

# An ACO Inspired Weighting Approach for the Spectral Partitioning of Co-authorship Networks

Pavel Krömer, Václav Snášel, Jan Platoš, Miloš Kudělka, Zdeněk Horák

Department of Computer Science

FEECS, VŠB Technical University of Ostrava

Ostrava, Czech Republic

Email: {pavel.kromer, vaclav.snasel, 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 use the spectral partitioning to detect communities in a co-authorship network. The partitioning depends heavily on the weighting of the underlying network. We use an intuitive weighting scheme based on the ant colony optimization and show the communities found by spectral partitioning when using the ACO inspired weighting and when using trivial weighting based on the number of interactions between the authors.

**Index Terms**—spectral partitioning; algebraic connectivity; ant colony optimization; weighting; co-authorship; DBLP

## I. 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<sup>1</sup>, 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, advisor-advisee relationship. Moreover, we can trace in the DBLP the development of each author's activities (and also types of activities, areas of interest and so on) in time.

## II. SPECTRAL GRAPH CLUSTERING

The basics of the spectral clustering (SC) were introduced in 1975 by M. Fiedler [1]. 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 intuitively 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

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

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

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

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

**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  $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$  be the eigenvalues of  $A_C(G)$ . Then  $a_C(G) = \lambda_2$ .

The algebraic connectivity  $a_C(G)$  is also known as the Fiedler value [2].

**Definition 3** (Characteristic valuation of  $G$ ). The characteristic valuation of  $G$  (also known as the Fiedler vector of  $G$ ) denoted  $\vec{a}(G) = (a_1, \dots, a_n)$  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  $\vec{a}$ , for example [1]–[3]:

- $a_C(G)$  is positive iff  $G$  is connected.
- if  $a_C(G)$  is small, then a graph cut according to the values of vertices in  $\vec{a}(G)$  will generate a cut with good ratio of cut edges to separated vertices.
- $\vec{a}(G)$  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  $(i, k)$ ,

<sup>1</sup><http://www.informatik.uni-trier.de/~ley/db/>

characteristic valuation  $\vec{a}(G)$ , and any  $r \geq 0$  let

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

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 [1].

The definitions above clearly show that the partitioning depends on the weighting of the underlying graph network because the generalized Laplacian is computed using edge weights. Inevitably, the properties and quality of the graph weighting affects the partitions generated by the spectral partitioning algorithm.

#### A. 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 [4]:

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

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 [4], [5]:

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

The average cut is a measure with known imperfections [4]. However, its usage is simple and its computation is fast.

### III. 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 [2]. 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 [4]. 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 [6].

Ding et al. [7] 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 [8] 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 [9]. 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 [10] 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 [10]. The usefulness and effectiveness of *Kcut* was demonstrated on several artificial and real world networks.

Mishra et al. [11] 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. [12], [13] 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. [14] 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. [15] 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 [16]. It was shown that for  $p \rightarrow 1$  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.

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.

#### IV. ANT COLONY OPTIMIZATION

Ant Colony Optimization (ACO) [17] is a popular meta-heuristic method based on certain behavioral patterns of foraging ants. Emulation of ants' behavior can be used as a probabilistic computational technique for solving complex problems that can be reduced to finding optimal paths in graphs [17]. An artificial ant  $k$  placed in vertex  $i$  moves to node  $j$  with probability  $p_{ij}^k$ :

$$p_{ij}^k = \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{l \in N_i^k} (\tau_{il}^\alpha \eta_{il}^\beta)} \quad (7)$$

where  $N_{ik}$  represents the neighborhood of ant  $k$  in node  $i$  (i.e. nodes that are available to move on),  $\tau_{ij}$  represents the amount of pheromones placed on arc  $a_{ij}$  and  $\eta_{ij}$  corresponds to a-priori information reflecting the cost of passing arc  $a_{ij}$ . After the ants finish their movement forward, they return to the nest with food. The tour of ant  $k$  is denoted as  $T^k$ . The length of  $T^k$ , called  $C^k$ , or the amount of food collected called  $L_k$  (i.e. the solution quality), is used to specify the amount of pheromones to be placed by ant  $k$  on each arc on the trail that led to the food source:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{1}{C^k} & \text{if arc } (i, j) \text{ belongs to } T^k \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

$$\tau_{ij} = \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (9)$$

After all the ants finish one round of their movement, the pheromones evaporate (i.e. the amount of pheromones on each arc is reduced). It can be expressed using the equation  $\tau_{ij} = (1 - \rho)\tau_{ij}$ . The coefficients  $\alpha$ ,  $\beta$  and  $\rho$  are general parameters of the algorithm. This basic version of the ACO algorithm is called an ant system (AS).

In this work, we use a swarm-intelligent weighting algorithm inspired by the ACO proposed first in [18]. It interprets each joint publication of two authors in a co-authorship network as if an ant would pass the link between the authors and updated its pheromones (i.e. its weight). When the two author do not publish a joint paper in a time period, the

pheromones on the edge between them evaporate and their tie becomes weaker.

#### V. SPECTRAL PARTITIONING OF CO-AUTHOR COMMUNITIES IN THE DBLP

We are using a simple iterative partitioning algorithms based on spectral clustering and algebraic connectivity to find co-author communities in the graph. The algorithm 1 (*simple iterative spectral partitioning*, SimpleISP) divides the initial connected graph into two subgraphs, each containing vertices (and incident edges) with positive valuation and vertices (and incident edges) with negative valuation respectively.

---

##### Algorithm 1 Simple iterative spectral partitioning (SimpleISP)

---

```

1: 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.
2: while  $|S| > 1$  do
3:   Compute  $\tilde{a}(S)$ 
4:   Cut  $S$  according to  $\tilde{a}(S)$ 
5:   Let  $S^+$  contain all vertices and incident edges for which the value of
      $\tilde{a}(S)_i \geq 0$  and  $S^-$  contain all vertices and incident edges for which
      $\tilde{a}(S)_i < 0$ .
6:   Remove all edges between vertices in  $S^+$  and  $S^-$ .
7:   if author vertex  $\in S^+$  then
8:      $S = S^+$ 
9:   else
10:     $S = S^-$ 
11:   end if
12:   Remove from  $S$  all vertices that are not connected to author vertex
13: end while

```

---

In the next iteration it takes as an input the subgraph from the previous step 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 ends when the subgraph contains only single vertex (the *author vertex*). The algorithm creates in every iteration a smaller community centered around the author.

##### A. Simple weighting

In this work, we compare the effect of and advanced weighting scheme inspired by the ACO on spectral partitioning. We compare it to a trivial weighting, in which the edges were weighted according to the number of joint publications between the two authors. If the 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  $n$ .

##### B. ACO inspired weighting in co-authorship network

The ant colony metaphor was applied to the co-authorship network weighting as follows: each joint publication of two authors triggers an ant passing between the two nodes. When an ant traverses the arc between two authors, the amount of pheromones on both, author nodes and the link in between, is increased by one. The whole ACO implementation is:

- i) Initialization: initialize the value of pheromones to zero for all vertices and arcs in the network

- ii) Ant movment: each month, when an ant passes the arc, the value of pheromones is increased  $\tau_{ij} = \tau_{ij} \cdot C$  where  $C > 0$  is an constant. We have used  $C = 1.2$ .
- iii) For all objects visited by an ant, let  $\tau_{ij} = S_{ini}$  iff  $\tau_{ij} < S_{ini}$ . This step assigns an extra amount of pheromone to objects that were interacting for the first time  $S_{min} = 12$ .
- iv) Evaporation: for each object that is not visited by an ant in given month, let  $\tau_{ij} = \tau_{ij}(1 - \delta)$  where  $0 < \delta < 1$ . We have used  $\delta = 0.25$ .
- v) For each idle (i.e. not visited) object, let  $\tau_{ij} = 0$  iff  $\tau_{ij} = S_{min}$ . This step erases longer unused connections and removes co-authors that are no longer actively linking with author.

The values of the coefficients  $C$ ,  $S_{ini}$  and  $S_{min}$  were set so that the first joint work of two authors creates a tie that is kept in the network (i.e. its pheromone remains non-zero) for at least 12 months.

### C. Experiments

To observe the communities generated by spectral partitioning when using different weighting schemes, 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 co-operations between these authors consisting of 2,054,403 items. Finally, the co-operations 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 [18]. 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.

### D. Results

The process of iterative spectral partitioning was quite different when using different weighting approaches. The initial

size of P. S. Yus component was 9607 and the initial size of F. Esposito's component was 1180 when using the trivial weighting. The SimpleISP required 20 iterations for P. S. Yu and 16 iterations for F. Esposito.

With ACO weighting, the initial sizes were much smaller because the network 'forgot' old interactions. The initial size of P. S. Yus component was 740 and initial size of F. Esposito's component was 20 with ACO weighting. The partitioning required 13 iterations for P. S. Yu and 7 iterations for F. Esposito.

Examples of the partitions in selected iterations of the SimpleISP when using different weighting schemes for P. S. Yu and F. Esposito are shown in Fig. 1, Fig. 2, Fig. 3, and Fig. 4 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.

## VI. CONCLUSIONS AND FUTURE WORK

In this work was used an ACO inspired approach for assigning weights to edges in a co-authorship network. The ACO mechanisms of pheromone increase and pheromone evaporation were used to model the changing strength of ties between authors in time. Intuitively, the strength of ties in social networks change in time and only a very few models of dynamic social networks use some formalism to model such changes. This work represents an effort to use a well known bio-inspired meta-heuristic method as a model of this aspect of social networks.

The weighting was used as an input for partitioning algorithm that was used to detect author communities in the network. We have selected two significant authors and computed their communities when using the presented weighting scheme and a trivial weighting based on the total number of interactions between authors.

The two weighting schemes lead (as expected) to different communities. With the trivial weighting, the partitioning started with larger initial components. The ACO inspired weighting generated smaller initial components. They captured the current collaborations of the selected author rather than her full professional history.

To illustrate the partitioning process, we have computed communities for two authors. They were selected on the basis of previous research of the DBLP and because they seem to have different qualities. Their communities were different in the final iterations of the SimpleISP when using the trivial weighting: the highly active author had less connected communities than the highly collaborative author. When using the ACO inspired weighting, the communities of both authors were more similar in terms of connectivity.

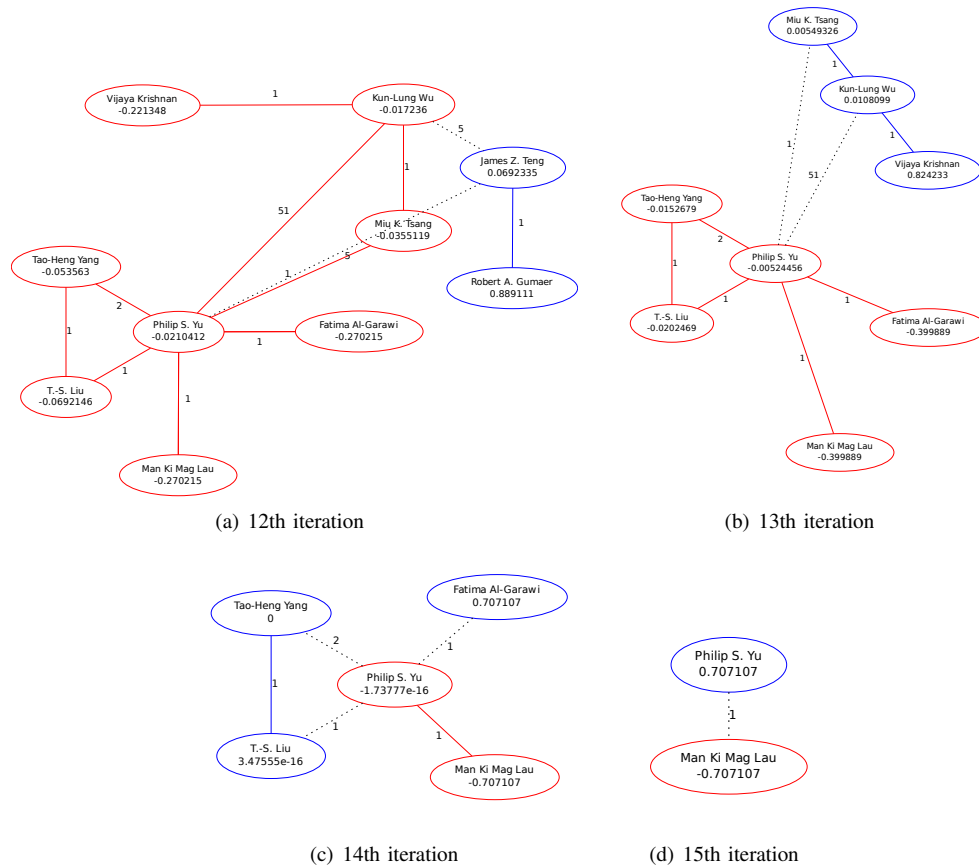


Fig. 1: Philip S. Yus network in selected iterations of SimpleISP when using trivial weighting.

#### ACKNOWLEDGEMENT

This paper has been elaborated in the framework of the IT4Innovations Centre of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070 supported by Operational Programme 'Research and Development for Innovations' funded by Structural Funds of the European Union and state budget of the Czech Republic.

#### REFERENCES

- [1] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory," *Czechoslovak Mathematical Journal*, vol. 25, 1975.
- [2] D. A. Spielman and S.-H. Teng, "Spectral partitioning works: Planar graphs and finite element meshes," *Linear Algebra and its Applications*, vol. 421, no. 23, pp. 284 – 305, 2007. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0024379506003454>
- [3] L. Grady and J. R. Polimeni, *Discrete Calculus - Applied Analysis on Graphs for Computational Science*. Springer, 2010.
- [4] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 8, pp. 888 –905, aug 2000.
- [5] X. Shen, X. Papademetris, and R. T. Constable, "Graph-theory based parcellation of functional subunits in the brain from resting-state fmri data," *NeuroImage*, vol. 50, no. 3, pp. 1027–35, 2010. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/20377452>
- [6] S. Sarkar and P. Soundararajan, "Supervised learning of large perceptual organization: graph spectral partitioning and learning automata," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 5, pp. 504 –525, may 2000.
- [7] C. Ding, X. He, H. Zha, M. Gu, and H. Simon, "A min-max cut algorithm for graph partitioning and data clustering," in *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, 2001, pp. 107 –114.
- [8] F. R. Bach and M. I. Jordan, "Learning spectral clustering," in *NIPS*, S. Thrun, L. K. Saul, and B. Schölkopf, Eds. MIT Press, 2003.
- [9] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Decision and Control, 2006 45th IEEE Conference on*. IEEE, 2006, pp. 6605–6611.
- [10] J. Ruan and W. Zhang, "An efficient spectral algorithm for network community discovery and its applications to biological and social networks," in *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 643–648. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1441428.1442132>
- [11] N. Mishra, R. Schreiber, I. Stanton, and R. E. Tarjan, "Clustering social networks," in WAW, ser. Lecture Notes in Computer Science, A. Bonato and F. R. K. Chung, Eds., vol. 4863. Springer, 2007, pp. 56–67.
- [12] M. Kurucz, A. Benczur, K. Csalogany, and L. Lukacs, "Spectral clustering in telephone call graphs," in *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, ser. WebKDD/SNA-KDD '07. New York, NY, USA: ACM, 2007, pp. 82–91. [Online]. Available: <http://doi.acm.org/10.1145/1348549.1348559>
- [13] M. Kurucz, A. A. Benczúr, K. Csalogány, and L. Lukács, "Advances in web mining and web usage analysis," H. Zhang, M. Spiliopoulou, B. Mobasher, C. L. Giles, A. McCallum, O. Nasraoui, J. Srivastava, and J. Yen, Eds. Berlin, Heidelberg: Springer-Verlag, 2009, ch. Spectral Clustering in Social Networks, pp. 1–20. [Online]. Available: [http://dx.doi.org/10.1007/978-3-642-00528-2\\_1](http://dx.doi.org/10.1007/978-3-642-00528-2_1)

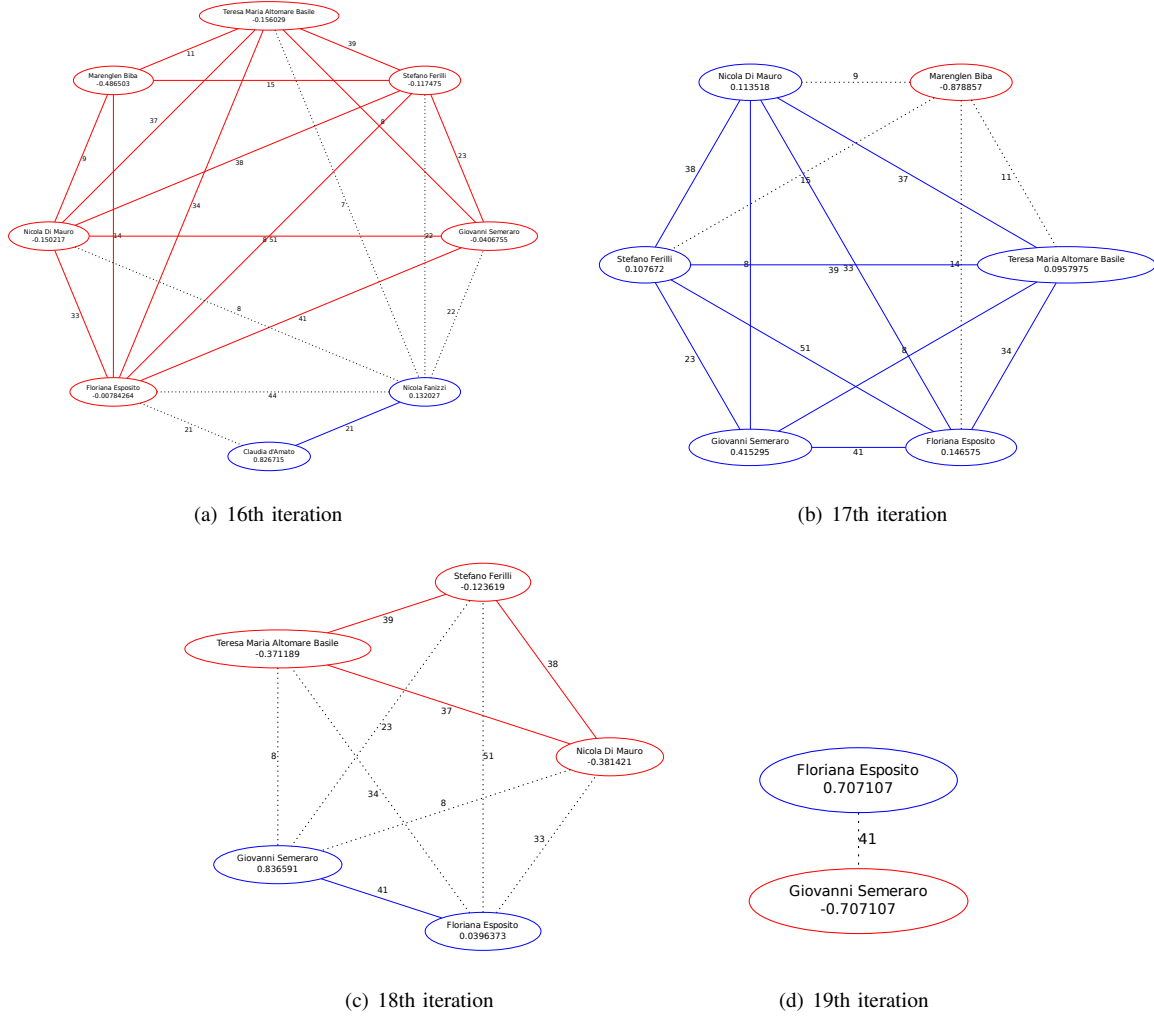


Fig. 2: Floriana Esposito's network in selected iterations of SimpleISP when using trivial weighting.

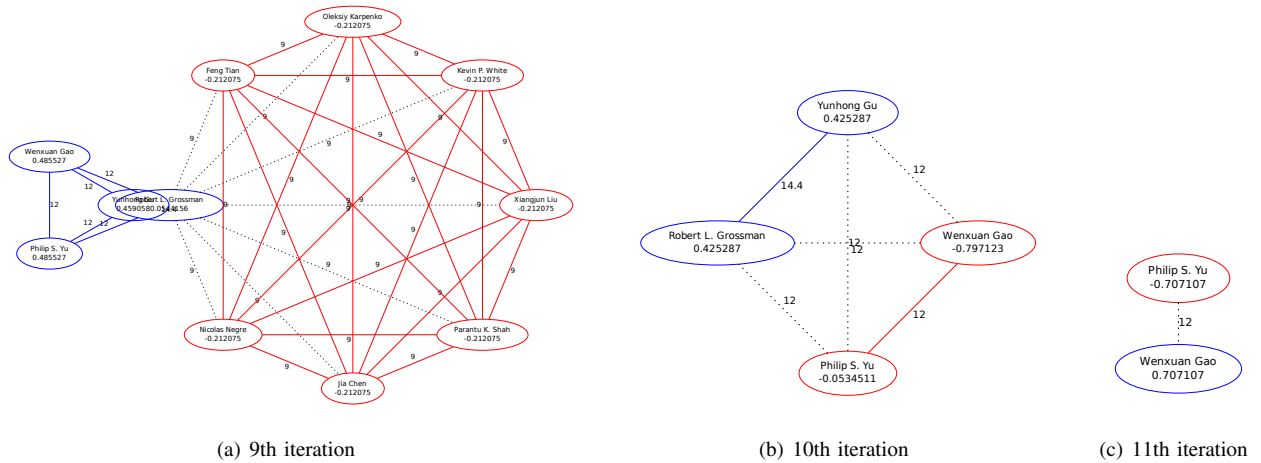


Fig. 3: Philip S. Yus's network in selected iterations of SimpleISP when using ACO weighting.

[14] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Statistical properties of community structure in large social and information networks," in *Proceedings of the 17th international conference on World*

*Wide Web*, ser. WWW '08. New York, NY, USA: ACM, 2008, pp. 695–704. [Online]. Available: <http://doi.acm.org/10.1145/1367497.1367591>

[15] K. S. Xu, M. Klinger, Y. Chen, P. J. Woolf, and A. O. Hero, III,

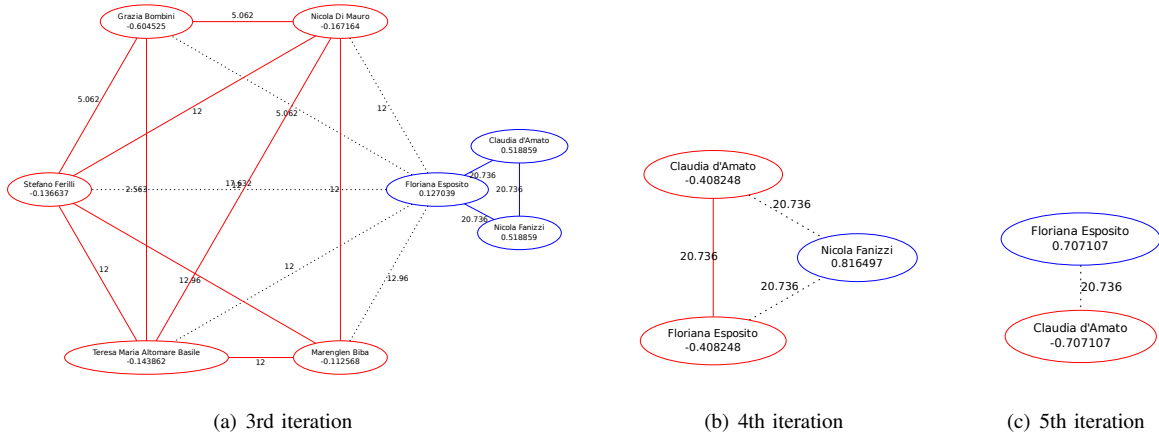


Fig. 4: Floriana Esposito's network in selected iterations of SimpleISP when using ACO weighting.

- "Revealing social networks of spammers through spectral clustering," in *Proceedings of the 2009 IEEE international conference on Communications*, ser. ICC'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 735–740. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1817271.1817409>
- [16] T. Bühler and M. Hein, "Spectral clustering based on the graph  $p$ -laplacian," in *ICML*, ser. ACM International Conference Proceeding Series, A. P. Danyluk, L. Bottou, and M. L. Littman, Eds., vol. 382. ACM, 2009, p. 11.
- [17] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Cambridge, MA: MIT Press, 2004.
- [18] M. Kudřilka, Z. Horák, V. Snášel, P. Krömer, J. Platoš, and A. Abraham, "Social and swarm aspects of co-authorship network," *Logic Journal of IGPL*, vol. Special issue: HAIS 2010, 2011, iSSN 1367-0751(print), ISSN 1368-9894 (online), IF: 0.326 (2008 JCR Science Edition). [Online]. Available: <http://jigpal.oxfordjournals.org/content/early/2011/02/08/jigpal.jzr002.abstract>