16th World Congress of the International Fuzzy Systems Association (IFSA)
9th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)

Fuzzy Graph Clustering based on Non–Euclidean Relational Fuzzy c–Means

Thomas A. Runkler[1] **Vikram Ravindra**[2]

[1]Siemens Corporate Technology, 80200 Munich, Germany
[2]Technical University of Munich, Germany

Abstract

Graph clustering is a very popular research field with numerous practical applications. Here we focus on finding fuzzy clusters of nodes in unweighted, undirected, and irreflexive graphs. We introduce three new algorithms for fuzzy graph clustering (Newman–Girvan NERFCM, Small World NERFCM, Signal NERFCM). Each of these three new algorithms uses a popular algorithm for crisp graph clustering and combines it with non–Euclidean relational fuzzy c–means clustering (NERFCM). Experiments with artificial and real world data indicate that all three proposed algorithms perform quite well for compact clusters. For less compact clusters, Newman–Girvan NERFCM and Signal NERFCM also perform well. Newman–Girvan NERFCM is more robust to cluster overlaps, and Signal NERFCM yields very smooth membership transitions.

Keywords: graph clustering, relational clustering, social networks

1. Introduction

Finding clusters in graphs is a popular research field with numerous applications such as social network analysis [1, 2], bioinformatics [3], scheduling [4], or optimization of communication networks [5]. A graph may be defined by a real–valued $n \times n$ adjacency matrix A, where n is the number of nodes, and each element $a_{ij} \in \mathbb{R}$ of A, $i, j = 1, \ldots, n$, represents the weight of the directed edge from node i to node j. In this paper we consider unweighted, undirected, and irreflexive graphs, so we require $a_{ij} \in \{0, 1\}$ for all $i, j = 1, \ldots, n$ (unweighted), $a_{ij} = a_{ji}$ for all $i, j = 1, \ldots, n$ (undirected), and $a_{ii} = 0$ for all $i = 1, \ldots, n$ (irreflexive). We may consider clusters of nodes or clusters of edges. This paper deals with clusters of *nodes*. A (node) cluster in a graph is a subset of nodes. Pairs of nodes in the same cluster are more strongly connected than pairs of nodes in different clusters. We often have graph structures that do not exhibit a well–separable cluster structure. Clusters may be overlapping, so we want to be able to *partially* assign each node to several clusters, which are then called *fuzzy graph clusters*.

Popular approaches to *crisp* graph clustering include the Newman–Girvan [6], the Small World [7], and the Signal [8] algorithms. For a survey of these and other crisp graph clustering methods we refer to [9]. Also for *fuzzy* graph clustering several approaches have been proposed in the literature. Nepusz *et al.* [10] minimize a modified fuzzy c–means functional [11] by gradient-based optimization. Zhang *et al.* [12] map each node to a point in the Euclidean space and then use conventional fuzzy c–means clustering. Havens *et al.* [13] optimize a fuzzy generalization of the Newman–Girvan centrality function [6]. Runkler *et al.* [14] transform the graph adjacency matrix to a dissimilarity matrix and then apply relational fuzzy clustering, more specifically non–Euclidean relational fuzzy c–means (NERFCM) [15]. The advantage of this approach is that it can be easily extended to other relational clustering schemes such as medoids [16] or possibilistic clustering [17]. This paper extends the ideas in [14] to perform fuzzy graph clustering using the NERFCM model, but we do not apply NERFCM to adjacency matrices but use the three popular crisp graph clustering methods listed above (Newman–Girvan, Small World, Signal) to produce dissimilarity matrices which are then clustered by NERFCM. So, this paper introduces and compares three new algorithms for fuzzy graph clustering: Newman–Girvan NERFCM, Small World NERFCM, and Signal NERFCM.

This paper is structured as follows. In section 2 we briefly present the considered crisp graph clustering algorithms Newman–Girvan, Small World, and Signal. In section 3 we quickly review relational clustering, and specifically focus on the NERFCM model. In section 4 we modify the crisp graph clustering algorithms from section 2 for fuzzy graph clustering using NERFCM. In section 5 we show some experiments with artificial and real world data, where we compare the performance of our three new fuzzy graph clustering algorithms. In section 6 we give our conclusions and sketch some promising future research directions in this field.

2. Crisp Graph Clustering

In this section we briefly present three popular algorithms for crisp graph clustering: Newman–Girvan, Small World, and Signal.

The idea of the Newman–Girvan algorithm [6] is to iteratively remove the edges that contribute most to the connectivity of the graph, so the graph is sep-

arated into several disconnected subgraphs which represent the graph clusters. In each step of the Newman–Girvan algorithm we compute the shortest paths between all pairs of nodes. For a connected graph with n nodes we have $n \cdot (n-1)$ shortest paths. Then, for each edge we compute the so–called *centrality*, i.e. we count, in how many of the shortest paths the particular edge is contained. Finally, the edge with the largest centrality (i.e. the edge that is contained in most of the shortest paths) is removed. This process is repeated, until the desired number of disconnected subgraphs (clusters) is achieved.

The Small World algorithm [7] does not consider graph connectivity but *cycles* in the graph, more specifically cycles of length three, i.e. triangles. The structure of the Small World algorithm is similar to the Newman–Girvan algorithm. For each edge we compute the coefficient

$$c_{ij} = \frac{z_{ij} + 1}{\min\{k_i - 1, k_j + 1\}} \tag{1}$$

where z_{ij} is the number of triangles through the edge between nodes i and j, and k_i, k_j are the maximum number of possible triangles through nodes i, j, respectively. We remove the edge with the smallest value of c_{ij} and repeat the whole process until the desired number of clusters is achieved.

The Signal algorithm [8] simulates the signal propagation processes from each of the nodes $i = 1, \ldots, n$ through the graph. In each step one of the nodes, say node i, is assigned one unit of signal, and all other nodes have no signal. First the source node i sends the signal to each of its neighbors. Next, each node sends as many units of signals as it has to each of its neighbors. This signal propagation process is repeated T times. The resulting signal values at the n nodes are stored as the i^{th} row of the $n \times n$ signal matrix S. The whole process is repeated for each other node as a source node. Then we add ones to the main diagonal of the signal matrix

$$s_{ii} = s_{ii} + 1, \quad i = 1, \ldots, n \tag{2}$$

and run (crisp) (non–relational) c–means clustering on S, which yields clusters of nodes.

3. Relational Fuzzy Clustering

Our goal here is to find clusters of objects specified by a matrix D of pairwise dissimilarities, where $d_{jk} \geq 0$ is the dissimilarity between objects j and k, where $j, k = 1, \ldots, n$. If there is a feature vector representation $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^p$ so that for each $j, k = 1, \ldots, n$ we have $d_{jk} = \|x_j - x_k\|$, where $\|.\|$ denotes the Euclidean norm, then we call D a *Euclidean* distance matrix. For an $n \times n$ Euclidean distance matrix D and a number $c \in \{2, \ldots, n\}$ of clusters, the *relational fuzzy c–means (RFCM)* model is defined by minimization of the objective function

$$J_{\mathrm{RFCM}}(U; D) = \sum_{i=1}^{c} \frac{\sum_{j=1}^{n} \sum_{k=1}^{n} u_{ij}^m u_{ik}^m d_{jk}^2}{\sum_{j=1}^{n} u_{ij}^m} \tag{3}$$

where

$$u_{ik} \in [0, 1] \tag{4}$$

$$\sum_{k=1}^{n} u_{ik} > 0 \tag{5}$$

$$\sum_{i=1}^{c} u_{ik} = 1 \tag{6}$$

for all $i = 1, \ldots, c$ and $k = 1, \ldots, n$. u_{ik} denotes the membership of object k in cluster i.

We optimize $J_{\mathrm{RFCM}}(U; D)$ by randomly initializing U and then iteratively updating U using the necessary conditions for extrema of J_{RFCM}.

$$u_{ik} = 1 \Bigg/ \sum_{j=1}^{n} \frac{\sum_{s=1}^{n} \frac{u_{is}^m r_{sk}}{\sum_{r=1}^{n} u_{ir}^m} - \sum_{s=1}^{n} \sum_{t=1}^{n} \frac{u_{is}^m u_{it}^m r_{st}}{2\left(\sum_{r=1}^{n} u_{ir}^m\right)^2}}{\sum_{s=1}^{n} \frac{u_{js}^m r_{sk}}{\sum_{r=1}^{n} u_{jr}^m} - \sum_{s=1}^{n} \sum_{t=1}^{n} \frac{u_{js}^m u_{jt}^m r_{st}}{2\left(\sum_{r=1}^{n} u_{jr}^m\right)^2}} \tag{7}$$

$i = 1, \ldots, c$, $k = 1, \ldots, n$, until some termination criterion holds, for example until all differences between successive estimates of u_{ik} are smaller than a given threshold.

When the distance matrix D is non–Euclidean, then it is possible that RFCM can fail and the resulting membership values may violate constraints (4–6). For example, RFCM might yield memberships $u_{ik} < 0$ or $u_{ik} > 1$. One way of solving this problem is to transform D to a Euclidean distance matrix D_β using the *beta–spread* method [15].

$$D_\beta = D + \beta \cdot B \tag{8}$$

with a suitable $\beta > 0$, where $B \in [0, 1]^{n \times n}$ is the off–diagonal matrix with $b_{ij} = 1$ for all $i, j = 1, \ldots, n$, $i \neq j$, and $b_{ii} = 0$ for all $i = 1, \ldots, n$. The value of β is sucessively increased, i.e. higher values of β are added to the off–diagonal elements of R, until the Euclidean case is achieved, so that the constraints (4–6) are satisfied. This algorithm is called the *non–Euclidean relational fuzzy c–means (NERFCM)* algorithm [15].

4. Fuzzy Graph Clustering based on NERFCM

In this section we introduce three new fuzzy graph clustering algorithms: Newman–Girvan NERFCM, Small World NERFCM, and Signal NERFCM. In each of these three algorithms we use one of the three considered crisp graph clustering algorithms (Newman–Girvan, Small World, and Signal) to

compute a dissimilarity matrix for which we then find fuzzy clusters using NERFCM.

For the Newman–Girvan NERFCM algorithm we first run the Newman–Girvan algorithm until termination, as presented in Section 2. For the resulting graph we compute the centrality matrix again and use this as a dissimilarity matrix D. Then we apply NERFCM to D and finally obtain a fuzzy partition matrix U.

For the Small World NERFCM algorithm we first run the Small World algorithm as in Section 2 and compute the connectivity matrix C for the resulting graph. Connectivity is a similarity measure, so we convert each connectivity value c_{ij} to a dissimilarity value d_{ij} using

$$d_{ij} = 1 - \frac{c_{ij} - c_{\min}}{c_{\max} - c_{\min}} \tag{9}$$

where $c_{\min}$ and $c_{\max}$ are the minimum and maximum elements of C, respectively. Finally, we apply NERFCM to D and obtain a fuzzy partition matrix U.

For the Signal NERFCM algorithm we use the Signal algorithm just as in Section 2 but instead of (crisp) (non–relational) c–means clustering we apply NERFCM to S and obtain a fuzzy partition matrix U.

5. Experiments

In this section we present two sets of experiments with artificial and real world data sets in order to assess and compare the performance of the three proposed fuzzy graph clustering algorithms Newman–Girvan NERFCM, Small World NERFCM, and Signal NERFCM.

In the first set of experiments we consider graphs with five different topologies: complete mesh, cycle, star, incomplete mesh, and tree. Each of the graphs contains $q \in \{5, 10\}$ nodes. Fig. 1 illustrates the five different topologies for $q = 5$ nodes. In each of our

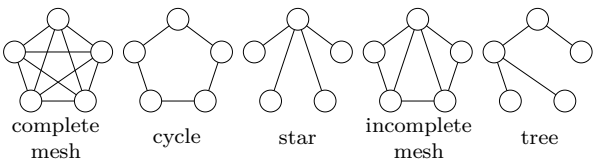

Figure 1: Five different graph topologies.

experiments we construct a graph by first building two equal subgraphs of one of these five topologies and then connecting these two subgraphs with

one single link: one link with $r \in \{2, 4, 6, 8, 10\}$ additional nodes (i.e. $r + 1$ additional edges) between one randomly chosen node in subgraph 1 and the corresponding node in subgraph 2, or

multiple links: $s \in \{1, \ldots, 5\}$ links with $r = 5$ nodes each (i.e. $5 \cdot s$ additional nodes and $6 \cdot s$ additional edges) between s randomly chosen nodes in subgraph 1 and the corresponding nodes in subgraph 2.

Figs. 2 and 3 illustrate examples for the two different linking schemes (single link with length $r = 8$, and $s = 3$ multiple links) for two complete mesh clusters with $q = 5$ nodes each. To avoid intersecting edges we have flipped one of the subgraphs in the Fig. 3.

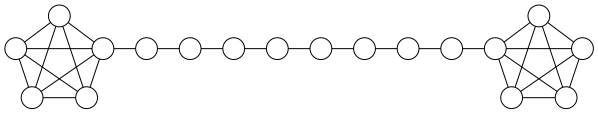

Figure 2: Single link data set for two complete mesh clusters, $q = 5$ nodes, length $r = 8$.

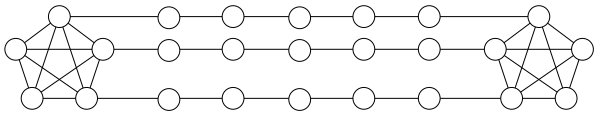

Figure 3: Multiple link data set for two complete mesh clusters, $q = 5$ nodes, $s = 3$ links.

For all experiments we have used random initialization of U and $c = 2$. We have tested several different values for the parameter T of the Small World NERFCM algorithm and used the ones that yielded the subjectively best results for most experiments: $T = 2$ for incomplete meshes, $T = 3$ for cycles and stars, $T = 4$ for trees, and $T = 5$ for complete meshes.

In our experiments we first consider the single link case. We set the number of nodes in each subgraph to $q = 10$, so for a link with $r \in \{2, 4, 6, 8, 10\}$ nodes we have a total of $n = 2 \cdot q + r \in \{22, 24, 26, 28, 30\}$ nodes. We sort the node indices so that $i = 1, \ldots, 10$ are the indices for subgraph 1, $i = 11, \ldots, 10 + r$ are the indices for the link (from subgraph 1 to subgraph 2), and $i = 11 + r, \ldots, 20 + r$ are the indices for subgraph 2. Figs. 4–8 show the membership values for cluster 1 (associated with the first subgraph) for the $n \in \{22, 24, 26, 28, 30\}$ nodes of the two subgraphs and the link. The vertical axes correspond to the ranges $u \in [-0.1, 1.1]$, so points at $u = 0$ and $u = 1$ are not hidden by the bounding boxes of the graphs. Each row in Figs. 4–8 shows the length $r \in \{2, 4, 6, 8, 10\}$ of the single link used in the respective experiment. Each column in Figs. 4–8 shows the results for one of the three algorithms Newman–Girvan NERFCM, Small World NERFCM, and Signal NERFCM. Due to limited space we do not show "NERFCM" in these graphs.

For two complete meshes (Fig. 4) all three algorithms produce memberships that match our intuitive expectation: The first ten nodes (from subgraph 1) obtain a high membership in cluster 1, and the last ten nodes (from subgraph 2) obtain a low membership in cluster 1. For the nodes of the

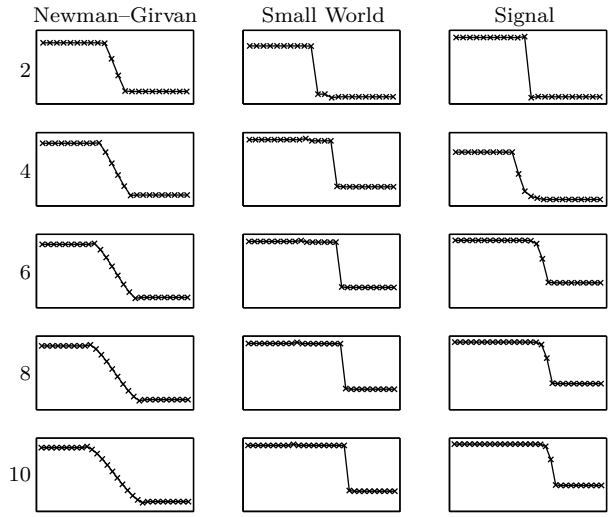

Figure 4: Memberships for two complete meshes connected by one link of length $2, \ldots, 10$.

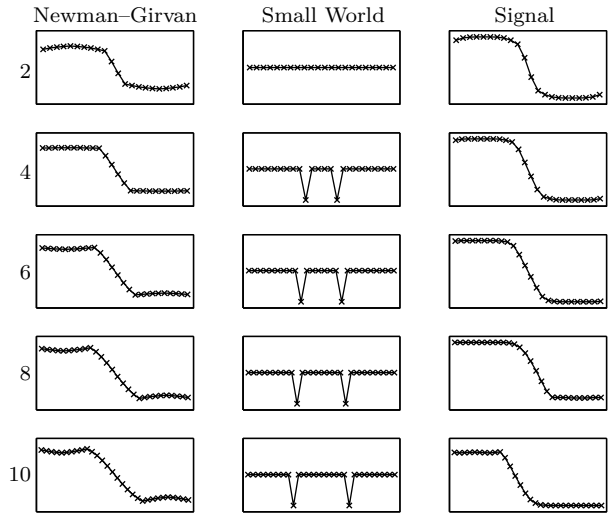

Figure 5: Memberships for for two cycles connected by one link of length $2, \ldots, 10$.

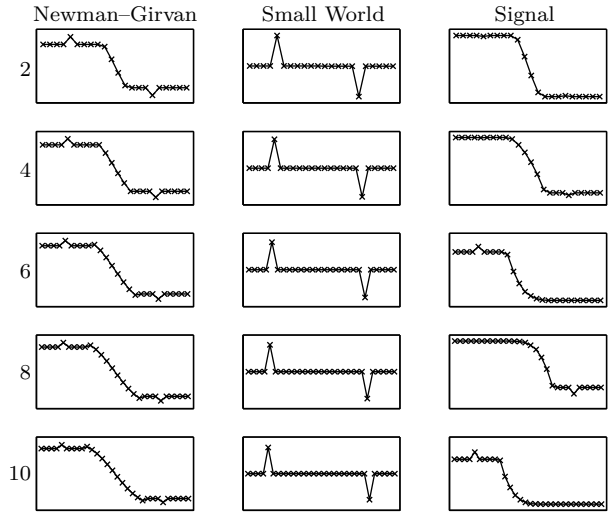

Figure 6: Memberships for two stars connected by one link of length $2, \ldots, 10$.

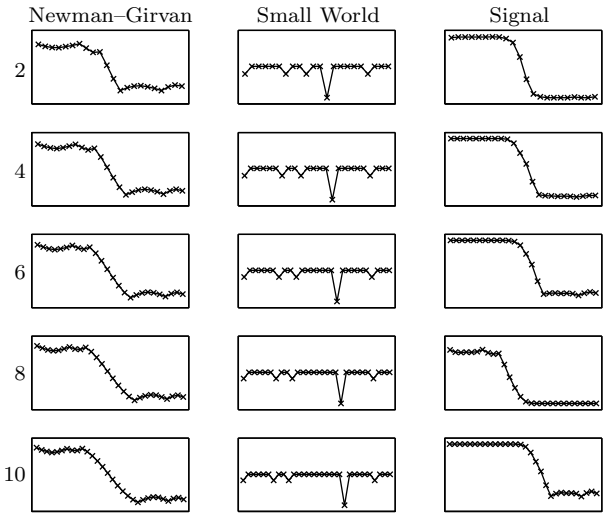

Figure 7: Memberships for two incomplete meshes connected by one link of length $2, \ldots, 10$.

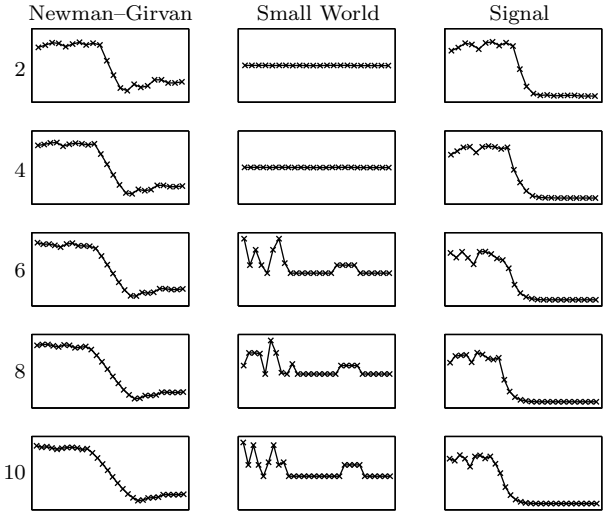

Figure 8: Memberships for two trees connected by one link of length $2, \ldots, 10$.

link between subgraphs 1 and 2 the memberships more or less gradually change from high to low. For Newman–Girvan NERFCM (left column) the memberships change almost linearly as we proceed from subgraph 1 to subgraph 2 (from left to right). For Small World NERFCM and Signal NERFCM (middle and right columns) the change of memberships is more abrupt and occurs at one of the link nodes, but not always at the middle nodes at $i = 10 + r/2$ or $i = 11 + r/2$ as we might have expected.

For two cycles (Fig. 5) Newman–Girvan NERFCM yields very similar results as for two complete meshes (Fig. 4) but less clearer cluster preferences for the nodes in the subgraphs, Small World NERFCM fails, and Signal NERFCM yields very clear cluster preferences for both subgraphs and a very smooth membership transition along the link.

For two stars, two incomplete meshes and two trees (Figs. 6–8) we obtain quite similar results as for two cycles (Fig. 5): Newman–Girvan NERFCM

and Signal NERFCM yield quite good results but Small World NERFCM fails. The most difficult topology seems to be the two trees graph.

After the single link case we now consider the multiple link case. Here we set the number of nodes in each subgraph to $q = 5$, so for $s \in \{1, \ldots, 5\}$ links with $r = 5$ nodes each we have a total of $n = 2 \cdot q + 5 \cdot s \in \{15, 20, 25, 30, 35\}$ nodes. For clarity and easier comparison we will not display the memberships for the nodes in the multiple links but only for the nodes in the two subgraphs. We sort the node indices so that $i = 1, \ldots, 5$ are the indices for subgraph 1 and $i = 6, \ldots, 10$ are the indices for the for subgraph 2. Figs. 9–13 show the membership values for cluster 1 for the $n = 10$ nodes of the two subgraphs (not the links).

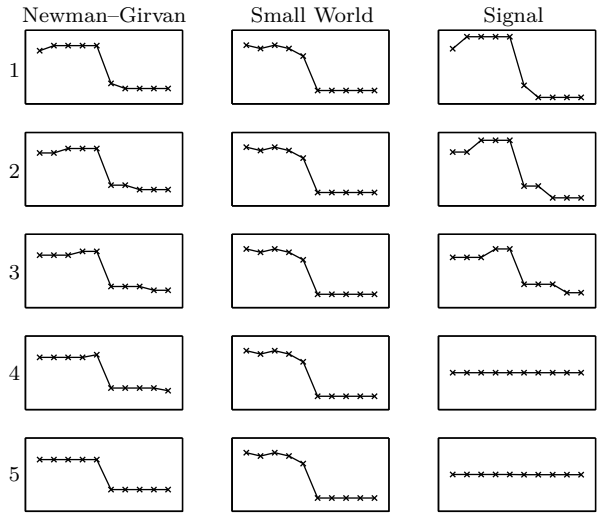

Figure 9: Memberships for two complete meshes connected by $1, \ldots, 5$ links.

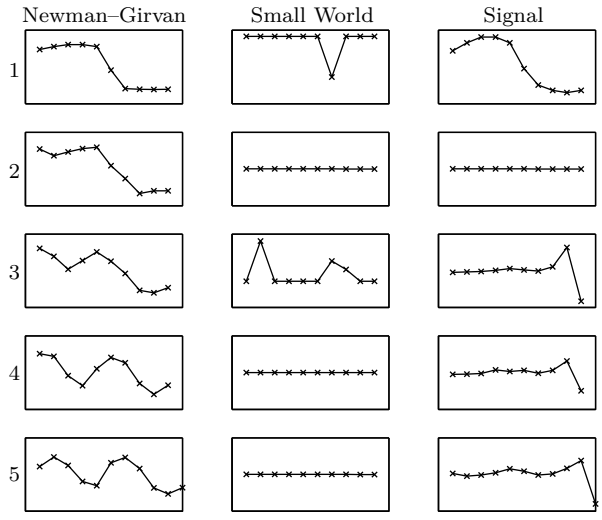

Figure 10: Memberships for for two cycles connected by $1, \ldots, 5$ links.

For two complete meshes (Fig. 9) all three algorithms correctly assign the first five nodes (subgraph 1) to cluster 1 and the other five nodes (subgraph 2)

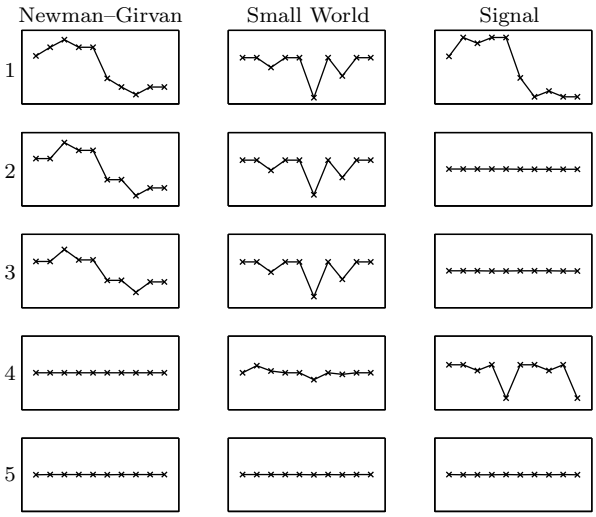

Figure 11: Memberships for two stars connected by $1, \ldots, 5$ links.

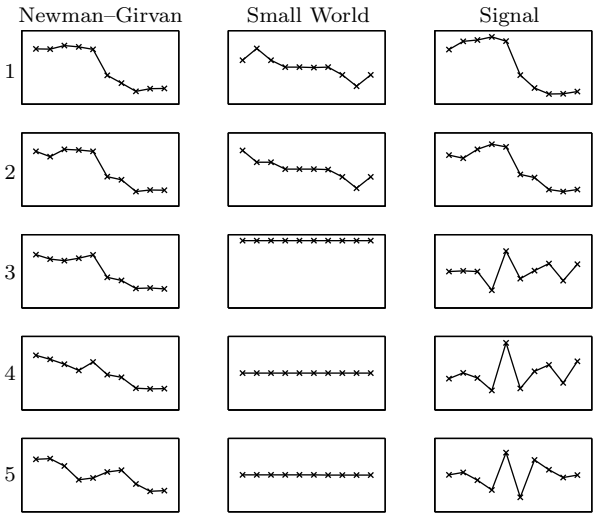

Figure 12: Memberships for two incomplete meshes connected by $1, \ldots, 5$ links.

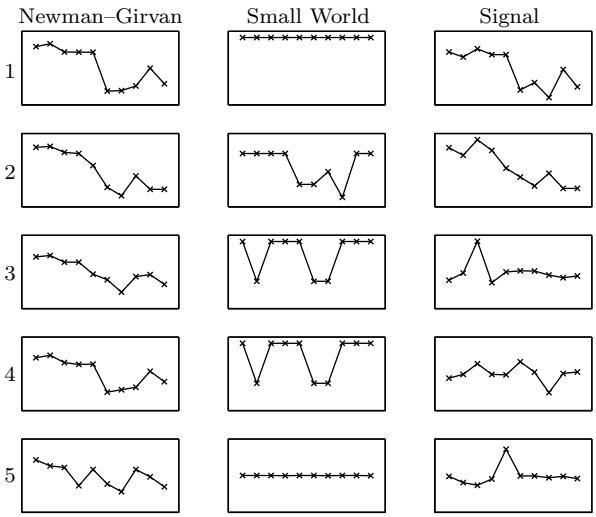

Figure 13: Memberships for two trees connected by $1, \ldots, 5$ links.

to cluster 2. Only Signal NERFCM fails for $s = 4$ and $s = 5$ links.

For all other topologies (Figs. 10–13) Newman–Girvan NERFCM and Signal NERFCM both work quite well for a low number s of links but are more likely to produce bad results as the number s of links increases, i.e. the more the two subclusters are merged. Newman–Girvan NERFCM is a little more robust against an increasing number of links than Signal NERFCM. The Small World NERFCM algorith almost always fails, except for two incomplete meshes with $s = 1$ or $s = 2$ links.

In our second set of experiments we consider Zachary's karate club benchmark data set [18] which represents a social network of friendships between 34 members of a karate club at a US university in the 1970s. In this graph, each node represents a member of the karate club, and each edge represents a tie (or connection) between two club members. The (nonempty irreflexive unweighted undirected) graph has 34 nodes and 78 edges. It is well known from the literature that this graph contains two well separated clusters associated with two key members of the karate club.

Figs. 14–16 show the results of our three new fuzzy graph clustering algorithms for this data set, where for the Small World NERFCM algorithm we set $T = 4$. For the karate club graphs we use the same layout as in [9]. The grey values of the nodes represent the memberhips in one of the clusters, where black and white correspond to the minimum and maximum memberships, respectively.

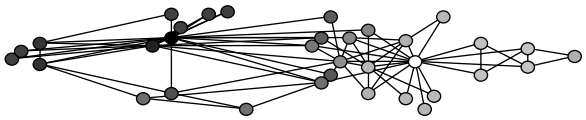

Figure 14: Memberships for the karate data set (Newman–Girvan NERFCM).

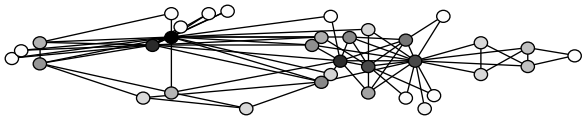

Figure 15: Memberships for the karate data set (Small World NERFCM).

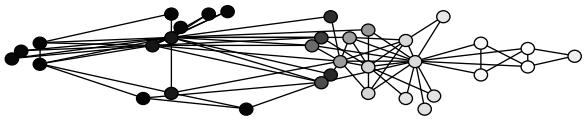

Figure 16: Memberships for the karate data set (Signal NERFCM).

Newman–Girvan NERFCM (Fig. 14) finds two clusters that match well the findings from the literature (one cluster on the left and one cluster on the right), and the two nodes that are often reported as the cluster centers in the literature obtain the highest and lowest memberships (black and white circles). The grey values change quite smoothly from dark grey to light grey as we proceed from left to right.

Small World NERFCM (Fig. 15) produces memberships that do not match the literature at all.

Signal NERFCM (Fig. 16) matches quite well the partition reported in the literature. It does not find the cluster centers from the literature but the transition of the grey values from left to right is much smoother than for Newman–Grivan NERFCM.

6. Conclusions

We have introduced three new methods for fuzzy graph clustering that combine popular non–fuzzy graph clustering algorithms with non–Euclidean relational fuzzy c–means clustering (NERFCM). We call these three new algorithms Newman–Girvan NERFCM, Small World NERFCM, and Signal NERFCM. Our experiments with artificial and real world data sets indicate that all three algorithms perform quite well for compact clusters (complete meshes). For less compact clusters Newman–Girvan NERFCM and Signal NERFCM also work quite well but Small World NERFCM fails. Newman–Girvan NERFCM is more robust to cluster overlaps (more links between the subgraphs) than Signal NERFCM but Signal NERFCM yields smoother membership transitions than Newman–Girvan NERFCM.

Future work in this field should address the following questions: How do the proposed algorithms perform for more than two clusters? How is the performance for other relational clustering schemes, for example NERPCM instead of NERFCM? Can other non–fuzzy graph clustering algorithms be fuzzified in a similar way?

References

[1] J. Scott. *Social network analysis*. SAGE Publications Limited, 2012.

[2] S. Wasserman and K. Faust. *Social network analysis: Methods and applications*, volume 8. Cambridge University Press, 1994.

[3] J. Chen and B. Yuan. Detecting functional modules in the yeast protein–protein interaction network. *Bioinformatics*, 22(18):2283–2290, 2006.

[4] B. Hendrickson and T. G. Kolda. Graph partitioning models for parallel computing. *Parallel computing*, 26(12):1519–1534, 2000.

[5] B. Krishnamurthy and J. Wang. On network–aware clustering of web clients. In *ACM SIGCOMM Computer Communication Review*, volume 30, pages 97–110, 2000.

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

[7] D. J. Watts and S. H. Strogatz. Collective dynamics of small–world networks. *Nature*, 393(6684):440–442, 1998.

[8] Y. Hu, H. Chen, P. Zhang, M. Li, Z. Di, and Y. Fan. Comparative definition of community and corresponding identifying algorithm. *Physical Review E*, 78(2):026121, 2008.

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

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

[11] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.

[12] S. Zhang, R. S. Wang, and X. S. Zhang. Identification of overlapping community structure in complex networks using fuzzy c–means clustering. *Physica A: Statistical Mechanics and its Applications*, 374(1):483–490, 2007.

[13] T. C. Havens, J. C. Bezdek, C. Leckie, J. Chan, W. Liu, J. Bailey, K. Ramamohanarao, and M. Palaniswami. Clustering and visualization of fuzzy communities in social networks. In *IEEE International Conference on Fuzzy Systems*, Hyderabad, India, July 2013.

[14] T. A. Runkler and J. C. Bezdek. Fuzzy relational approaches to graph clustering and visualization. In *GMA/GI Workshop Computational Intelligence, Dortmund*, pages 39–56, December 2013.

[15] R. J. Hathaway and J. C. Bezdek. NERF c–means: Non–Euclidean relational fuzzy clustering. *Pattern Recognition*, 27:429–437, 1994.

[16] R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi. Low–complexity fuzzy relational clustering algorithms for web mining. *IEEE Transactions on Fuzzy Systems*, 9(4):595–607, August 2001.

[17] T. A. Runkler. Kernelized non–Euclidean relational possibilistic c–means clustering. In *IEEE Three Rivers Workshop on Soft Computing in Industrial Applications*, Passau, August 2007.

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