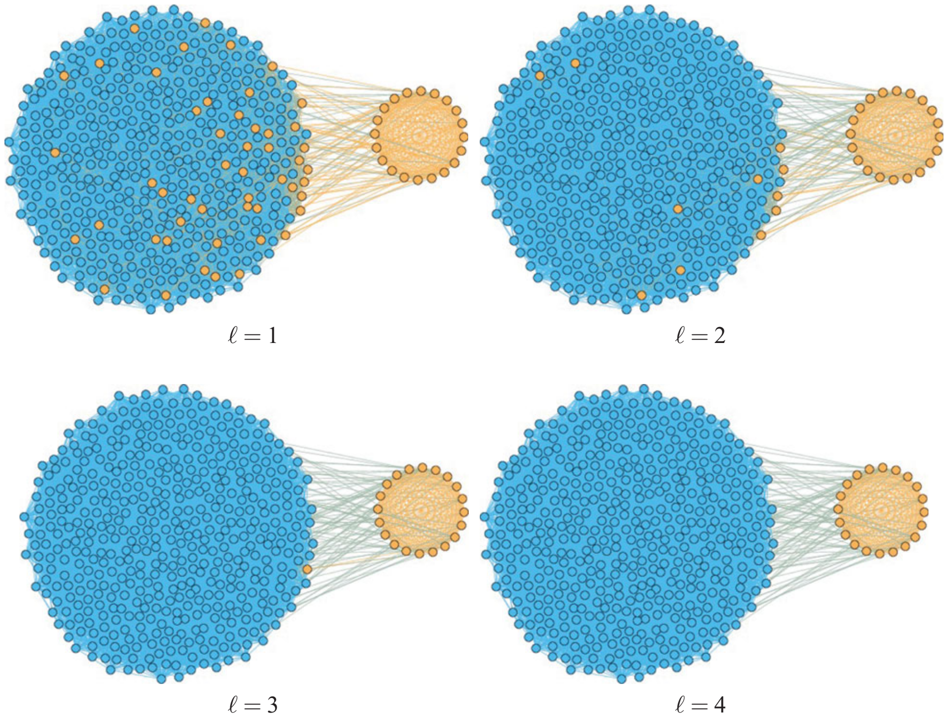

Fig. 1. Random network ($n = 500$) with an embedded K_{20}. The two communities found by edge-modularity, $\ell = 1$, have 50 misplaced nodes (top left). The two communities found by walk-modularity with $\ell = 2$ have 11 misplaced nodes (top right). When $\ell = 3$ there is only 1 misplaced node (bottom left), and when $\ell = 4$ every single node is placed correctly (bottom right). (color online)

We next demonstrate a benchmark test from Lancichinetti et al. (2008). This synthetic test simulates a real-world network by placing each node in a well-defined community following a user-specified average degree, $\bar{k}$, and then randomly rewires nodes between communities according to a mixing parameter μ. The result is a network with several communities with an approximate proportion of $1 - \mu$ edges among each community and μ edges between any one community and the others. One such network, having six communities, is shown in Figure 2, for parameters $n = 500, \mu = 0.15$, and $\bar{k} = 25$. Walk-modularity outperforms edge-modularity in terms of recovering the original six communities. Figure 3 shows the communities found by edge-modularity, and Figure 4 shows the result of walk-modularity maximization with $\ell = 8$.

4.2 Real-world networks

The dolphin association network is a social network of 62 dolphins from Doubtful Sound, New Zealand, with edges representing social relations between individuals, as established by observation over the course of seven years (Lusseau et al., 2003). During the period of observation, the network of dolphins split into two separate communities. In a partition of the dolphin network, a node is considered misplaced if the algorithm places it into a different community than that observed in Lusseau

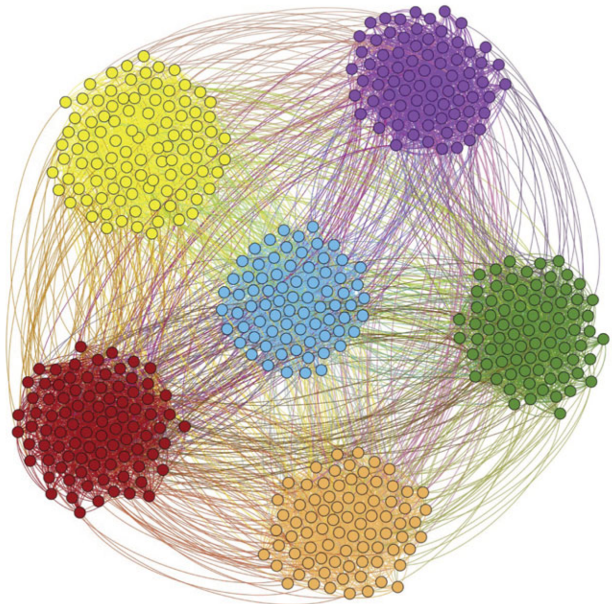

Fig. 2. Six communities as defined by the benchmark test Lancichinetti et al. (2008) with parameters $n = 500, \mu = 0.15, \bar{k} = 25$. (color online)

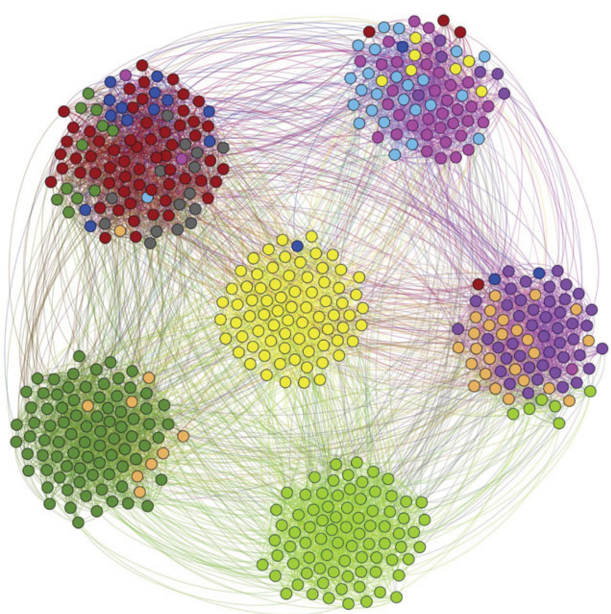

Fig. 3. Communities as detected by edge-modularity, $\ell = 1$, using the leading eigenvector method to maximize Q_1. (color online)

et al. (2003). We should emphasize that here we consider a node as being "misplaced" only relative to the empirical observations concerning the dolphin social network. Partitioning this network with edge-modularity, using only the leading eigenvector method, finds three nodes misplaced, as seen in Figure 5 (top). After increasing the length of walks considered to $\ell = 8$, the diameter of this network, the number

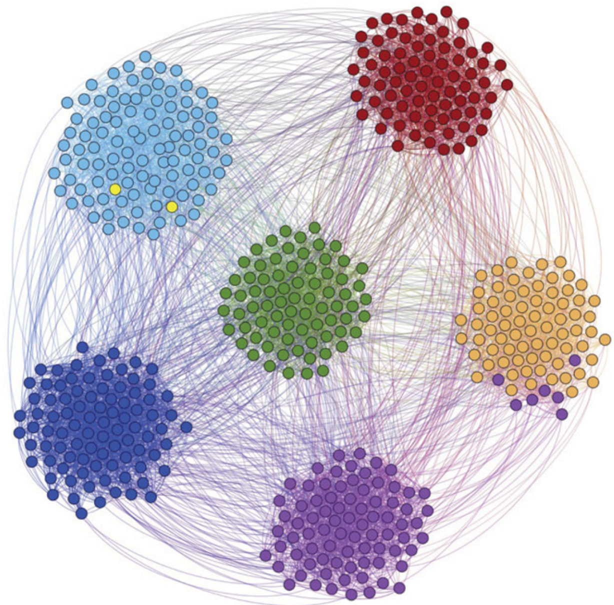

Fig. 4. Communities as detected by walk-modularity, $\ell = 8$, using the leading eigenvector method to maximize Q_8. (color online)

of nodes misplaced drops to two, and a value of $\ell = 10$ produces only a single misplaced node, Figure 5 (bottom).

The Zachary karate club network is a real-world social network consisting of 34 members of a karate club which split into two communities following a disagreement between the club's instructor and administrator (Zachary, 1977). The nodes of this network represent individuals, and the edges represent friendships, as observed by Zachary. As before, a node is considered misplaced if the algorithm places it in a community which differs from the observed real-world partition. In this example, there are no misplaced nodes for walk lengths between $\ell = 1$–7 (Figure 6).

For a given network, the user-controlled parameter ℓ can significantly impact the quality of the partition found by maximizing walk-modularity. In practice, we have found that choosing a value for ℓ somewhat near to the diameter of the network gives excellent results, as demonstrated in Table 1. Although it is generally computationally expensive to find the exact diameter of a network, there are algorithms which can approximate the diameter in $O(mn)$ time (Crescenzi et al., 2012). Moreover, we emphasize that even by using small values of $\ell > 1$, we obtained significant improvements in the observed quality of partitions. We summarize the test case results of this section in Table 1.

Both the dolphin network and the Zachary karate club network have been quite commonly employed as real-world tests for many community detection algorithms. Nevertheless, it is important to keep in mind that it is an empirical observation that these social networks explain the observed separations, but this does not necessarily imply that a method finds more meaningful communities if it reproduces these two splits more accurately.

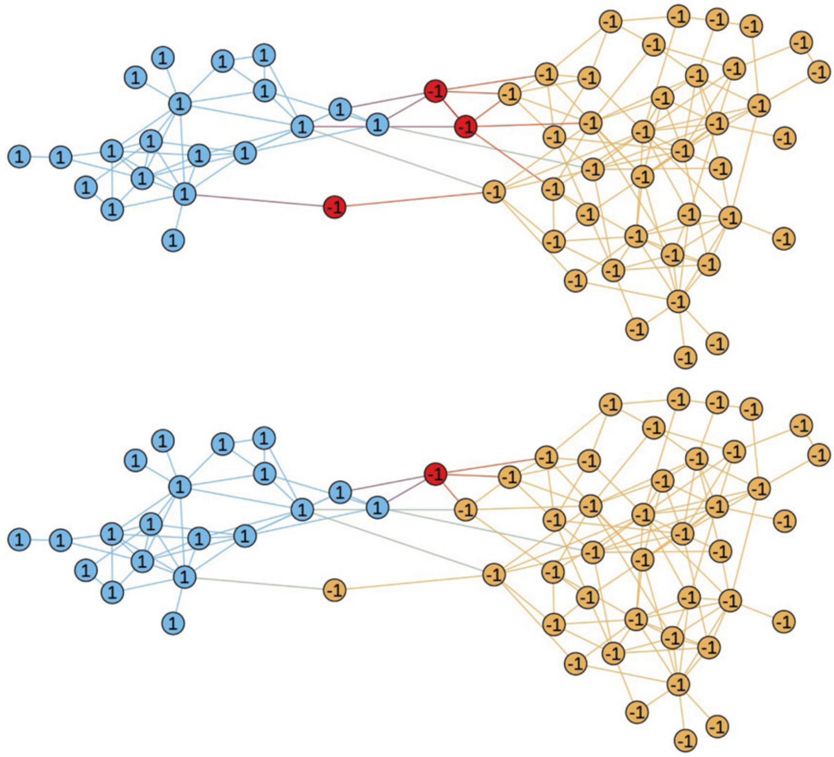

Fig. 5. The dolphin network Lusseau et al. (2003). Nodes are labeled with ±1 according to the empirically observed split of the dolphin network. (Top) Nodes are colored according to the partitioning found by edge-modularity maximization. Nodes differing from empirical observation are colored red. (Bottom) Nodes are colored according to the partitioning found by walk-modularity maximization with $\ell = 10$. There is only a single node differing from empirical observation of the dolphin network split. (color online)

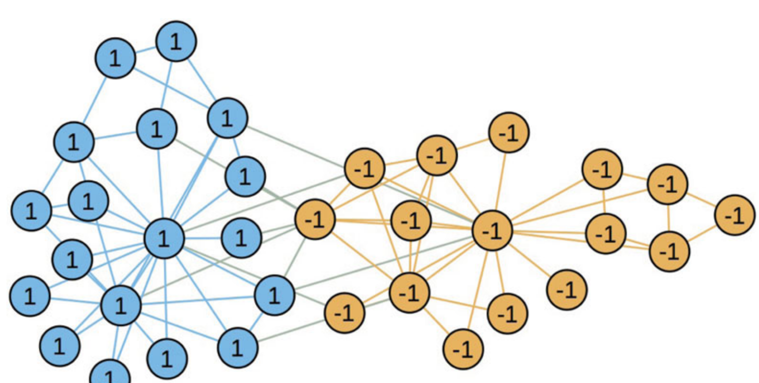

Fig. 6. The Zachary karate club network Zachary (1977). The nodes are labeled ±1 according to the real-world split of the club, and colored according to how the algorithm determined community membership. The partition is the same for both edge-modularity ($\ell = 1$) and walk-modularity ($\ell = 2$–7). (color online)

Table 1. *Summary of the number of nodes placed in incorrect communities as the walk length parameter, ℓ, varies. In each case, as ℓ increases the community detection becomes more accurate.*

Network	Diameter	Walk length ℓ	Nodes misplaced
Embedded K_{20}	3	1	50
		2	11
		3	1
		4	0
Benchmark Test	4	1–3	>100
Lancichinetti et al. (2008)		4	72
$n = 500$		5	92
$\mu = 0.15$		6	40
$\bar{k} = 25$		7	11
		8	9
Dolphins	8	1–7	3
Lusseau et al. (2003)		8, 9	2
		10	1
Karate Club	5	1–7	0
Zachary (1977)			

5 Conclusion

Walk-modularity is a natural generalization of modularity because it considers the difference between the actual and expected number of walks of length ℓ within communities. Mathematically, it is a very simple and elegant generalization in that it only involves taking the ℓ-th powers of both the adjacency matrix and the familiar expected adjacency matrix. As with traditional edge-modularity, walk-modularity can be maximized by finding the leading eigenvector of a matrix, called the walk-modularity matrix. Although we have only explored, in this paper, a single technique to maximize walk-modularity, it is a quantity that is compatible with maximization algorithms other than spectral maximization.

Even with small values of $\ell > 1$, we demonstrated with test cases that maximizing walk-modularity produces partitions with many fewer misplaced nodes than traditional edge-modularity. For a random network with an embedded complete graph, K_{20}, walk-modularity is capable of perfectly identifying the complete subgraph and separating it from the random network. Walk-modularity is also more successful in identifying the six communities in a randomly generated benchmark test where edge-modularity did not perform well. With two standard real-world community detection test cases, namely the dolphin network and the karate club network, walk-modularity performs in a manner comparable to the other most common community detection algorithms, and perhaps a bit better than edge-modularity in our comparison on the dolphin network.

Walk-modularity should suffer the well-known resolution limit of edge-modularity (Fortunato & Barthelemy, 2007), but in this case likely resulting from a comparison of the number of length ℓ walks of the interconnected communities and the total number of length ℓ walks on the network. Therefore, walk-modularity may miss substructures smaller than some scale. Moreover, walk-modularity is likely not exempt from the so-called detectability threshold limitation that may be inherent in all community detection algorithms (Decelle et al., 2011a,b; Nadakuditi & Newman, 2012; Radicchi, 2013, 2014a,b). In these recent works, it has been demonstrated that even though a community may be well defined when its in-degree exceeds its out-degree, it may not be detectable unless this difference exceeds a strictly positive threshold. The detectability threshold and modularity maximization is discussed in Nadakuditi & Newman (2012); Radicchi (2013), and it seems plausible that the approach taken in these works could be extended to address detectability and walk-modularity maximization.

Lastly, we note that edge-modularity has been previously modified to uncover the structure of networks with overlapping communities (Nepusz et al., 2008; Nicosia et al., 2009). It was shown that one straightforward way of modifying modularity to handle overlapping communities is to replace the Kronecker delta in the definition of the modularity function with a fuzzy co-membership quantity that measures the extent to which two nodes belong to the same community. The exact same approach could be used to extend walk-modularity to detect fuzzy overlapping communities.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. DMS-1062128. The authors thank the referees and the editor for useful comments.

References

Agarwal, G., & Kempe, D. (2008). Modularity-maximizing graph communities via mathematical programming. *The European Physical Journal B*, **66**(3), 409–418.

Arenas, A., Fernández, A., Fortunato, S., & Gómez, S. (2008). Motif-based communities in complex networks. *Journal of Physics A: Mathematical and Theoretical*, **41**(22), 224001.

Barber, M. J. (2007). Modularity and community detection in bipartite networks. *Physical Review E*, **76**(6), 066102.

Berry, J. W., Hendrickson, B., LaViolette, R. A., & Phillips, C. A. (2011). Tolerating the community detection resolution limit with edge weighting. *Physical Review E*, **83**(5), 056119.

Blondel, V. D., Guillaume, J., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, **2008**(10), P10008.

Brandes, U., Delling, D., Gaertler, M., Görke, R., Hoefer, M., Nikoloski, Z., & Wagner, D. (2008). On modularity clustering. *IEEE Transactions on Knowledge and Data Engineering*, **20**(2), 172–188.

Clauset, A., Newman, M. E. J., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E*, **70**(6), 066111.

Crescenzi, P., Grossi, R., Habib, M., Lanzi, L., & Marino, A. (2012). On computing the diameter of real-world undirected graphs. *Theoretical Computer Science*, **514**, 84–95.

Decelle, A., Krzakala, F., Moore, C., & Zdeborová, L. (2011a). Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, **84**(6), 066106.

Decelle, A., Krzakala, F., Moore, C., & Zdeborová, L. (2011b). Inference and phase transitions in the detection of modules in sparse networks. *Physical Review Letters*, **107**(6), 065701.

Duch, J., & Arenas, A. (2005). Community detection in complex networks using extremal optimization. *Physical Review E*, **72**(2), 027104.

Estrada, E. (2011a). Community detection based on network communicability. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, **21**(1), 016103.

Estrada, E. (2011b). *The structure of complex networks: Theory and applications*. Oxford, England: Oxford University Press.

Estrada, E., & Hatano, N. (2008). Communicability in complex networks. *Physical Review E*, **77**(3), 036111.

Estrada, E., & Hatano, N. (2009). Communicability graph and community structures in complex networks. *Applied Mathematics and Computation*, **214**(2), 500–511.

Estrada, E., Hatano, N., & Benzi, M. (2012). The physics of communicability in complex networks. *Physics Reports*, **514**(3), 89–119.

Fortunato, S. (2009). Community detection in graphs. *Physics Reports*, **486**(3), 75–174.

Fortunato, S., & Barthelemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, **104**(1), 36–41.

Good, B. H., de Montjoye, Y., & Clauset, A. (2010). Performance of modularity maximization in practical contexts. *Physical Review E*, **81**(4), 046106.

Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical Review E*, **78**(4), 046110.

Lusseau, D., Schneider, K., Boisseau, O. J., Haase, P., Slooten, E., & Dawson, S. M. (2003). The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, **54**(4), 396–405.

Nadakuditi, R. R., & Newman, M. E. J. (2012). Graph spectra and the detectability of community structure in networks. *Physical Review Letters*, **108**(18), 188701.

Nepusz, T., Petróczi, A., Négyessy, L., & Bazsó, F. (2008). Fuzzy communities and the concept of bridgeness in complex networks. *Physical Review E*, **77**(1), 016107.

Newman, M. E. J. (2004). Fast algorithm for detecting community structure in networks. *Physical Review E*, **69**(6), 066133.

Newman, M. E. J. (2006a). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, **74**(3), 036104.

Newman, M. E. J. (2006b). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, **103**(23), 8577–8582.

Newman, M. E. J. (2011). Communities, modules and large-scale structure in networks. *Nature Physics*, **8**(1), 25–31.

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, **69**(2), 026113.

Nicosia, V., Mangioni, G., Carchiolo, V., & Malgeri, M. (2009). Extending the definition of modularity to directed graphs with overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment*, **2009**(03), P03024.

Porter, M. A., Onnela, J., & Mucha, P. J. (2009). Communities in networks. *Notices of the AMS*, **56**(9), 1082–1097.

Radicchi, F. (2013). Detectability of communities in heterogeneous networks. *Physical Review E*, **88**(1), 010801.

Radicchi, F. (2014a). Driving interconnected networks to supercriticality. *Physical Review X*, **4**(2), 021014.

Radicchi, F. (2014b). A paradox in community detection. *EPL (Europhysics Letters)*, **106**(3), 38001.

Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E*, **74**(1), 016110.

Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, **33**(4), 452–473.