On Spectral Partitioning of Co-authorship Networks

Václav Snášel[1], Pavel Krömer[1], Jan Platoš[1],
Miloš Kudělka[1], and Zdeněk Horák[1]

Department of Computer Science, VŠB-Technical University of Ostrava,
17.listopadu 15/2172, 708 33 Ostrava-Poruba, Czech Republic
{vaclav.snasel,pavel.kromer,jan.platos,
milos.kudelka,zdenek.horak}@vsb.cz

Abstract. Spectral partitioning is a well known method in the area of graph and matrix analysis. Several approaches based on spectral partitioning and spectral clustering were used to detect structures in real world networks and databases. In this paper, we explore two community detection approaches based on the spectral partitioning to analyze a co-authorship network. The partitioning exploits the concepts of algebraic connectivity and characteristic valuation to form components useful for the analysis of relations and communities in real world social networks.

Keywords: spectral partitioning, algebraic connectivity, co-authorship, DBLP

1 Introduction

Spectral clustering (or spectral partitioning) is a useful method for partitioning and clustering of graphs and networks with solid mathematical background and clear interpretation. The ubiquity of social and communication networks in today's information society hand in hand with the increasing power of computers makes the usage of algebraic techniques such as spectral clustering very practical. In this work, we use the spectral partitioning to analyze selected parts of the DBLP[1], a large database of computer science publications. The DBLP can be seen as a vast, dynamic and constantly updated social network that captures several years of author co-operations in the form of joint publications. It is very interesting for social network (SN) researcher because the authors can be easily grouped based on their affiliations, areas of interest, and advisor-advisee relationship. Moreover, we can trace in the DBLP the development of each author's activities, types of activities, areas of interest and so on.

In this paper, we present two spectral partitioning based algorithms to iteratively detect communities in the DBLP (and social networks in general).

[1] http://www.informatik.uni-trier.de/ ley/db/

2 Spectral graph clustering

The basics of the spectral clustering (SC) were introduced in 1975 by M. Fiedler [4]. Fiedler's work defined spectral clustering for both, unweighted and weighted graphs. The following definitions apply to weighted graphs because the edges in a co-authorship network intuitivelly have different weights. An edge between two authors that have published one joint paper has different quality (i.e. weight) than an edge between two authors that have published a large number of joint papers through the years. The frequency, regularity, and age of such co-operations can be a hint for an edge weighting scheme.

Definition 1 (Generalized Laplacian of weighted graph G) *For a graph $G = (V, E)$, the generalized Laplacian is the matrix of the quadratic form*

$$(A_C(G)x, x) = \sum_{(i,k)\in E} c_{ik}(x_i - x_k)^2 \tag{1}$$

The $A_C(G)$ can be easily computed:

$$a_{ik} = \begin{cases} 0 & \text{if } i \neq k \text{ and } (i,k) \notin E \\ -c_{ik} & \text{if } i \neq k \text{ and } (i,k) \in E \end{cases} \tag{2}$$

$$a_{ii} = -\sum_{k\neq i} c_{ik} \quad i,k \in N \tag{3}$$

Definition 2 (Algebraic connectivity of weighted graph G) *The algebraic connectivity of graph G denoted $a_C(G)$ is the second smallest (first non-zero) eigenvalue of $A_C(G)$. Let $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$ be the eignevalues of $A_C(G)$. Then $a_C(G) = \lambda_2$.*

The algebraic connectivity $a_C(G)$ is also known as the Fiedler value [17].

Definition 3 (Characteristic valuation of G) *The characteristic valuation of G (also known as the Fiedler vector of G) denoted $\boldsymbol{a}(G) = (a_1, \ldots, a_n)$ is defined by the values of the eigenvector corresponding to $a_C(G)$.*

The characteristic valuation assigns a non-zero (positive or negative) value to each vertex in the graph in a natural way. There is a number of interesting properties of $a_C(G)$ and $\boldsymbol{a}$, for example [4, 6, 17]:

- $a_C(G)$ is positive iff G is connected.
- if $a_C(G)$ is small, then a graph cut according to the values of vertices in $\boldsymbol{a}(G)$ will generate a cut with good ratio of cut edges to separated vertices.
- $\boldsymbol{a}(G)$ represents an ordering (Fiedler ordering) which can be used for *spectral* partitioning of connected graphs (for the rationale see theorem 1).

Theorem 1 *For a finite connected graph G with n vertices that has a positive weight c_{ik} assigned to each edge (i,k), characteristic valuation $\boldsymbol{a}(G)$, and any $r \geq 0$ let*

$$M(r) = \{i \in N | y_i + r \geq 0\} \tag{4}$$

The subgraph $G(r)$ induced by G on $M(r)$ is connected.

Via theorem 1 can be defined iterative (stepwise) partitioning of connected graph G into connected subgraph $G(r)$ and general subgraph $G \setminus G(r)$. Via theorem 1 can be also defined iterative elimination of vertices with lowest significance to the graph so that the remainder of the graph is connected. The proof of theorem 1 can be found in [4].

2.1 Graph partitioning

A graph $G = (V, E)$ can be partitioned into two disjoint sets A, B such that $A \cup B = V$ and $A \cap B = \emptyset$. The *cut* value, which describes the dissimilarity between the two partitions, can be defined as the sum of weights of the edges removed by the cut [16]:

$$cut(A, B) = \sum_{i \in A, j \in B} c_{ij} \tag{5}$$

It can be shown that the Fiedler vector represents solution for finding partitions A and B such that the following cost function (the *average cut*) is minimized [15, 16]:

$$Acut(A, B) = \frac{cut(A, B)}{|A|} + \frac{cut(B, A)}{|B|} \tag{6}$$

The average cut is a measure with known imperfections [16]. However, its usage is simple and its computation is fast.

3 Related work

As the need for efficient analysis of graph-like structures including social networks is growing, there was much attention given to spectral partitioning and spectral clustering of graphs. In this section, we provide brief state of the art of graph partitioning methods based on spectral clustering.

The use of spectral partitioning for graph analysis was advocated by Spielman and Teng [17]. They have shown that spectral partitioning works well for bounded-degree planar graphs and well-shaped d-dimensional meshes. Today, methods based on spectral clustering are being used to analyze the structure of a number of networks.

An influential study on spectral clustering and its application to image segmentation was published in 2000 by Shi and Malik [16]. The authors approached

the graph partitioning task from the graph cuts point of view. They described the graph cut defined by the Fiedler vector and called it *average cut*. The average cut was shown to be good at finding graph splits whereas the newly defined *normalized cut* was designed to compute the cut costs as a ratio of cut edge weights to all edge weights in the segments. The normalized cut was shown to be useful when seeking partitions that are both, balanced and tight. On the other hand, a study by Sarkar and Soundararajan showed that the increased computational cost of the normalized cut does not result in statistically better partitions [14].

Ding et al. [3] have proposed in 2001 another graph cut algorithm, the *min-max cut*, and showed its usefulness for partitioning real world graphs into balanced parts. Bach and Jordan [1] proposed an algorithm based on a new cost function evaluating the error between given partition and a minimum normalized graph cut. The partitions can be learned from given similarity matrix and vice-versa - the similarity matrix can be learned from given clusters. Similarity of nodes i and j in this context means large weight of the edge (i, j), i.e. large c_{ij}. The method leads to clusters with large in-cluster similarity and small inter-cluster similarity of nodes.

The algebraic connectivity has been used to define a new method for construction of well-connected graphs by Gosh and Boyd in 2006 [5]. The algorithm uses the properties of algebraic connectivity and defines an edge perturbation heuristic based on the Fiedler vector to choose from the set of candidate edges such edges that would improve the value of $a_C(G)$.

The work of Ruan and Zhang [13] presents an application of spectral partitioning in the area of social networks. The authors developed an efficient and scalable algorithm *Kcut* to partition the network to k components so that the modularity Q of community structures is maximized. For more details on Q see [13]. The usefulness and effectiveness of *Kcut* was demonstrated on several artificial and real world networks.

Mishra et al. [12] have used spectral clustering for social network analysis in 2007. They aimed at finding good cuts on the basis of conductance, i.e. the ratio of edges crossing the cut to the minimum volume of both partitions. Volume in this context means the number of edges incident with vertices in the sub-graph. Moreover, the proposed algorithm was able to find overlapping clusters with maximum internal density and external sparsity of the edges.

Kurucz et al. [8, 9] have applied spectral clustering to telephone call graphs and to social networks in general. In their studies, the authors discussed various types of Laplacians, edge weighting strategies, component size balancing heuristics, and the number of eigenvectors to be utilized. The work proposed a *k-way* hierarchical spectral clustering algorithm with heuristic to balance clusters and showed its superiority over the Divide-and-Merge clustering algorithm.

In 2008, Leskovec et al. [10] investigated the statistical properties of communities in social and information networks. They used the *network community profile plot* to define communities according to the conductance measure. Their work demonstrated that the largest communities in many real world data sets

blend with the rest of the graph with increasing size, i.e. their conductance score is decreasing.

Xu et al. [18] have analyzed social networks of spammers by spectral clustering. They have used the normalized cut diassociation measure that is known to minimize the normalized cut between clusters and simultaneously maximize the normalized association within clusters.

A recent work on generalized spectral clustering based on the graph p-Laplacian is due to Bühler and Hein [2]. It was shown that for $p \to 1$ the cut defined by Fiedler vector converges to the Cheeger cut. The p-Spectral Clustering using the p-Laplacian, a nonlinear generalization of the graph Laplacian, was in this paper evaluated on several data sets.

An overview of spectral partitioning with different Laplacians was given by Luxburg in [11]. The study contained a detailed description of the algorithm, properties of different Laplacians and a discussion on suitability of selected Laplacians for given task.

In general, many variants of the basic spectral clustering algorithm were used to partition graphs and detect network structure in multiple application areas with good results. Real world networks and social networks constituted by the natural phenomena of communication, interaction, and cooperation are especially interesting application field for the spectral partitioning.

4 Spectral partitioning of co-author communities in the DBLP

We have defined two iterative partitioning algorithms based on spectral clustering and algebraic connectivity to find co-author communities in the graph. In the algorithm 1 (*simple iterative spectral partitioning*, SimpleISP) was the initial connected graph divided into two subgraphs, each containing vertices with positive valuation (and incident edges) and vertices with negative valuation (and incident edges) respectively. For the next iteration was used as an input the subgraph that contained the *author vertex*. If the *author vertex* belonged to the negative subgraph (that was not guaranteed to be connected), all vertices that were not connected to the *author vertex* were removed. The partitioning ended when the subgraph contained only single vertex (*author vertex*). This variant of the algorithm creates in every iteration a smaller (narrower) community centered around the author.

In the algorithm 2 (*iterative spectral patritioning*, ISP), the graph for next iteration was created differently. In each iteration, we removed all vertices that had lower characteristic valuation than the *author vertex*. It is guaranteed that the resulting subgraph is connected. The algorithm ended when the *author vertex* had the lowest valuation among all vertices in the graph (i.e. it was not possible to remove loosely connected vertices). This variant of the algorithm centers on the community to which the author belongs rather than on the author herself.

Algorithm 1 Simple iterative spectral partitioning (SimpleISP)

Find a connected subgraph S containing the vertex of selected author (*author vertex*), vertices of all his or her co-authors, vertices of all their co-authors, and edges among them.
while $|S| > 1$ **do**
  Compute $\boldsymbol{a}(S)$
  Cut S according to $\boldsymbol{a}(S)$
  Let S^+ contain all vertices and incident edges for which the value of $a(S)_i \geq 0$ and S^- contain all vertices and incident edges for which $a(S)_i < 0$.
  Remove all edges between vertices in S^+ and S^-.
  if *author vertex* $\in S^+$ **then**
    $S = S^+$
  else
    $S = S^-$
  end if
  Remove from S all vertices that are not connected to *author vertex*
end while

Algorithm 2 Iterative spectral partitioning (ISP)

Find a connected subgraph S containing the vertex of selected author (*author vertex*), vertices of all his or her co-authors, vertices of all their co-authors, and edges among them.
repeat
  Compute $\boldsymbol{a}(S)$
  Get the valuation of author vertex $a_{AV} = a(S)_{\text{author vertex}}$
  Remove from S all vertices with valuation lower than a_{AV}. The rest is connected.
until $\min a(S) < a_{AV}$

4.1 Experiments

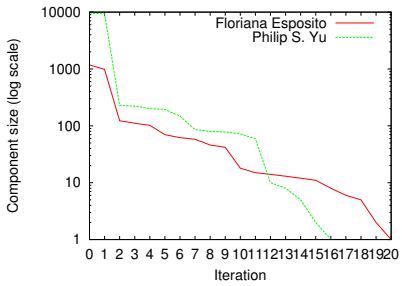
To observe the communities generated by proposed algorithms, we have conducted a series of experiments with the DBLP data. We have downloaded the DBLP dataset from April 2010 in XML and preprocessed it for further usage. We have selected all conferences held by IEEE, ACM or Springer, which gave us 9,768 conferences. For every conference we identified the month and year of the conference. In the next step we extracted all authors having at least one published paper in the mentioned conferences (as authors or co-authors). This gave us 443,838 authors. Using the information about authors and their papers we were able to create a set of cooperations between these authors consisting of 2,054,403 items. Finally, the cooperations were represented as a graph. A vertex in the graph represented one author and an edge represented a co-operation between the authors (joint publication). The edges were weighted according to the number of joint publications between the two authors, i.e. if two authors published one joint work, the weight of the edge between their vertices was 1. If they co-operated on n papers, the weight of the edge between their vertices was n. We note that this weighting scheme is quite naïve and much more sophisticated approaches can be used, but such a research is out of the scope of this paper.

We have selected two authors and investigated spectral partitions of the connected graph consisting of their co-authors and their co-authors' co-authors.

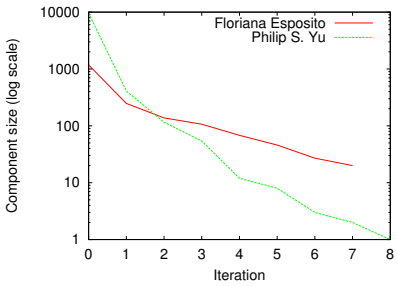
We investigated only two levels of co-authors to obtain components that could be manually inspected. Floriana Esposito and Philip S. Yu were investigated in a recent work on co-authorship network analysis [7]. Floriana Esposito is an author who has been active since 1990 and who has a lot of strong ties whereas Philip S. Yu is an author with the greatest number of records in the data set and with a number of strong co-authors. We have applied both, simple iterative spectral partitioning and strict iterative spectral partitioning to the subgraphs around selected authors.

4.2 Results

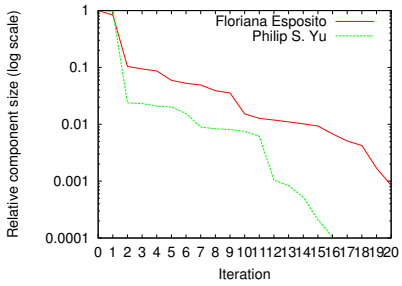
The process of iterative spectral partitioning of subgraphs for Philip S. Yu and Floriana Esposito is captured in Fig. 1. The figures illustrate the sizes of components (communities) of both authors in each iteration of SimpleISP and ISP. Initial size of P. S. Yus component was 9607 and initial size of F. Espositos component was 1180, so for a better comparison, the relative component sizes are compared in Fig. 1(c) and Fig. 1(d). Figures Fig. 1(a) and Fig. 1(c) show the process of SimpleISP. We can see that both authors loose the majority of their collaborators in the second iteration. However, Floriana Espositos network keeps larger fraction of the original nodes during the whole process and it be-



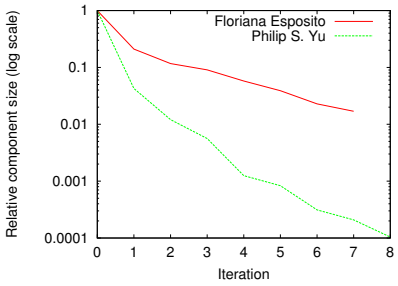
(a) Component size (SimpleISP).



(b) Component (ISP).



(c) Relative component size (SimpleISP).



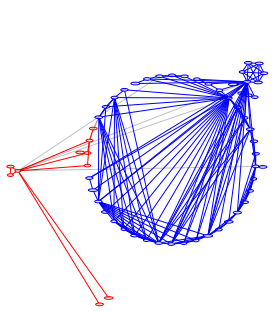
(d) Relative component size (ISP).

Fig. 1: Size of author components during the partitioning.

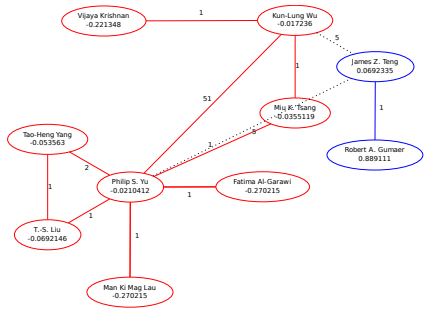
comes larger than Philip S. Yus network after 12th iteration. The SimpleISP ended for Floriana Esposito after 20 iterations and for Philip S. Yu after 16 iterations.

The ISP process is shown in Fig. 1(b) and Fig. 1(d). In this case, the most significant reduction of the communities was done in the first iteration. Floriana Espositos network lost 932 out of 1180 nodes and Philip S. Yus network reduced from 9607 to 410 nodes. Again, the relative component size of Floriana Esposito was greater than the relative component size of Philip S. Yu during the whole ISP and it becomes larger than P. S. Yus community after second iteration. The ISP ended for F. Esposito after 7 iterations and the final network conatined 20 nodes. In contrast, the ISP for P. S. Yu ended after 8 iterations and the final network contained only one node - the *author node*.

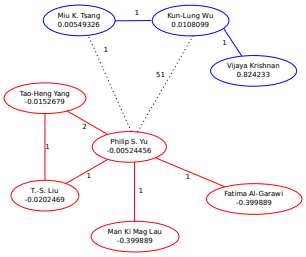
Examples of the partitions in selected iterations of the SimpleISP and ISP for P. S. Yu and F. Esposito are shown in Fig. 2, Fig. 3, Fig. 4, and Fig. 5 respectively. Blue and red vertices and edges represent the components and dotted edges represent the cut. The number on each vertex corresponds to characteristic valuation of the vertex and the number on each edge represents the weight of the edge, i.e. the multiplicity of author co-operation in this experiment. We note that larger graphs are shown to illustrate the structure of the community and cut rather than to provide the names of the co-authors which is printed using very small font.



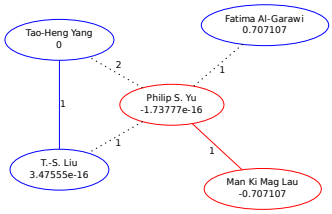
(a) 11th iteration



(b) 12th iteration



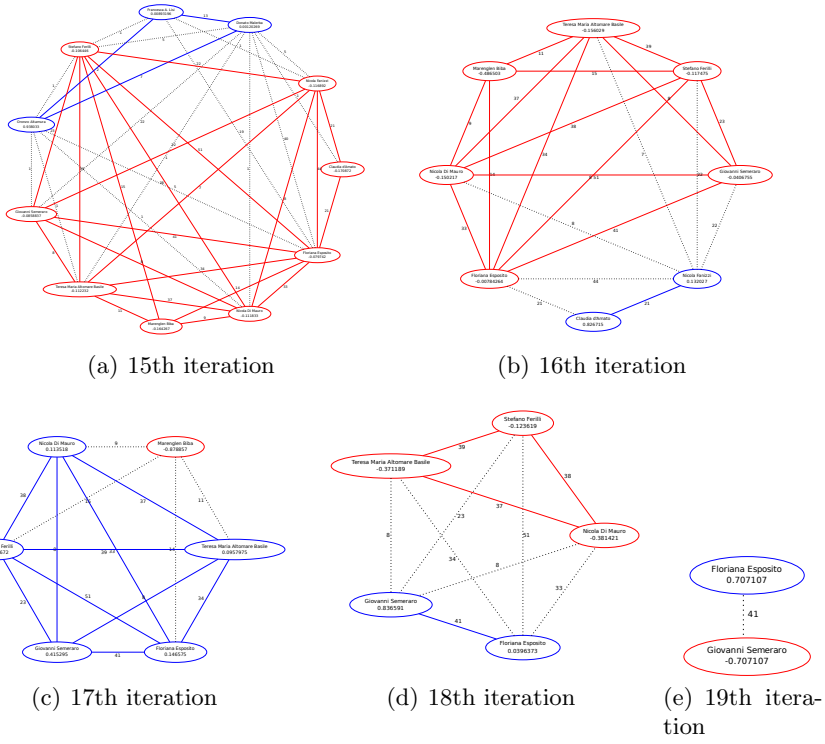
(c) 13th iteration



(d) 14th iteration

(e) 15th iteration

Fig. 2: Philip S. Yus network in selected iterations of SimpleISP.



(a) 15th iteration (b) 16th iteration

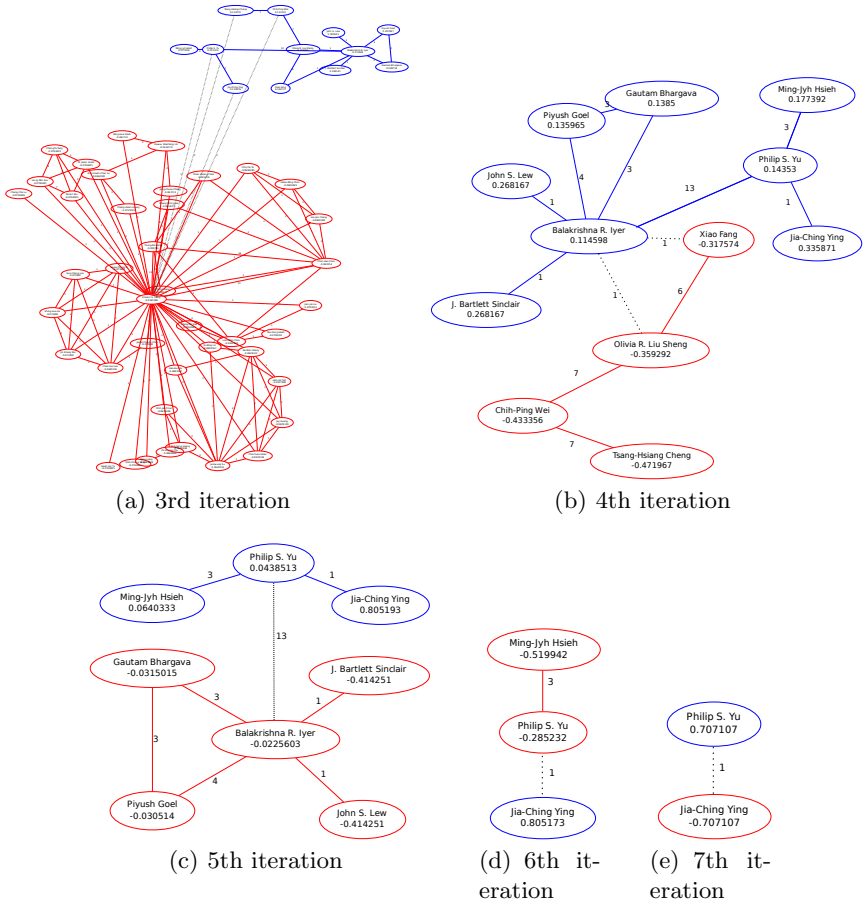
(c) 17th iteration (d) 18th iteration (e) 19th iteration

Fig. 3: Floriana Espositos network in selected iterations of SimpleISP.

5 Conclusions and future work

In this paper we present two algorithms for iterative spectral partitioning of social networks. The goal of the algorithms is to find meaningful communities in networked data. We demonstrate the application of the algorithms on a co-authorship network, namely the DBLP, in which we sought for communities of selected authors. The first algorithm focused on a central node around which it iteratively created connected subgraphs, i.e. possible communities. It was searching for communities around an author. The second algorithm, following more closely the idea of algebraic connectivity and spectral clustering, focused on a community rather than on the author. It was highlighting the community to which the author belonged. We have selected two authors with different statistical properties and searched for their communities using both approaches.

The results of the experiment show that both, the partitioning process and generated partitions, were quite different for the two authors, no matter which algorithm was used. An author with strong ties to other authors retained connection to a large number of co-author nodes during most of the partitioning process. On the other hand, a highly co-operative author lost the links to majority of his/her co-authors very early. The results support the intuition that the



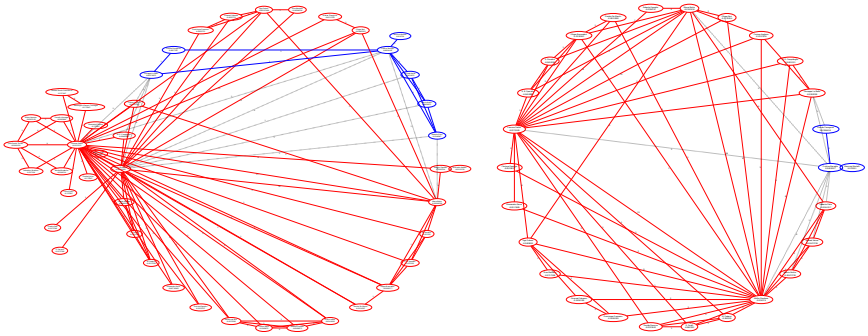
(a) 3rd iteration (b) 4th iteration

(c) 5th iteration (d) 6th iteration (e) 7th iteration

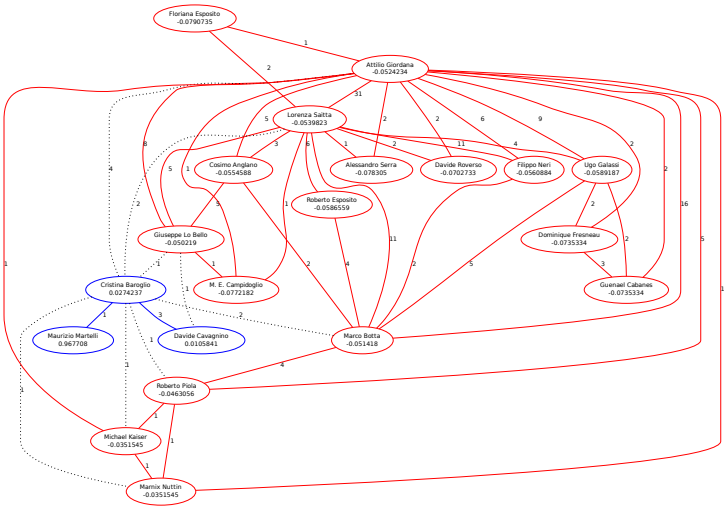
Fig. 4: Philip S. Yus network in selected iterations of ISP.

partitioning of such a different authors will be different. We have also observed, that the author with strong relationships to others was placed to a community of twenty collaborators whereas the highly collaborative author ended alone.

There are many directions in which this work can continue. First, the observations presented in this paper should be confirmed on a large number of authors. Second, the weighting scheme used in this study was rather simple - a different edge weighting schemes should be applied and their influence on the partitioning should be investigated. Third, in this work we have used the simple *average cut* in which we have split the network according to negative and positive values of vertex characteristic valuation. Many different cuts were proposed and their effect on co-authorship network partitioning should be investigated. Also the effect of different Laplacians should be investigated. Finally, the results of the spectral clustering of the co-authorship network should be compared to other non-spectral network and graph analytical methods.



(a) 5th iteration (b) 6th iteration



(c) 7th iteration

Fig. 5: Floriana Espositos network in selected iterations of ISP.

Acknowledgments. This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the Bio-Inspired Methods: research, development and knowledge transfer project, reg. no. CZ.1.07/2.3.00/20.0073 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic.

References

1. Bach, F.R., Jordan, M.I.: Learning spectral clustering. In: Thrun, S., Saul, L.K., Schölkopf, B. (eds.) NIPS. MIT Press (2003)

2. Bühler, T., Hein, M.: Spectral clustering based on the graph p-laplacian. In: Danyluk, A.P., Bottou, L., Littman, M.L. (eds.) ICML. ACM Int. Conf. Proceeding Series, vol. 382, p. 11. ACM (2009)
3. Ding, C., He, X., Zha, H., Gu, M., Simon, H.: A min-max cut algorithm for graph partitioning and data clustering. In: Data Mining, 2001. ICDM 2001, Proceedings IEEE Int. Conf. on. pp. 107 –114 (2001)
4. Fiedler, M.: A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. Czechoslovak Mathematical Journal 25 (1975)
5. Ghosh, A., Boyd, S.: Growing well-connected graphs. In: Decision and Control, 2006 45th IEEE Conference on. pp. 6605–6611. IEEE (2006)
6. Grady, L., Polimeni, J.R.: Discrete Calculus - Applied Analysis on Graphs for Computational Science. Springer (2010)
7. Kudělka, M., Horák, Z., Snášel, V., Krömer, P., Platoš, J., Abraham, A.: Social and swarm aspects of co-authorship network. Logic Journal of IGPL Special issue: HAIS 2010 (2011)
8. Kurucz, M., Benczur, A., Csalogany, K., Lukacs, L.: Spectral clustering in telephone call graphs. In: Proc. of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis. pp. 82–91. WebKDD/SNA-KDD '07, ACM, New York, NY, USA (2007)
9. Kurucz, M., Benczúr, A.A., Csalogány, K., Lukács, L.: Advances in web mining and web usage analysis. chap. Spectral Clustering in Social Networks, pp. 1–20. Springer-Verlag, Berlin, Heidelberg (2009)
10. Leskovec, J., Lang, K.J., Dasgupta, A., Mahoney, M.W.: Statistical properties of community structure in large social and information networks. In: Proceedings of the 17th Int. Conf. on World Wide Web. pp. 695–704. WWW '08, ACM, New York, NY, USA (2008)
11. Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (Dec 2007)
12. Mishra, N., Schreiber, R., Stanton, I., Tarjan, R.E.: Clustering social networks. In: Bonato, A., Chung, F.R.K. (eds.) WAW. Lecture Notes in Computer Science, vol. 4863, pp. 56–67. Springer (2007)
13. Ruan, J., Zhang, W.: An efficient spectral algorithm for network community discovery and its applications to biological and social networks. In: Proceedings of the 2007 Seventh IEEE Int. Conf. on Data Mining. pp. 643–648. IEEE Computer Society, Washington, DC, USA (2007)
14. Sarkar, S., Soundararajan, P.: Supervised learning of large perceptual organization: graph spectral partitioning and learning automata. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22(5), 504 –525 (may 2000)
15. Shen, X., Papademetris, X., Constable, R.T.: Graph-theory based parcellation of functional subunits in the brain from resting-state fmri data. NeuroImage 50(3), 1027–35 (2010)
16. Shi, J., Malik, J.: Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22(8), 888 –905 (aug 2000)
17. Spielman, D.A., Teng, S.H.: Spectral partitioning works: Planar graphs and finite element meshes. Linear Algebra and its Applications 421(23), 284 – 305 (2007)
18. Xu, K.S., Kliger, M., Chen, Y., Woolf, P.J., Hero, III., A.O.: Revealing social networks of spammers through spectral clustering. In: Proceedings of the 2009 IEEE International Conference on Communications. pp. 735–740. ICC'09, IEEE Press, Piscataway, NJ, USA (2009)