Subspace Based Network Community Detection Using Sparse Linear Coding

Arif Mahmood and Michael Small

Abstract—Information mining from networks by identifying communities is an important problem across a number of research fields including social science, biology, physics, and medicine. Most existing community detection algorithms are graph theoretic and lack the ability to detect accurate community boundaries if the ratio of intra-community to inter-community links is low. Also, the algorithms based on modularity maximization may fail to resolve communities smaller than a specific size if the community size varies significantly. In this paper we present a fundamentally different community detection algorithm based on the fact that each network community spans a different subspace in the geodesic space. Therefore, each node can only be efficiently represented as a linear combination of nodes spanning the same subspace. To make the process of community detection more robust, we use sparse linear coding with ℓ_1 norm constraint. In order to find a community label for each node, sparse spectral clustering algorithm is used. The proposed community detection technique is compared with more than ten state of the art methods on two benchmark networks (with known clusters) using normalized mutual information criterion. Our proposed algorithm outperformed existing algorithms with a significant margin on both benchmark networks. The proposed algorithm has also shown excellent performance on three real-world networks.

Index Terms—Complex Networks, Community Detection, Sparse Linear Coding, Sparse Subspace Clustering, Subspace Community Detection

1 Introduction

Many real world systems emerging from computational areas in social science, biology, physics, and medicine naturally map to network data structures [3], [36]. To analyse the structure of the original systems, often the corresponding network structure is studied using groups of nodes having more intra-group and less inter-group edges. Such groups exist in most real world networks and influence the behaviour of the underlying system. The community detection is an important problem and it has the potential to solve many real world problems. For example, information propagation across the globe is influenced by the group structure in the online social communities [31]. Spread of disease across continents depends on the network of migratory birds or humans. Failure propagation in an electrical supply system can be predicted by the grid community structure [39]. Efficient layout of an electric circuit is computed by finding the community structure [6], [35]. The Internet, the web of hyper links, the connections between neurons and the protein-protein interaction networks collectively demonstrate the importance of community detection [16], [18].

Most of the existing community detection algorithms [8], [13], [20], [37], [40], [42], [44], [50] lack the ability to detect accurate community boundaries if the difference between the internal and the external node degree does not exceed a detectability threshold [41]. Most of these methods use modularity [37], [19] as the quality index of a community scheme. It has been observed that in networks with communities of significantly different sizes, modularity maximization algorithms may fail to accurately resolve communities smaller than a specific size. This behaviour has been reported even in cases with well defined community structure [27].

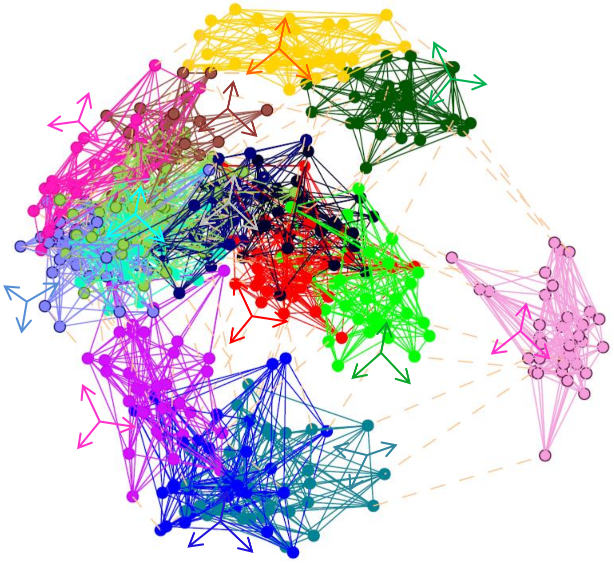

Fig. 1. We consider a community as a set of nodes spanning the same subspace in the geodesic space. Each community spans a different subspace due to different degree distribution and internal structure from the remainder of the communities.

The community detection algorithm proposed in this paper is fundamentally different from existing methods because it does not directly operate on the adjacency matrix and also it is not based on modularity maximization. We propose to represent each network node by a vector of geodesic distances with respect to all other nodes in the network. In case of unweighted networks, geodesic distance is the number of links between the two nodes along the shortest path. For the case of weighted graphs geodesic distance is the sum of the link weights along the shortest path. Such a mapping is one-to-one because each node is mapped to

- *A. Mahmood and M. Small are with the Department of Mathematics and Statistics, The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009. E-mails: {arif.mahmood, michael.small}@ uwa.edu.au*

Manuscript received December 2014.

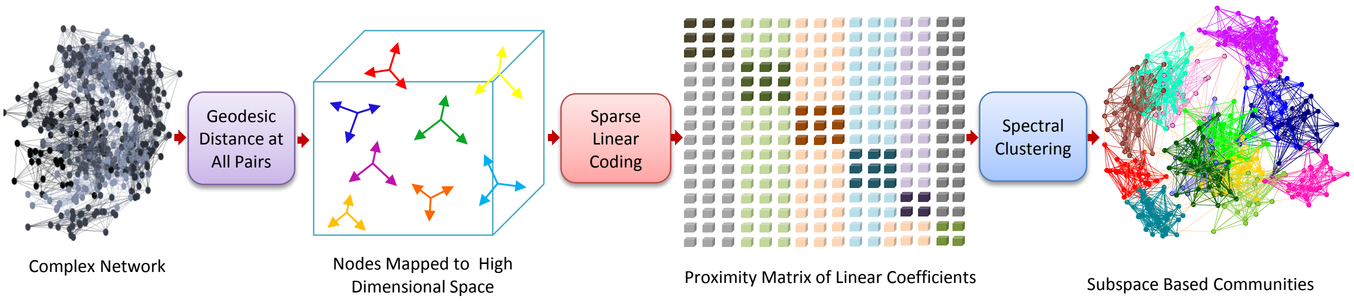
Geodesic Distance at All Pairs
Sparse Linear Coding
Spectral Clustering
Complex Network
Nodes Mapped to High Dimensional Space
Proximity Matrix of Linear Coefficients
Subspace Based Communities

Fig. 2. The proposed subspace based community detection algorithm has three main steps. For a given network geodesic distances are computed for all pairs and each node is represented by the corresponding vector of distances. Then a sparse linear coding is used to decompose each node as a linear combination of all other nodes. Finally a spectral clustering algorithm is used to find communities in the network.

a unique point in a high dimensional geometric space. Since communities are defined as group of nodes having more intra-group and less inter-group links, therefore the expected value of geodesic distance between two nodes in the same community will be smaller than that of the two nodes in two different communities. As a result, in the mapped geometric space, each community will span a different subspace. Although the apparent dimensionality of a node is the same as the number of nodes in the network, the actual dimensionality is significantly smaller depending on the size of the community that node occupied.

As an example, consider an unweighted and undirected network having two disconnected 3-cliques, with nodes labeled as $\{1, 2, 3\}$ and $\{4, 5, 6\}$. Geodesic distance is the minimum number of edges or the shortest path distance between two nodes. In this network, each node is represented by a column in a matrix of Geodesic distances P:

$$P = \begin{bmatrix} 0 & 1 & 1 & \infty & \infty & \infty \\ 1 & 0 & 1 & \infty & \infty & \infty \\ 1 & 1 & 0 & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & 1 & 1 \\ \infty & \infty & \infty & 1 & 0 & 1 \\ \infty & \infty & \infty & 1 & 1 & 0 \end{bmatrix}.$$

An inverse exponential mapping of the type $s = \exp(-p^2/\sigma^2)$ will map P to a matrix of similarities S:

$$S = \begin{bmatrix} 1 & 0.85 & 0.85 & 0 & 0 & 0 \\ 0.85 & 1 & 0.85 & 0 & 0 & 0 \\ 0.85 & 0.85 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0.85 & 0.85 \\ 0 & 0 & 0 & 0.85 & 1 & 0.85 \\ 0 & 0 & 0 & 0.85 & 0.85 & 1 \end{bmatrix}.$$

Nodes in each community span a different subspace $\mathcal{R}^3$ while the overall dimensionality is $\mathcal{R}^6$. A similar network connected by a single edge has been analyzed in Fig. 3.

A vector with apparent dimensionality of n and actual dimensionality of n' such that $n' \leq n$ can only be represented as a linear combination of n' independent vectors spanning the same subspace. In such a decomposition the linear coefficients corresponding to the vectors spanning different subspaces will become zero. Due to the complex network structure, a simple linear decomposition estimated by least squared error solution may result in non-zero linear coefficients of small magnitude for nodes corresponding to different communities. In order to solve this problem we use sparse linear coding in which linear coefficients are estimated such that in addition to minimization of reconstruction error, the number of linear coefficients, or ℓ_o norm, is also constrained. By introducing this constraint we ensure that the linear coefficients corresponding to dimensions spanned by other communities become zero.

In order to obtain subspace based communities, we represent each node as a sparse linear combination of all other nodes in the same network and use the magnitude of the linear coefficients as the similarity values to define a proximity matrix. We then use spectral clustering to partition the graph represented by this proximity matrix into k clusters. The value of k is obtained corresponding to the minimum reduction rate of the clustering error in the Euclidean space.

In dense networks the small world phenomena [53], [52] renders the process of accurate community boundary identification more challenging. In order to mitigate this effect, node representation from the traditional spectral clustering algorithms is fused with the proposed geodesic distance based algorithm. The information captured by the both algorithms being different, complement each other and improve the accuracy of community detection. Both versions of the proposed algorithm are compared with more than ten existing community detection methods on two benchmark networks with varying community sizes. The proposed algorithms have shown excellent performance in all experiments.

2 Related Work

Most of the existing community detection algorithms directly operate on the adjacency matrix which encodes local network structure at each node. Although it appears simple and natural, it constraints the set of algorithms which may potentially be applied for community detection. In particular, most machine learning and data mining algorithms cannot be directly applied to this network representation. Recently, some researchers have proposed the use of clustering algorithms such as K-means or Voronoi diagrams [12], DBSCAN [21], DENCLUDE [25] and sub graph detection [7] for community detection. However, in most of these techniques a network is not globally mapped to a space, rather the mapping is local or discrete, considering only two nodes at a time. In contrast, in the current work we propose to map a network to a continuous high dimensional space. We define a community as a set of nodes spanning the same subspace within the high dimensional space.

Most of the classical community detection algorithms are graph theoretic. For example, S. Dongen [50] proposed Markov cluster algorithm [50] based on the idea of current flow in the graph. If natural groups are present in a network, then the current across group borders will be small thus revealing group structure in the graph. Radicchi et al. [42] proposed a divisive algorithm based on edge-clustering coefficient, the ratio of number of triangles an edge belongs to the potential number of such triangles. Edges connecting different groups have low clustering coefficient and are removed first. Girvan and Newman [20], [37] proposed a community detection algorithm (GN) based on the concept of edge betweenness which is the number of shortest paths that run along an edge. The edge with highest betweenness is removed and shortest paths are recomputed each time. Clauset et al.[8] proposed a fast greedy modularity optimization algorithm which is very efficient on sparse graphs with hierarchical structure. Blondel et al.[4] proposed a modularity optimization based fast heuristic algorithms for community structure extraction in large networks. Recently Deritei *et al* [12] represented distance between two nodes based on the edge-clustering coefficient and used Voronoi diagrams for community detection. Palla et al. [40] first located all cliques of the network and then found communities by carrying out a standard component analysis of the clique−clique overlap. Rosvall and Bergstrom have proposed an information-theoretic framework for resolving community structure in complex networks [45] known as Infomap. A network is divided into small modules such that Minimum Description Length (MDL) is minimized. Wang *et al.* [51] have proposed dynamic community detection algorithm. Tang et al. [48] have suggested an approach for evolving group detection in dynamic networks. Liu et al. [32] has proposed a community detection algorithm in directional networks.

Our proposed algorithm is fundamentally different from all of these methods because we define a community as a set of nodes spanning the same subspace. A community is differentiated from another community based on the difference of the subspace spanned by each. The closest existing algorithm is Donetti and Munoz (DM) algorithm [13]. They represented each node by a column of adjacency matrix. First a few eigenvalues and eigenvectors of the network Laplacian matrix were computed and then based on complete link clustering algorithm, network communities were found. In contrast, we represent each node by a vector of geodesic distances and then we use sparse linear coding to compute a proximity matrix based on the subspace spanned by each node. The matrix of linear coefficients is used as a proximity matrix for the spectral clustering algorithm [15]. To the best of our knowledge, no such network community detection algorithm has been proposed before. Our work also bridges the gap between subspace based clustering techniques and complex networks. Note that the proposed algorithm is equally applicable to both the weighted and un-weighted networks as well as directed and undirected networks.

3 Sparse Linear Decomposition of Nodes

Consider a network G with n nodes and m links represented by an adjacency matrix $A \in \mathbb{R}^{n\times n}$ such that if there is a link between the two nodes $\{v_i, v_j\}$ then $A(i,j) = 1$, otherwise $A(i,j) = 0$. Each column of adjacency matrix is a vector in $\mathbb{R}^n$ and records the nodes directly incident on v_i, therefore it only captures the local structure at v_i. It does not record information of the nodes which are further than one link from v_i.

We propose to map $\mathcal{V} = \{v_1, ..., v_i, ...v_n\} \mapsto \mathbb{R}^n$ such that each dimension represents some type of distance of v_i from a particular node. Since geodesic distance satisfies the three properties of a metric including non-negativity, symmetry and triangular inequality, it can be considered as an appropriate choice to define such a representation. Another advantage is the availability of many fast geodesic distance computation algorithms [24] that can be utilized to efficiently compute this representation.

Each community has a relatively high number of intra-community edges and low number of inter-community edges therefore the nodes within the same community are expected to have smaller geodesic distances as compared to the nodes in the different communities. Also, due to different node and edge distribution in each community, the geodesic distance vectors $\mathbf{p}_i$ corresponding to a particular community span a different subspace. Hence our proposed sub-space based approach will yield a new insight into the network community structure.

Let the vector $\mathbf{p}_i$ be the set of geodesic distances of v_i from all $v_j \in G$. The set of all such vectors is a matrix $P \in \mathbb{R}^{n\times n}$ such that $P = [\mathbf{p}_1 \;\; \mathbf{p}_2 \;\; \cdots \;\; \mathbf{p}_n]$. Note that $P(i,j)$ is the distance between v_i and v_j and all diagonal entries of P are zero, $P(i,i) = 0$, that is the network does not have any self loops. We transform the geodesic distance vectors $\mathbf{p}_i$ to similarity score vectors by using Gaussian kernel function:

$$S = \exp\Big(-\frac{P \odot P}{2\sigma_s^2}\Big), \qquad (1)$$

where σ_s controls the rate of decay and $\odot$ is point-wise multiplication operator. If a node v_i is not reachable from a node v_j then $P(i,j) = \infty$ and $S(i,j) = 0$.

A column vector $\mathbf{s}_i \in S$ can only be represented as a linear combination of other vectors $\mathbf{s}_j \in S$ spanning the same subspace. Therefore, if $\mathbf{s}_i$ is decomposed as a linear combination of the rest of the vectors in S, defined as set difference: $\hat{S} = S \setminus \mathbf{s}_i$, the linear coefficients $\boldsymbol{\alpha}_i$ may be found by using $\mathbf{s}_i = \hat{S}\boldsymbol{\alpha}_i$. The least squares solution is given by

$$\boldsymbol{\alpha}_i = (\hat{S}^\top \hat{S})^{-1} \hat{S}^\top \mathbf{s}_i, \qquad (2)$$

where the vector of linear coefficients $\boldsymbol{\alpha}_i$ will have large magnitudes corresponding to nodes in the same community and small magnitudes for nodes in different communities.

As a simple example, consider a network with k isolated communities with no inter-community links. In this case, the geodesic distance between two nodes in two different communities will be $P(i,j) = \infty$, therefore $S(i,j) = 0$. The subspace spanned by each community will be independent, therefore a node in a particular community will only have nonzero coefficients in $\boldsymbol{\alpha}_i$ for nodes within the same community. Coefficients in $\boldsymbol{\alpha}_i$ corresponding to the nodes in the other communities will be zero.

In real world networks, community detection is significantly more challenging than this simple example. Mostly there are a significant number of inter-community links compared to the intra-community links. Therefore the subspaces spanned by different communities are neither independent nor disjoint; rather may have significant overlap depending on the structure of the network. In such cases, the least squares solution will yield non zero coefficients in $\boldsymbol{\alpha}_i$ corresponding to the nodes in the other communities making community detection more challenging. To illustrate this fact a simple network having two 3-cliques connected with a single edge (Fig. 3a) is considered. The matrix of linear coefficients computed by least squares solution is shown in Fig. 3c. This matrix has a clear block structure however nodes in each community have nonzero linear coefficients corresponding to the nodes from the other community. Fig. 3b shows the plot of linear coefficients obtained by sparse linear coding as discussed in the following

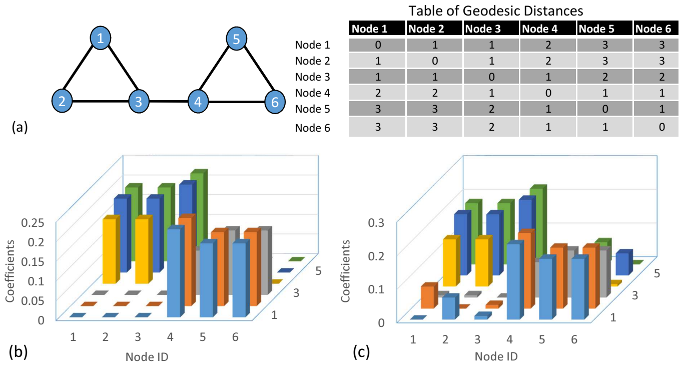

Table of Geodesic Distances

	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6
Node 1	0	1	1	2	3	3
Node 2	1	0	1	2	3	3
Node 3	1	1	0	1	2	2
Node 4	2	2	1	0	1	1
Node 5	3	3	2	1	0	1
Node 6	3	3	2	1	1	0

Fig. 3. (a) A network with two 3-cliques connected by a single edge along with the table of geodesic distances. (b) The matrix of $\boldsymbol{\alpha}_i$ computed by sparse linear coding (4). (c) The matrix of linear coefficients $\boldsymbol{\alpha}_i$ computed by least error squares solution (2). The block structure in (b) is significantly enhanced compared to the block structure in (c).

paragraph. In this case, most of the noisy coefficients have been suppressed to zero.

In order to suppress response from less relevant subspaces and to enable community detection in the presence of significant inter community links, the linear decomposition must be performed with sparsity constraint

$$\min_{\boldsymbol{\alpha}_i} ||\mathbf{s}_i - \hat{\mathcal{S}}\boldsymbol{\alpha}_i||_2^2 \text{ s. t. } ||\boldsymbol{\alpha}_i||_0 \le \lambda \ , \tag{3}$$

where $|| \cdot ||_0$ is the ℓ_0 norm which is the number of nonzero coefficients in $\boldsymbol{\alpha}_i$. The parameter λ is the number of allowed non zero coefficients. This constraint ensures that only λ best nodes can participate to represent a given node, therefore the value of λ must not exceed the size of the corresponding community. In case of smaller communities, a larger vale of λ may allow selection of some nodes from other communities. If the coefficients corresponding to these nodes are not very small an error may be introduced in the community boundary. We avoid this by constraining the sparsity indirectly and instead of minimizing ℓ_0 norm, we minimize ℓ_1 norm:

$$\boldsymbol{\alpha}_i^* := \arg\min_{\boldsymbol{\alpha}_i} \left(||\mathbf{s}_i - \hat{\mathcal{S}}\boldsymbol{\alpha}_i||_2^2 + \lambda ||\boldsymbol{\alpha}_i||_1 \right) \ , \tag{4}$$

where the parameter λ assigns a relative importance to the sparsity constraint as compared to the reconstruction error magnitude. In this formulation, for each $\mathbf{s}_i$, the value of λ is automatically computed from the $\hat{\mathcal{S}}$. Sparse linear decomposition with ℓ_1 norm regularization given by (4) is an unconstrained convex optimization problem also known as Least Absolute Selection and Shrinkage Operator (LASSO) [49]. A fast solution in Least Angle Regression (LARS) framework [14] has the same asymptotic complexity as the simple least squares regression.

4 Sparse Subspace Community Detection

In our formulation, a community $\mathcal{C}_k$ containing a node v_i is a set of nodes containing the support of v_i as its subset. The support is the set of nodes correspond to the coefficients of larger magnitude in $\boldsymbol{\alpha}_i$ compared to a threshold. Two nodes are considered to belong to the same community if their supports have an overlap larger than a given threshold. The overall community is then the union of the supports of both nodes. Therefore all nodes having overlapping supports will correspond to the same community which is then the union of all these supports. In order to find all sets of nodes with overlapping supports, clustering of the sparse linear coefficient vectors $\boldsymbol{\alpha}_i$ is required.

Due to randomness in the optimization process in (4), the coefficient $\boldsymbol{\alpha}_i(j)$ may be different from the coefficient $\boldsymbol{\alpha}_j(i)$. It is also partially because of the fact that the set of vectors used to represent v_i is slightly different from the set used to represent v_j. We make the relationship normalized and symmetric by taking the average of the both coefficients each normalized by the maximum value of its own set:

$$\mathcal{F}(i,j) = \mathcal{F}(j,i) = \frac{1}{2}\left(\left| \frac{\boldsymbol{\alpha}_i(j)}{\max(\boldsymbol{\alpha}_i)} \right|_1 + \left| \frac{\boldsymbol{\alpha}_j(i)}{\max(\boldsymbol{\alpha}_j)} \right|_1 \right), \tag{5}$$

where $\mathcal{F}$ is the resulting matrix of symmetric linear coefficients. Instead of directly applying an Euclidean space clustering algorithm on $\mathcal{F}$, we apply Spectral Clustering (SC) approach.

4.1 Spectral Clustering

SC acts as a kernel and maps data from nonlinear manifolds to the Euclidean space where clustering methods such as K-means can then be efficiently used to find linear groups [15], [33]. For this purpose, a degree matrix $\mathcal{D}$ is computed

$$\mathcal{D}(i,j) = \begin{cases} \sum_{\hat{i}=1}^{n} \mathcal{F}(\hat{i},j) & \text{if } i = j \\ 0 & \text{if } i \ne j \end{cases} , \tag{6}$$

Using $\mathcal{F}$ and $\mathcal{D}$, a symmetric Laplacian matrix $\mathcal{L}_s$ is computed

$$\mathcal{L}_s = I - \mathcal{D}^{-1/2}\mathcal{F}\mathcal{D}^{-1/2}. \tag{7}$$

The eigenvectors of $\mathcal{L}_s$ embed a graph vertices into the Euclidean space where linear approaches can be used for clustering. The second least significant eigenvector of $\mathcal{L}_s$ is known as the *Fiedler vector* and divides the network into two partitions based on the NCut criterion [47]. Thus *Fiedler vector* is useful to discover hierarchical structure of the network. On the resulting two partitions of the network, again same process is repeated to divide each part into two new partitions. Alternatively one may select k least significant eigenvectors of $\mathcal{L}_s$ and directly compute k clusters. Let E_s be the matrix of k least significant eigenvectors of $\mathcal{L}_s$. Rows of E_s are normalized to unit magnitude and clustered using Euclidean distance based linear clustering algorithm.

4.2 Information Fusion

For the case of well structured sparse networks the sub-spaces spanned by different communities remain more independent while for the case of poorly structured complex networks, these subspaces become more overlapped causing the community detection a challenging task. It is due to the small world phenomenon in complex networks [53], [52] the average difference between the geodesic distances between nodes in the same community and nodes in different communities reduces particularly as the number of across community edges increases. Moreover, while performing the sparse linear coding, the sparsity of the linear coefficients corresponding to the communities other than the actual community of the current node reduces. Both of these phenomena result in an increase in the overlap of the sub-spaces spanned by different communities. Therefore to enhance the performance of community detection, information from other sources need to be augmented with the subspace based community detection.

In the proposed subspace based community detection algorithm, the input is the matrix of geodesic distances which encode

```
Algorithm 1 Sparse Subspace Communities with Fusion (SSCF)
Require: $A \in \mathcal{R}^{n\times n}$   {Network Adjacency Matrix}
Ensure: $\ell \in \mathcal{R}^{n\times 1}$ {Community Labels}, $\epsilon_e$ {Clustering Error}
  $P \Leftarrow$Find-Geodesic-Distances($A$)
  $S \Leftarrow \exp(-(P \odot P)/(2\sigma_s^2))$                {Eq. (1)}
  for $i \le n$ do
    $\boldsymbol{s}_i \Leftarrow S(:,i)$                         {$i$-th column of $S$}
    $\widehat{S} = [\boldsymbol{s}_1 \cdots \boldsymbol{s}_{i-1}\ \mathbf{0}\ \boldsymbol{s}_{i+1} \cdots \boldsymbol{s}_n]$
    $\boldsymbol{\alpha}_i^* \Leftarrow$Find-Linear-Sparse-Code($\boldsymbol{s}_i, \widehat{S}$)   {Eq. (4)}
    $\boldsymbol{\alpha}_i^* \Leftarrow \boldsymbol{\alpha}_i^*/\max(\boldsymbol{\alpha}_i^*)$
    $\mathcal{F} \Leftarrow [\mathcal{F}\ \ \boldsymbol{\alpha}_i^*]$
  end for
  $\mathcal{F} \Leftarrow 1/2(\mathcal{F}+\mathcal{F}^\intercal)$                {Eq. (5)}
  $\mathcal{D}_s \Leftarrow$Find-Degree-Matrix($\mathcal{F}$)          {Eq. (6)}
  $\mathcal{D}_a \Leftarrow$Find-Degree-Matrix($A$)          {Eq. (6)}
  $\mathcal{L}_s \Leftarrow I - \mathcal{D}_s^{-1/2}\mathcal{F}\mathcal{D}_s^{-1/2}$          {Eq. (7)}
  $\mathcal{L}_a \Leftarrow I - \mathcal{D}_a^{-1/2}A\mathcal{D}_a^{-1/2}$          {Eq. (7)}
  $E_s \Leftarrow$Find-Eigen-Vectors($\mathcal{L}_s$)
  $E_a \Leftarrow$Find-Eigen-Vectors($\mathcal{L}_a$)
  $E_x \Leftarrow [E_s \quad E_a]$                              {Eq. (8)}
  $(\boldsymbol{\ell}, \epsilon_e) \Leftarrow$Find-Low-Error-Clusters($E_x$)       {Eq. (10)}
```

the minimum number of links between any two nodes in a given network. Since the geodesic distance computation algorithms take the adjacency matrix as input, the information contained in the geodesic distance matrix is essentially another form of the information in the adjacency matrix.

We observe that a combination of both forms of information improves the accuracy of community boundaries. Therefore we also compute a matrix of eigenvectors, E_a, by using the adjacency matrix as the proximity matrix in the spectral clustering algorithm. We append the node representation obtained in E_s with the E_a to get an extended node vector:

$$E_x = [E_a \quad E_s]. \tag{8}$$

Linear clustering algorithm is then applied on the E_x to find community labels in the network. The resulting algorithm is named as Sparse Subspace Communities with Fusion (SSCF) and is outlined as Algorithm 1.

4.3 Quality of a Community Scheme

Traditionally modularity maximization has been considered to be a measure of quality or goodness of a community scheme. Although modularity maximization can also be integrated with the proposed subspace based community detection algorithm, we observe that such approach may fail to resolve communities of smaller sizes in networks with communities of significant size variations.

Average reconstruction error over all communities may also be used as the measure of goodness of a community scheme. Let $\mathcal{U}_k$ be the subspace bases of k-th community $\mathcal{C}_k$. Reconstruction error of all nodes $v_i \in \mathcal{C}_k$ is given by $\sum_{i:v_i\in\mathcal{C}_k} ||\mathbf{s}_i - \mathcal{U}_k\boldsymbol{\alpha}_i||_1$. Summation over all communities yields the overall reconstruction error:

$$\epsilon_s = \frac{1}{K}\sum_{k=1}^{K}\sum_{i:v_i\in\mathcal{C}_k} ||\mathbf{s}_i - \mathcal{U}_k\boldsymbol{\alpha}_i||_1, \tag{9}$$

where K are the total number of communities. We observe that ϵ_s can more efficiently compare two community schemes having the same number of communities. While in case of two schemes with different number of communities, ϵ_s may actually yield less value for the smaller number of communities. It is because of the fact that if two communities are contained in one as in the case of hierarchical schemes, ϵ_s will yield less error for the combined community compared to the separate communities. Also in (9) it is assumed that one node is member of only one community. Therefore the projection of a node in subspaces other than its main community is considered as error. In case of overlapped communities, it would be necessary to exclude the projection of a node on a shared subspace from the error computations.

Alternatively, we can also use average error in the Euclidean space spanned by the columns $\mathbf{e}_i$ of the extended node vectors E_x, as the measure to decide appropriate number of clusters

$$\epsilon_e = \frac{1}{K}\sqrt{\sum_{k=1}^{K}\sum_{i:v_i\in\mathcal{C}_k}(\mathbf{e}_i - \mathbf{m}_k)^2}, \tag{10}$$

where $\mathbf{m}_k$ is the center of cluster $\mathcal{C}_k$. As the value of K is increased, ϵ_e initially decreases at a significant rate, however after a specific value of K, the error reduction rate converges to almost constant rate yielding the efficient number of clusters.

In our implementation of the function `Find-Low-Error-Clusters()` in Algorithm 1, clustering is started from a minimum value (mostly 2 in our experiments) and increased in increments of 1 cluster each time. The slope of the clustering error is computed and correct number of clusters are assumed to be found when the error slope converges to a constant value. To elaborate this, the clustering error slopes are plotted for three real world networks in Figures 8, 10, and 14. These plots are shown for Euclidean error (10) because we observe that this criterion produces more accurate cluster boundaries than (9). We also observe that (10) is able to find small as well as large clusters with quite good accuracy (Figures 9, 15 and 11).

5 Experiments and Results

The two versions of the proposed algorithm including Geodesic Sparse Subspace Communities (GSSC) using E_s (8) as the node representation and Sparse Subspace Communities with Fusion (SSCF) using E_x as the node representation (Algorithm 1) are tested on real and synthetic network datasets. The sparse linear coding (4) is solved using the implementation of [15] which is based on ADMM [5]. Alternating Direction Method of Multipliers (ADMM) solves convex optimization problems by breaking them into smaller and easy to solve problems. Alternatively (4) can also be solved using SPArse Modeling Software (SPAMS) [34] library.

The proposed algorithms are compared with the existing state of the art methods on two standard benchmark networks implemented by Lancichinetti et. al. [29]. Each node in the benchmark network has a ground truth community label. The networks are divided into communities in unsupervised fashion, without using the ground truth labels. The communities found are compared with the ground truth communities using Normalized Mutual Information (NMI) proposed by Danon et. al. [10] and also used by Lancichinetti et. al. [28]. In each of the benchmark network, three different mixing parameter values are used corresponding to the gradually increasing ratio of the out-degree to the in-degree. To normalize the effect of randomness, in each setting, 100 realizations of the benchmark are used and average NMI is reported.

The accuracy performance of the proposed algorithms is compared with ten existing algorithms including fast modularity

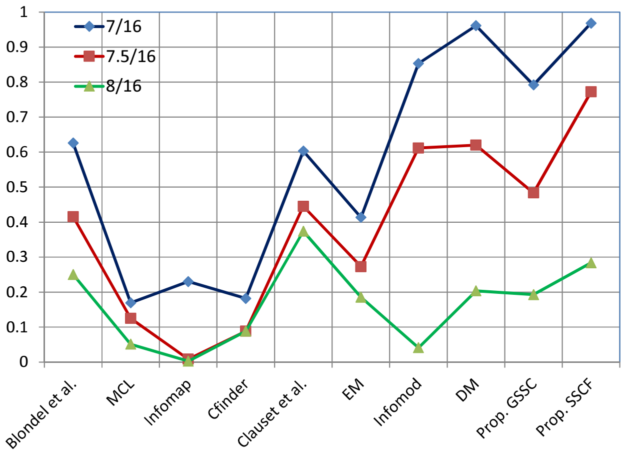

Fig. 4. GN Benchmark Network: Comparison of average Normalized Mutual Information (NMI) over 100 realizations of network with each value of $\mu = \{7/16, 7.5/16, 8/16\}$.

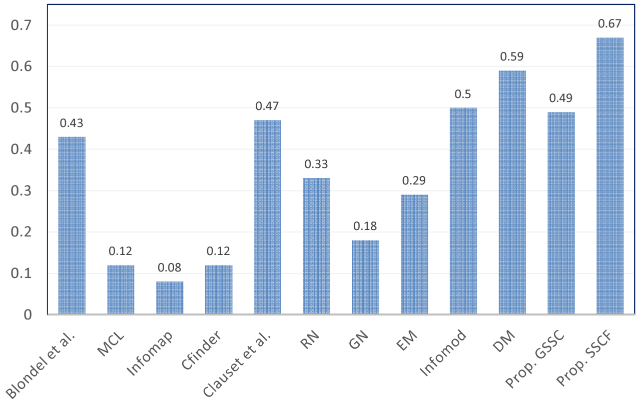

Fig. 5. GN Benchmark Network: Comparison of overall average Normalized Mutual Information (NMI) over 300 realizations of the network. Average accuracy improvement of the proposed SSCF algorithm is 8.00% over the existing best performing algorithm of Donetti and Munoz (DM).

optimization by Blondel et al. [4], Markov Cluster algorithm (MCL) [50], Infomap [45], Cfinder [40], fast greedy modularity optimization by Clauset et al. [8], Radicchi et al. [42], algorithm of Girvan and Newman (GN) [20], [37], spectral algorithm by Donetti and Munoz (DM) [13], Expectation-Maximization (EM) algorithm by Newman and Leicht (EM) [38] and Potts model approach by Ronhovde and Nussinov (RN) [44]. For most of the algorithms results are reported from the original authors or from the comparative studies performed by [10], [26]. For the EM algorithm, results are reported for random initial community boundaries. DM algorithm is an improved version of spectral clustering algorithm using laplacian of the adjacency matrix and E_a (8) as the node representation. We have implemented DM algorithm using complete link hierarchical clustering with angular distance which yields better performance than the Euclidean distance. The proposed algorithms are also compared with the existing algorithms on three real-world networks by using both NMI and the reconstruction error as the quality of the resulting community schemes. The proposed algorithms have exhibited better performance in all cases.

5.1 GN Benchmark

The Girvan-Newman (GN) benchmark [20] has 128 nodes and 4 implanted communities each of 32 nodes. Each node has probability p_{in} of being connected to the nodes of the same cluster and p_{out} of being connected to the nodes of different clusters. Total degree of each node is fixed to 16. A mixing parameter μ is defined as the ratio of the external degree of a node to the total degree. For example, $\mu = 7/16$ means for each node out of 16 links, 7 links are to the outside world. For small values of μ the structure is well defined, while for $\mu \geq 0.50$, $p_{out} \geq p_{in}$, the graph becomes random with subtle structure.

Experiments are repeated by varying $\mu = \{7/16, 7.5/16, 8/16\}$. This range of μ is significant because for $\mu \leq 6/16$, most algorithms have 100% accuracy due to well defined communities; while for $\mu \geq 9/16$, community structure is subtle and most algorithm cannot find any meaningful communities. For each value of μ, average NMI over 100 network realizations is shown in Fig. 4. For values $\mu < 7/16$ the proposed algorithms obtained NMI ≈ 1.00. For $\mu = \{7/16, 7.5/16\}$ the Geodesic Sparse Subspace Communities (GSSC) algorithm has exhibited lower performance than DM and Infomod (Fig. 4). It is because of the fact that the GN network has only 128 nodes each connected to 16 other nodes. Due to the small world phenomenon, GSSC has shown lower performance than DM. The STD of GSSC algorithm is $\{11.37\%, 18\%, 8.75\%\}$ and SSCF algorithm $\{3.68\%, 15.70\%, 14.77\%\}$ over the three versions of the GN network.

The SSCF algorithm has exhibited better performance than all algorithms including DM. It is because of the fact that the information captured by E_s and E_a node representations in (8) compliment each other. Which also shows that the proposed sparse subspace community detection using geodesic vectors is inherently different from the traditional spectral clustering approach and can improve the accuracy of community detection in complex networks. As the mixing parameter is further increased, $\mu = 8/16$, performance of all algorithms significantly decreased (Fig. 4). Infomod suffered more, while both DM and GSSC performed similar. The proposed SSCF exhibited better performance than both DM and GSSC algorithms. Average performance comparison of all algorithms is shown in Fig. 5. On the average, the proposed SSCF algorithm was able to achieve NMI of 0.67 which is better than all other algorithms.

5.2 LFR Benchmark

The LFR network has power law degree distribution and variable sized communities. The number of nodes in the network is 1000, the average degree is 20 and the maximum degree is 50. Minimum planted community size is 30 and maximum is 100. The mixing parameter is varied as $\mu = \{0.60, 0.65, 0.70\}$. For $\mu \leq 0.55$ most algorithms are able to obtain 100% accuracy while for $\mu \geq 0.75$ community structure is not well defined therefore no algorithm can find any meaningful communities. Due to variable degree, communities of different sizes and increased mixing parameter, the performance of most algorithms has remained lower than the GN benchmark network. An NMI comparison for different algorithms is shown in Fig. 6. The value of σ_s in (1) is fixed to 8.

The performance of the both versions of the proposed algorithm, GSSC and SSCF is on the average better than all other algorithms on LFR network (Fig. 7). It may be due to the larger number of nodes, 1000 compared to 128 in GN network, the small world phenomenon has less effect. Therefore the geodesic

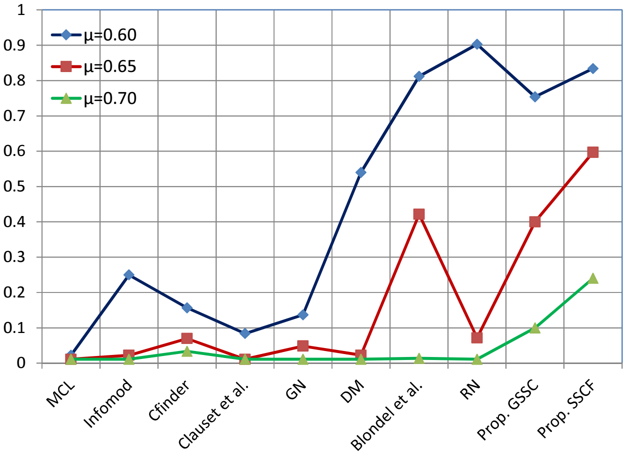

Fig. 6. LFR Benchmark Network: Normalized Mutual Information (NMI) obtained by different algorithms averaged over 100 realizations for each value of the Mixing Parameter $\mu = \{0.60, 0.65, 0.70\}$.

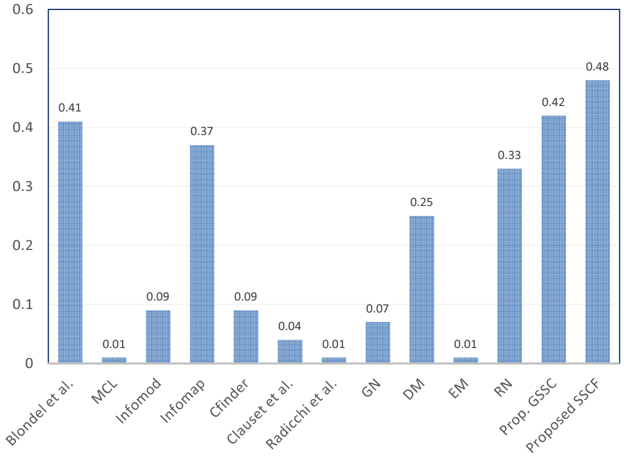

Fig. 7. LFR benchmark network: Average Normalized Mutual Information (NMI) over 300 realizations of the network. Average accuracy improvement of the proposed SSCF algorithm is 7% over the existing best performing algorithm of Blondel et al.

vectors are more discriminating resulting in better performance of GSSC algorithm. The performance of DM [13] algorithm using E_a as node representation in (8) has significantly deteriorated due to more challenges in the LFR benchmark. Other algorithms including Infomap and RN performed better for $\mu = 0.60$ while for $\mu = 0.65$ only the algorithm of Blondal et al. has shown comparatively good performance. For $\mu = 0.70$ all existing algorithms have shown almost zero performance (Fig. 6). This may be because the modularity based methods perform poor when the community size reduces and the network size increases [17]. Also for zero or negative detectability thresholds, the performance of these methods deteriorates. Also a comparative improvement in performance of the proposed GSSC and SSCF algorithms for $\mu = \{0.65, 0.70\}$ demonstrates the ability of this approach to accurately detect communities in more challenging situations. The STD of GSSC algorithm is {5.02%, 8.43%, 3.78%} and for SSCF algorithm {4.45, 8.82, 4.30} over the three versions of the LFR network.

The proposed GSSC algorithm with E_s node representation in (8) has obtained average improvement of 17.0 % over the DM algorithm which is an improved version of the traditional spectral clustering algorithm. The GSSC has also obtained an average improvement of 1.00% while the proposed SSCF algorithm has obtained an average improvement of 7% over the best performing algorithm of Blondal et al. Improvement in performance due to information fusion demonstrates that the node representations E_a and E_s complement each other. Therefore the proposed geodesic distance based spectral clustering algorithm captures different type of information compared to the traditional spectral clustering algorithm. The STD of GSSC algorithm is {5.02%, 8.43%, 3.78%} and for SSCF algorithm {4.45, 8.82, 4.30} over the three versions of the LFR network.

TABLE 1
Experiments on the real-world networks, n, m the number of nodes and edges, σ_s as in (1), R^{ext} is dimensions of the E_{ext}, k_{gt} number of ground truth communities, k_{sc} number of found communities, ϵ_e Euclidean space clustering error as percentage of the error over 2 communities.

Network	n, m	σ_s	R^{ext}	k_{gt}	k_{sc}	ϵ_e
Karat [55]	34, 78	1	6×34	2	7	1.66%
Football [20]	115, 631	10	24× 115	12	12	0.83%
Polblog [1]	1220, 16717	5	14× 1220	2	9	1.44%

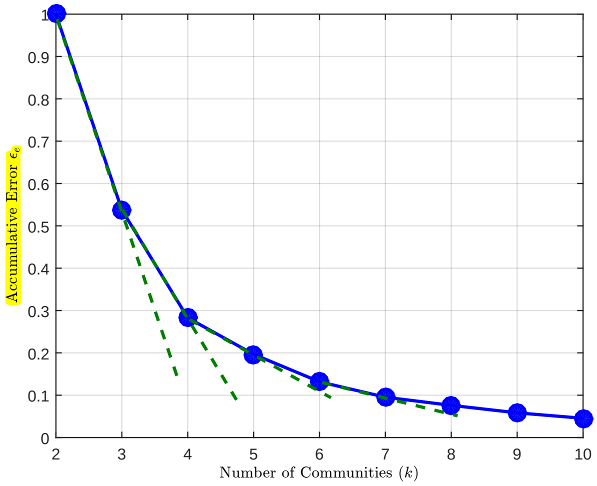

Fig. 8. Zachary Karate Club Network: variation of the normalized sum of squared Euclidean error (SSE or ϵ_e in (10)) with the number of communities (k). Error ϵ_e for the case of two communities is considered to be 1.00 and the remaining values are scaled accordingly. As the communities are increased from 2 to 3, ϵ_e reduced by 46.22%. However this reduction slowed down with further increase in the number of communities and converged to 1.66% for 7 communities or more.

5.3 Real-World Networks

In most of the real-world networks, there is no ground truth node labeling therefore comparison between different algorithms becomes difficult. Most of the comparisons have been made by the maximum modularity achieved by a particular algorithm. However it has been found that modularity may not be an appropriate measure for the goodness of the partitions, especially when the size of the communities vary significantly. To avoid this issue, we perform experiments on the real world networks with known ground truth communities. The performance comparisons are made by using the Normalized Mutual Information (NMI) between the found and

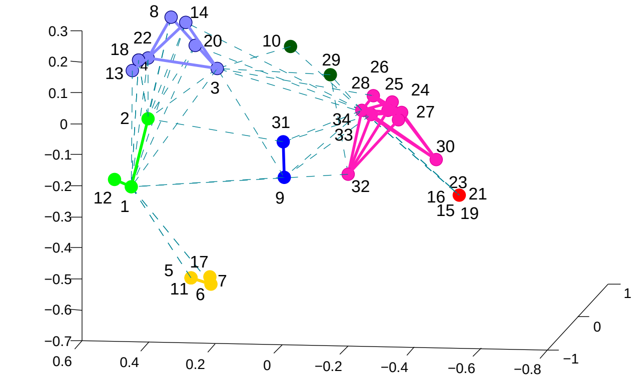

Fig. 9. Zachary Karate Club Network, in 3D space spanned by the three principal components. The network is divided into seven communities using the proposed algorithm. All communities except {9, 31} have consistent labels. Node 9 has label 0 while node 31 has label 1.

the ground truth communities. Experimental settings are given in Table 1.

Zachary Karate Club [55] has been considered as a benchmark social network of friendships in a karate club. It has 34 nodes and 78 links. After a dispute, the club split into two new groups named Mr. Hi and John A. For two partitions of the network, the proposed SSCF algorithm achieved NMI of 0.785. Only the individual 9 is incorrectly classified which is in accordance with other algorithms. Person 9 was a weak supporter of John A but he joined the club of Mr. Hi after the split due to technical reasons rather than his friendships in the club. Our results are better than many existing algorithms including CliquePerc [43], Conclude [11], Demon [9], Ganix [54], Infomap [45], InfomapSingle [46], LinkCommunities [2] and Louvain [4] as reported in a recent study by Hric et al. [23]. The sparse reconstruction error ϵ_s in (9) for the two ground truth communities is 221.82 while for the two found communities is 220.11 which also demonstrates that the found communities represent slightly better structure of the network. Fig. 8 shows the variation of clustering error ϵ_e in (10) as the communities are increased from 2 to 10. The clustering error reduction rate is initially 46.22% and gradually reduced to 1.66% from 7 or more communities. This shows that the actual number of communities in this network is 7 (Fig. 9).

Since NMI is a measure of similarity between the marked and the found communities, as the number of found communities is increased to more than two, NMI decreases accordingly. Therefore, variation of NMI with increasing number of communities is not shown for this network. However we observe that the marked communities in this network are not compact, rather each of the marked community has a group of smaller communities. Our algorithm has identified these smaller communities, without violating the boundaries of the two coarser communities (Fig. 9). Therefore the network structure predicted by the proposed algorithm matches the natural structure of the network.

American college football dataset [20] is a network of football games between Division IA colleges during the regular season in Fall 2000. It has 115 nodes (teams) and 631 links. If two teams played a game a link was marked and the teams were divided into 12 conferences. Variation of accumulative error in the Euclidean space ϵ_e in (10) and the NMI between the detected partitions and the ground truth labels is plotted in Fig. 10. As the number of communities is increased, clustering error decreases at a constant rate of 8.635% as shown by the dotted line from 2 to 10 communities. At 11 communities the error rate reduced significantly and after 12 communities the error rate again became stable at 0.828% which is 10 times less than the initial rate of reduction. Both of the rates are shown by the dotted lines in Fig. 10. Thus based on the error analysis, the proposed SSCF algorithm was able to find 12 communities in this network.

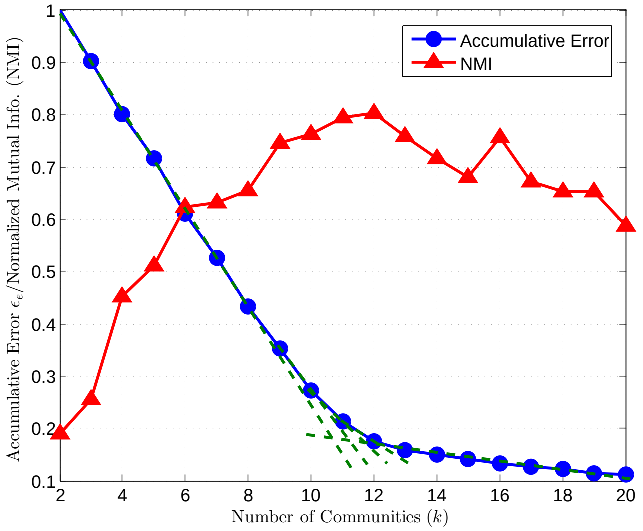

Fig. 10. American college football dataset: variation of accumulated error in the Euclidean space ϵ_e in (10) and normalized mutual information (NMI) with the varying number of communities. Error for the two communities is considered to be 100 % and the rest of the errors are scaled accordingly. Maximum error reduction occurs at 12 communities that matches with the known number of marked communities.

Normalized Mutual Information is measured between the found communities and the ground truth as shown in Fig. 10. NMI increases as the number of communities is increased and achieves a maximum value of 0.793 for 12 communities. We observe that two communities in this network {37, 43, 81, 83, 91} and {12, 25, 51, 60, 64, 70, 98} have a weak structure, therefore cause most of the error. Once these two communities are removed, our algorithm achieves NMI of 0.945. Compared to the existing algorithms, our results are similar or better than the results of Conclude [11], Copra [22], Demon [9] and Ganix [54] as reported by Hric et al. [23]. In the original network, for the 12 found communities, the reconstruction error given by Eq. (9) is 20.860 which is again less than 20.979, the error for the ground truth. This fact also shows that the ground truth labels are relatively less dependent on the network structure. The 12 found communities are shown in Fig. 11. Community level comparison reveals the individual detection accuracy in percentage as {100, 100, 100, 100, 90, 20, 100, 100, 100, 100, 57.14, 80}. That is, eight communities are found with full accuracy while two communities {6, 11} with non compact structure caused most of the error.

Political blogs network (polblog) consists of 1490 nodes which are weblogs on U. S. politics, recorded in 2005 by Adamic and Glance [1]. It is a directed network of 16716 front-page hyperlinks between weblogs at the time of the crawl. The network is divided into two groups. Each weblog is assigned a label by either blog directories or by self-evaluation. For the case of two communities, NMI between the found and the ground truth communities is 0.764. The sparse reconstruction error ϵ_s in (9) for the two found communities is 10727 while for the ground truth communities

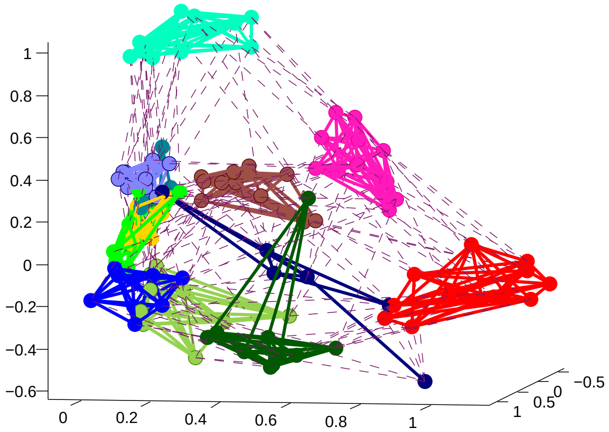

Fig. 11. American college football Network projected on three PCs and divided into 12 communities using the proposed algorithm. Edges across communities are shown as dotted while edges within the communities are shown as solid lines.

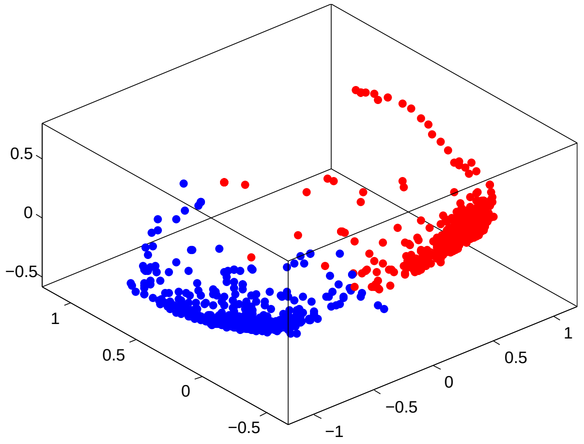

Fig. 13. Political blogs network (polblog): The two communities found by the proposed SSCF algorithm. NMI of this partitioning scheme with the two marked communities is 0.764.

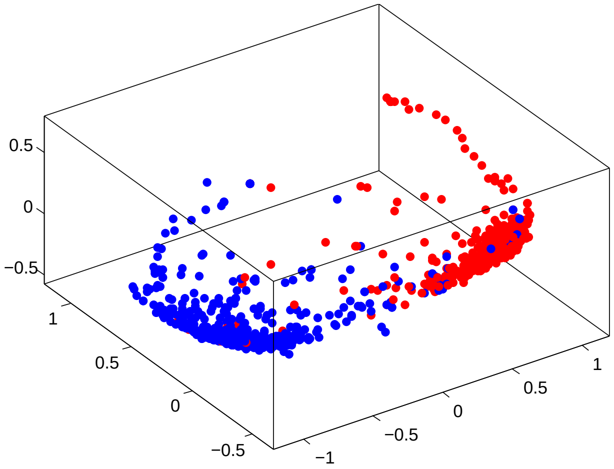

Fig. 12. Political blogs network (polblog): Marked community labels. Some blue labels well within the red community and some red labels within the blue community can be seen which contradict the network structure. Such labels are not possible to be recovered by using network structure alone.

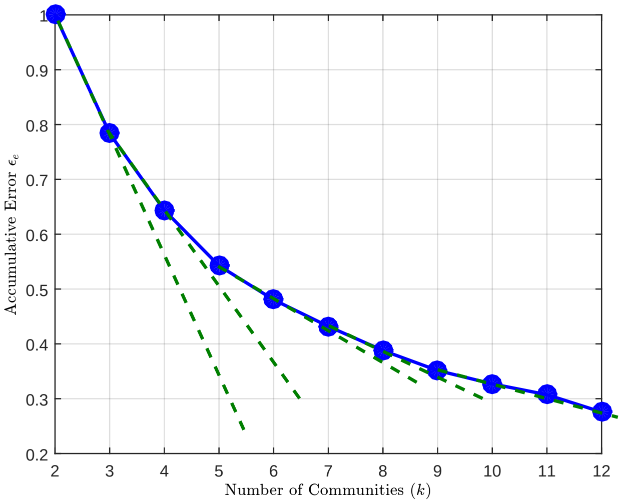

Fig. 14. Political blogs network (polblog): variation of accumulated error ϵ_e in (10) with the number of communities k as percentage of error at two communities. The rate of error reduction from 2 to 3 communities is 21.64 and it reduces to 1.44% from 9 to 10 communities. Beyond that the rate of error reduction remains almost the same.

is 10835. Compared to the existing algorithms, our results are better than Conclude [11], Demon [9], Infomap [45], InfomapSingle [46], LinkCommunities [2], Louvain [4], Oslom [30] as reported by Hric et al. [23].

Variation of accumulated clustering error ϵ_e in (10) with the number of communities k is shown in Fig. 14. We observe that the rate of error reduction beyond 9 communities is almost the same, 1.44%. These nine communities are shown in Fig. 15. Number of nodes in different communities in the increasing order are $\{24$ 25 31 46 56 120 165 330 $425\}$. Most of the communities are very compact except the one shown in brown color which is also the smallest community and not a compact one. The community shown in red is in the middle of the network. By increasing the number of communities beyond 9 will only divide the nodes close to the brown or red communities therefore the error reduction is quite small beyond nine communities.

Similar to the Karate network, in the Polblog network the marked communities are quite coarse and contain groups of smaller communities. Our algorithm was able to find these smaller communities successfully without violating the coarser boundaries (Fig. 15). However as the number of communities is increased, NMI between the two marked communities and more than two found communities decreases. Thus the proposed SSCF algorithm has correctly identified the hierarchical structure of the network.

Execution time of the proposed community detection algorithm has been compared for different networks on Intel 2.7GHz quad-core i5 processor machine with 16GB RAM as shown in Fig. 16. For smaller networks such as Karate and Football, the algorithm is quite fast. For the synthetic LFR network having 1000 nodes, the execution time increases with the increasing value of the mixing parameter μ. It is because of the fact that as μ is increased, the community structure reduces and sparsity in the linear coefficient matrix α corresponding to other communities also reduces accordingly. We observe that for well structured sparse networks the optimization given by (4) converges quite fast as compared to the poorly structured dense networks. For the

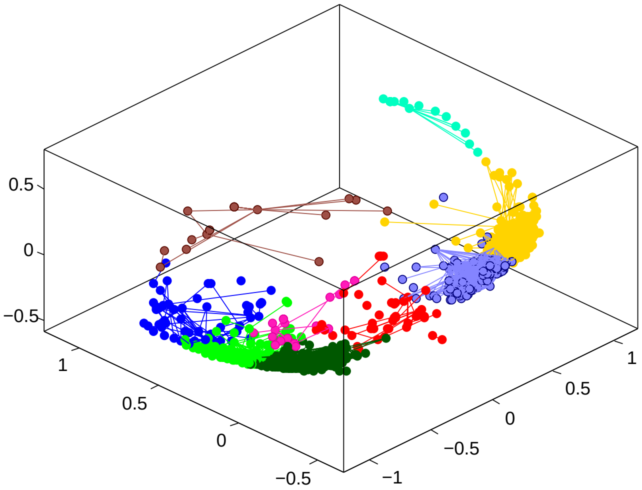

Fig. 15. Political blogs network (polblog) in the 3D space spanned by three principal components of the feature vectors E_x. The nine communities found by the proposed algorithm are shown in different colors. For clarity, inter-community edges are not displayed.

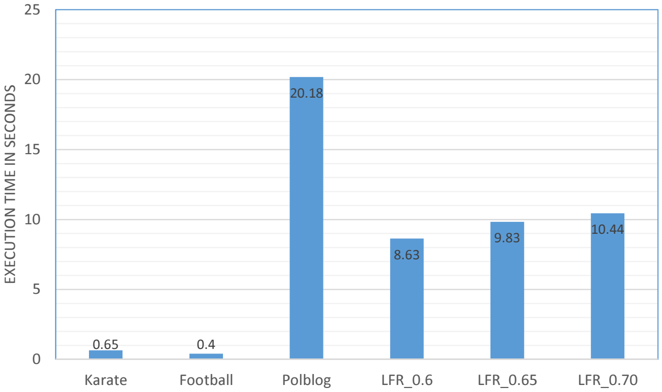

Fig. 16. Execution time of the proposed algorithm on real networks (Karate (nodes=34, edges=78), Football(nodes=115, edges=631), Polblog(nodes=1490, edges=16716)) and synthetic network (LFR μ = $\{0.60, 0.65, 0.70\}$ (nodes=1000, edges=9774)). As μ increases across community edges also increase causing lack of sparsity in coefficients corresponding to other communities resulting in increased execution time.

case of polblog network having 1490 nodes the execution time is almost double than that of the LFR network. It is partially because of increased number of nodes and partially due to reduced sparsity in the network. The ratio of the number of edges to the number of nodes in this network is 11.22 while for the LFR network with $\mu = 0.70$ the same ratio is 9.77 which shows that polblog is more dense network. Both of these factors has contributed to increase the execution time for the polblog network. For networks with larger number of nodes the execution time can be reduced by implementing the proposed algorithm on parallel processing machines. Particularly, the optimization given by (4) has been independently applied for each node, which facilitates parallel implementation of the algorithm.

6 Conclusion

In this paper a subspace based algorithm is proposed for the task of network community detection. The algorithm is based on the fact that each network community spans a different subspace in the geodesic space spanned by the geodesic vectors representing each node. A geodesic vector represents the shortest distance between a given node and all other nodes in the network in terms of number of links. Each node can only be efficiently represented as a linear combination of nodes spanning the same subspace. To make the process of community detection more robust, sparse linear coding with ℓ_1 norm constraint was proposed to be used instead of simple least squares estimation. To find the community labels for each node, spectral clustering was applied on the normalized symmetric linear coefficients. For the goodness of a community scheme, two different criteria were proposed. The proposed community detection algorithm was compared with more than ten state of the art methods on two benchmark networks and three real world networks with known ground truth communities using Normalized Mutual Information. The proposed algorithm consistently outperformed most of the existing algorithms with a significant margin.

Acknowledgments

This work is funded by an Australian Research Council Discovery Project (DP140100203). M. Small is supported by an Australian Research Council Future Fellowship (FT110100896).

References

[1] L. Adamic and N. Glance. The political blogosphere and the 2004 US election: divided they blog. *Proc. 3rd Int. Workshop on Link Discovery*, 411:36–43, 2005.

[2] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.

[3] A. L. Barabasi. *Network Science (online available: barabasilab.neu.edu / networksciencebook)*. 2012.

[4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks j. *Stat. Mech.: Theory Exp.*, page P10008, 2008.

[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of mulimage processing, ieee transactions onliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.

[6] R. F. Cancho, C. Janssen, and R. V. Sole. Topology of technology graphs: Small world patterns in electronic circuits. *Phys. Rev. E*, 64:046119, 2001.

[7] J. Chen and Y. Saad. Dense subgraph extraction with application to community detection. *Knowledge and Data Engineering, IEEE Transactions on*, 24(7):1216–1230, July 2012.

[8] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Phys Rev. E*, 70:066111, 2004.

[9] M. Coscia, G. Rossetti, F. Giannotti, and D. Pedreschi. Demon: A local-first discovery method for overlapping communities. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 615–623, New York, NY, USA, 2012. ACM.

[10] L. Danon, A. Daz-Guilera, and A. Arenas. The effect of size heterogeneity on community identification in complex networks j. *Stat*, 2006.

[11] P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Mixing local and global information for community detection in large networks. *Journal of Computer and System Sciences*, 80(1):72–87, 2014.

[12] D. Deritei, Z. I. Lázár, I. Papp, F. Járai-Szabó, R. Sumi, L. Varga, E. R. Regan, and M. Ercsey-Ravasz. Community detection by graph voronoi diagrams. *New Journal of Physics*, 16(6):063007, 2014.

[13] L. Donetti and M. A. Munoz. Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(10):P10012, 2004.

[14] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[15] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *TPAMI*, 35(11):2765–2781, 2013.

[16] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[17] S. Fortunato and M. Barthlemy. Resolution limit in community detection. *Proc. Natl. Acad. Sci. USA*, 104:36–41, 2007.

[18] M. V. Fragkiskos D. Malliaros. Clustering and community detection in directed networks: A survey. *Physics Reports*, 533(4):95–142, 2013.

[19] M. Girvan and M. E. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[20] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[21] M. Gong, J. Liu, L. Ma, Q. Cai, and L. Jiao. Novel heuristic density-based method for community detection in networks. *Physica A: Statistical Mechanics and its Applications*, 403:71–84, 2014.

[22] S. Gregory. Finding overlapping communities in networks by label propagation. *New Journal of Physics*, 12(10):103018, 2010.

[23] D. Hric, R. K. Darst, and S. Fortunato. Community detection in networks: Structural communities versus ground truth. *Physical Review E*, 90(6):062805, 2014.

[24] X. Jiang, H. Wang, S. Tang, L. Ma, Z. Zhang, and Z. Zheng. A new approach to shortest paths on networks based on the quantum bosonic mechanism. *New Journal of Physics*, 13:013022, 2013.

[25] H. Jin, S. Wang, and C. Li. Community detection in complex networks by density-based clustering. *Physica A: Statistical Mechanics and its Applications*, 392(19):4606–4618, 2013.

[26] A. Lancichinetti and S. Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[27] A. Lancichinetti and S. Fortunato. Limits of modularity maximization in community detection. *Physical Review E*, 84(6):066122, 2011.

[28] A. Lancichinetti, S. Fortunato, and J. Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.

[29] A. Lancichinetti, S. Fortunato, and F. Radicchi. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E*, 78(4):046110, 2008.

[30] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato. Finding statistically significant communities in networks. *PloS one*, 6(4):e18961, 2011.

[31] C. Leberknight, H. Inaltekin, M. Chiang, and H. Poor. The evolution of online social networks: A tutorial survey. *Signal Processing Magazine, IEEE*, 29(2):41–52, March 2012.

[32] Y. Liu, J. Moser, and S. Aviyente. Network community structure detection for directional neural networks inferred from multichannel multisubject eeg data. *Biomedical Engineering, IEEE Transactions on*, 61(7):1919–1930, July 2014.

[33] A. Mahmood, A. Mian, and R. Owens. Semi-supervised spectral clustering for image set classification. CVPR, 2014.

[34] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *ICML*, 2009.

[35] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.

[36] M. Newman. *Networks: an introduction*. Oxford University Press, 2010.

[37] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[38] M. E. J. Newman and E. A. Leicht. Mixture models and and exploratory analysis in networks proc. *Natl. Acad. Sci. USA*, 104:9564–9569, 2007.

[39] G. A. Pagani and M. Aiello. The power grid as a complex network: a survey. *arXiv preprint arXiv:1105.3338*, 2011.

[40] G. Palla, I. Dernyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature (London)*, 435:814–818, 2005.

[41] F. Radicchi. A paradox in community detection. *EPL*, 106:38001, 2014.

[42] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi. Defining and identifying communities in networks proc. *Natl. Acad. Sci. USA*, 101:2658–2663, 2004.

[43] F. Reid, A. McDaid, and N. Hurley. Percolation computation in complex networks. In *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, pages 274–281. IEEE Computer Society, 2012.

[44] P. Ronhovde and Z. Nussinov. Multiresolution community detection for megascale networks by information based replica correlations. *Phys. Rev. E*, 80:016109, 2009.

[45] M. Rosvall and C. T. Bergstrom. An information-theoretic framework for resolving community structure in complex networks proc. *Natl. Acad. Sci. USA*, 104:7327–7331, 2007.

[46] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008.

[47] J. Shi and J. Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, 2000.

[48] L. Tang, H. Liu, and J. Zhang. Identifying evolving groups in dynamic multimode networks. *Knowledge and Data Engineering, IEEE Transactions on*, 24(1):72–85, Jan 2012.

[49] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[50] S. van Dongen. *Graph clustering by flow simulation*. Ph.D. Thesis, Dutch National Research Institute for Mathematics and Computer Science, University of Utrecht, Netherlands, 2000.

[51] C.-D. Wang, J.-H. Lai, and P. Yu. Neiwalk: Community discovery in dynamic content-based networks. *Knowledge and Data Engineering, IEEE Transactions on*, 26(7):1734–1748, July 2014.

[52] X. F. Wang and G. Chen. Complex networks: small-world, scale-free and beyond. *Circuits and Systems Magazine, IEEE*, 3(1):6–20, 2003.

[53] D. J. Watts. Networks, dynamics, and the small-world phenomenon 1. *American Journal of sociology*, 105(2):493–527, 1999.

[54] J. Xie and B. K. Szymanski. Towards linear time overlapping community detection in social networks. In *Advances in Knowledge Discovery and Data Mining*, pages 25–36. Springer, 2012.

[55] W. Zachary. An information flow modelfor conflict and fission in small groups1. *Journal of anthropological research*, 33(4):452–473, 1977.

Arif Mahmood Arif received his Masters and the Ph.D degrees in Computer Science from the Lahore University of Management Sciences in 2003 and 2011 respectively. Currently he is a Research Assistant Professor with the School of Mathematics and Statistics, the University of the Western Australia. His major research interests are in Machine Learning and Pattern Recognition. More specifically he has performed research in data clustering, classification, action and object recognition using image sets. Previously he has worked on the computation elimination algorithms for fast template matching, video compression, object removal and image mosaicing. Currently he is interested in exploring the applications of Machine Learning techniques for the complex network structure characterization.

Michael Small Michael is an Australian Research Council (ARC) Future Fellow and Winthrop Professor in Applied Mathematics in the School of Mathematics and Statistics at the University of Western Australia (UWA). His academic career began with undergraduate and doctoral degrees in Pure and Applied Mathematics at UWA, after a string of post-doc appointments he joined the faculty of the Department of Electronic and Information Engineering of the Hong Kong Polytechnic University (2001-2011). In 2012 he moved back to UWA. He is a Senior Member of IEEE and on the editorial board of several international journals including IEEE Circuits and Systems Magazine and Newsletters. His research interests are in: complex systems, complex network, chaos and nonlinear dynamics, nonlinear time series analysis and computational modelling. Applications of his research include: phenomics, genomics, physiology, biomedical signal processing, financial analysis, granular mechanics, animal movement and behaviour, epidemiology and mechanical systems.