Empirical Evaluation of Clustering Algorithms for Large Networks

Guillaume Artignan, Mountaz Hascoët

Empirical Evaluation of Clustering Algorithms for Large Networks

Guillaume Artignan

Mountaz Hascoët

LIRMM, 161 rue ADA, UMR 5506 du CNRS
Université Montpellier II

Abstract— **Clustering is probably one of the most frequently used approaches when facing a scaling problem in large networks. In many situations, however, the choice of the most appropriate algorithm for clustering can turn into a real dilemma. Numerical criteria have been proposed to evaluate the quality of the results of clustering algorithms. However, so many different criteria have been proposed that the dilemma gets even worse. Most criteria reveal different aspects of the quality of the results and hide others. The aim of this paper is to help with the understanding of clustering and to facilitate the comparison and the choice of clustering algorithm for a given purpose. Our proposal consists of studying both quality evaluation criteria and clustering algorithms. We start by discussing a selected set of representative criteria, and further conduct a case study on a large set of real data, measuring not only the quality of different representative clustering algorithms but also the impact of each criterion on the ranking of the algorithms. By providing empirical results on several large-scale corpus of either inter-related documents or lexical networks, we hope to clarify the field and facilitate designers' choices.**

Keywords-component; clustering; networks; quality; visual analysis.

I. INTRODUCTION

Clustering is often cited as the most efficient way so far known to face the challenging scaling problem. Using clustering makes it possible to manage and control large and complex networks at a higher level of abstraction. However, anyone eager to perform clustering on any network has to face a very dense, diverse and mature literature about clustering and choose amongst thousands of algorithms. As recalled by Jain in his recent review on that subject [10], "There is no best clustering algorithm". This does not mean that the choice of the algorithm does not matter. It just means that the choice of the appropriate algorithm for one purpose is multi-factorial by nature. Furthermore, the quality of the results given by different algorithms may vary significantly depending on the criteria used to evaluate the results.. Surprisingly, very few efforts have been made to help non-clustering experts to understand what is at stake and how to compare the quality of various clustering algorithms and the evaluation of the quality of clustering results, remains mostly obscure and difficult to clarify in general.

Our aim is to propose a selection of criteria and to report the results of a case study conducted on large networks of inter-related documents and lexical networks. The purpose of our case study is to evaluate the quality of clustering, and the data on which we have worked was chosen carefully. The first set of data was extracted from "Jeux de Mots", a lexical network of the French language. Jeux de Mots is one of the most accurate and complete publicly available lexical network for French[11]. The second set of data is extracted from papers published in the information retrieval field since 1980 and relations between papers are computed on the fly as will be explained in the next section.

Our proposal is twofold. First, we select and discuss a set of quality criteria and introduce a graph-based representation as a common ground to simplify and unify notations from different fields. As most networks can be conceptually represented by graphs, most current clustering algorithms applicable to networks can be simplified using graph-based representations. This notation is useful to compare and discuss quality criteria.

Second, we report the results of empirical evaluation of clustering quality based on the criteria and visual analysis to explore the results of our experiments.

In this paper, we first describe the datasets used for the experiments. We further describe and discuss the quality criteria used to analytically evaluate the results. We then rapidly review the clustering algorithms selected for the experiment. Finally, the two last sections report the results of the comparative evaluation of the selected set of clustering algorithms, and further discuss them.

II. DATASETS

Four datasets are derived from "Jeux de Mots", *JdmAll* contains *Jdm2000* which contains *Jdm200* containing in turn *Jdm20* and that they respectively contains 111701 nodes, 2000 nodes, 200 nodes and 20 nodes.

We built two datasets from a corpus of 635 research papers in the field of information retrieval and digital libraries. We compute similarities between each pair of documents using the TF-IDF measure [15] and a Pearson's correlation. A large network is then obtained, where nodes are documents and similarities are weighted links. We construct *Sig1000* and *Sig10000* by keeping respectively the 1000 and 10000 best similarity relations. For all these six datasets we studied the degree distribution. Jdm200, *Jdm2000*, *JdmAll* and *Sig1000* have degree distributions following a power-law tail cf. Fig 1. We note γ the exponent. The Tab. 1 describes the datasets by providing the name, the number of nodes N, the total number of edges E, the exponent γ, the graph diameter D, the averaged clustering coefficient C, an URL describing the different datasets and proposing a link for downloading them.
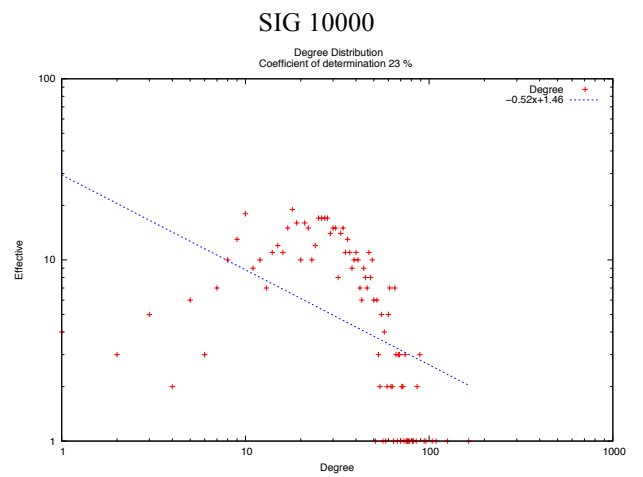
JDM 20

Degree Distribution
Coefficient of determination 76 %

JDM ALL

Degree Distribution
Coefficient of determination of 93 %

JDM 200

Degree Distribution
Coefficient of determination 78 %

SIG 1000

Degree Distribution
Coefficient of determination 76 %

JDM 2000

Degree Distribution
Coefficient of determination 92 %

SIG 10000

Degree Distribution
Coefficient of determination 23 %

Figure 1.   Degree Distribution for each Dataset

Name	|N|	|E|	γ	Diam.	C	URL
JDM 20	20	19	-1.0	6	0.0	Anonymized for Review
JDM 200	200	265	-1,58	11	0.1140	Anonymized for Review
JDM 2000	2000	3476	-1.8	13	0.1357	Anonymized for Review
JDM ALL	111701	441854	-1.9	13	0.1933	Anonymized for Review
SIG 1000	378	903	-1.48	20	0.3928	Anonymized for Review
SIG 10000	626	10000	-0.52	5	0.4002	Anonymized for Review

III. QUALITY CRITERIA

Criteria for the evaluation of the quality of clustering may vary widely in terms of notations and subtlety in terms of concept. We propose notations that can express different criteria in a consistent notation manner in order to help with their comparison. The notation is based on graph theory basic concepts. A graph G is composed of a set of nodes denoted by N and a set of edges denoted by E that represent links between nodes. Applying clustering to G usually results in k clusters denoted by $\{C_1...C_k\}$ as k subsets of N.

To describe the six different criteria selected from the literature and used in the experiment, we further introduce the following notations:

n number of nodes in G
e number of edges in G
k number of clusters after clustering
n_i number of nodes in the cluster C_i
we_i number of edges *within* the cluster C_i
oe_i number of edges *outgoing* from the cluster C_i
be_{ij} number of edges *between* two clusters C_i and C_j
pe_i number of *possible* edges in C_i. For an undirected graph: $pe_i = n_i(n_i-1)/2$. For a directed graph: $pe_i = n_i(n_i-1)$.
me_i number of missing edges in C_i. $me_i = pe_i - we_i$.
we, be, pe and me total number of *w*ithin-cluster *e*dges, *b*etween-cluster *e*dges, *p*ossible *e*dges and *m*issing *e*dges.

$$we = \sum_{i=1}^{k} we_i \quad be = \sum_{i=1}^{k} be_i \quad me = \sum_{i=1}^{k} me_i$$

With these notations, the criteria selected for the evaluation can be written as follows:

Cut [23] is computed as the number of between-edges (also called extra-edges) over the number of within-edges (also called intra-edges). Lowest values correspond to best clustering results.

$$Cut(G) = \frac{be}{we} \tag{1}$$

Perf [22] takes into account the number of undesirable edges that can be considered as errors compared to an ideal clustering. These undesirable links are considered as edges between clusters, as well as missing edges within clusters. The number of missing edges is equivalent to the number of couples of nodes grouped in the same cluster without any edge relating them. Perf finally measures the ratio of undesirable edges over the number of possible edges and compares it to 1. Best values for Perf correspond to highest values.

$$Perf(G) = 1 - \frac{be + me}{pe} \tag{2}$$

Cond [22] criterion equals an average over the conductance of each pair of clusters. The conductance of a pair of clusters C_i and C_j is the proportion of edges between C_i to C_j divided by the minimum number of edges within C_i and C_j. Lowest values correspond to best clustering results. To avoid divisions by zero, we_i or we_j equals are set to 1 in case of singleton clusters.

$$Cond(G) = \frac{\sum_{i<j}^{k} \frac{be_{ij}}{min(we_i, we_j)}}{k(k-1)/2} \tag{3}$$

Cov [22] is the ratio of the number of within-edges to the total number of edges in the graph. It can be considered as the inverse of a normalized version of cut.

$$Cov(G) = \frac{we}{e} \tag{4}$$

MQ [4] is a difference between the average within-cluster edge density and between-cluster edge density. Therefore it varies between -1 and +1, and highest values correspond to best clustering results. In the case of a

singleton cluster, we_i and pe_i equal 0. In this case, we do not compute we_i / pe_i but use the value of 1 instead.

$$MQ(G) = \frac{\sum_{i=1}^{k}(we_i/pe_i)}{k} - \frac{\sum_{i<j}^{k}(be_{ij}/n_i n_j)}{(k(k-1))/2} \qquad (5)$$

Mod [5] can be considered as a measure of *Cov* defined above corrected by the *Cov* computed for a random clustering of the same graph denoted as *rCov*. Therefore, the highest values for Mod correspond to best clustering results according to *Cov* and values below 0 correspond to clustering that are worse than a random clustering according to the *Cov* criteria. However, the computation of *rCov* is probably debatable even though discussing it would lead us beyond the scope of this paper.

$$Mod(G) = Cov - rCov \qquad (6)$$

IV. CLUSTERING ALGORITHMS

Clustering has a huge and multidisciplinary history since it has been used in many scientific fields including in information retrieval [19], data visualization [1], physics [5], etc. Several surveys have partially reviewed this literature [20,10,16]. In order to choose the algorithms to be tested in our study we had three criteria in mind. First, authors either provide source codes for the proposed algorithm or the description of the algorithm is sufficiently clear, complete and precise to be implemented. Second, the algorithm is relevant to clustering data such as complex networks. Third the set of algorithms tested should be representative of different types of clustering approaches. The Tab. 2 summarizes the choices made in terms of algorithms and indicates the URL of the implementation used in the experiment.

The CNM algorithm [5] has a bottom-up approach. Communities are made for each node and further merged iteratively merged with others to increase the criteria of modularity defined in equation Equ. 6. CNM results can be represented by a hierarchical clustered graph or a simple clustered graph depending on how merging is handled.

The BGLL algorithm [3] approach is very similar to CNM, but the definition of modularity differs and it makes the hierarchical clustered graph explicit as well as the level at which the clusters are extracted from the hierarchical clustered graphs.

The CMJA algorithm [2] has a different approach from the two previous ones. CMJA is proposed for detecting communities in small world networks by identifying weak edges. The algorithm operates in two steps. Firstly, it processes a score on each edge, this score is proportional with the number of 4-cycles and 3-cycles containing the edge. Secondly it removes the k edges with the lowest scores. Clusters are the resulting connected components.

The InfoMap approach [14] treats the problem of finding community structures in networks as an information-coding problem. The approach has three steps: (1) Infomap processes a random walk on the graph and generates the random path, (2) assigns a codeword to each node in the random pass using Huffman coding [8], (3) searches a clustering minimizing the average number of bits useful to describe it.

The MCL Algorithm [7] detects communities using a Markov Matrix. The algorithm computes random walks by flow simulation. An operator named "Expansion" computes *n* multiplications of the matrix with itself. An operator named "Inflation" computes the Hadamard

TABLE II. ALGORITHM'S AND IMPLEMENTATIONS USED IN THIS EVALUATION

Algorithm Name	Article	Author's Impl.	Implementation
CNM	[5]	Yes	http://www.cs.unm.edu/~aaron/research/fastmodularity.htm
SPK-MEANS	[6]	Yes	http://www.cs.utexas.edu/users/dml/datamining/spkmeans.html
Cluto	[21]	Yes	http://glaros.dtc.umn.edu/gkhome/views/cluto
LinLog	[13]	Yes	http://www.informatik.tu-cottbus.de/~an/GD/
InfoMap	[14]	Yes	http://www.tp.umu.se/~rosvall/code.html
CMJA	[2]	No	our implementation (link after blind reviews)
BGLL	[3]	Yes	http://sites.google.com/site/findcommunities/
Simple K-Means	[12]	No	our implementation (link after blind reviews)
NCut Algorithm	[18]	Yes	http://www.cis.upenn.edu/~jshi/software/
MCL	[7]	No	http://www.arbylon.net/projects/

matrix [17].

K-Means Algorithm [12] is one of the most frequently used algorithms for clustering and many slightly different versions have been proposed. The main principle is to start with an arbitrary partition of the dataset and try to move each element to a better cluster as long as possible to improve the overall within clusters cohesion. It is one very efficient and very simple algorithm to implement. However, it's based on centroid computation. Therefore it requires that as a prerequisite over other algorithms that meaningful centroids can be computed for the datasets.

LinLog Algorithm [13] is a layout algorithm based on an energy model that aims at geometrically exhibiting clusters. Its principle is to optimize the layout accounting mainly for attraction and repulsion forces between nodes.

The NCut Algorithm [18] comes from the image segmentation domain but can be adapted to graphs. Its principle is to optimize a criterion named "Normalized Cut", using a spectral technique.

The Cluto Toolkit [21] is a toolkit made of several clustering algorithms. Four approaches are tested in this paper: (1) The rb-based clustering approach proposed clustering computed by K-1 bisections, (2) the direct-based clustering approach, (3) an agglomerative approach, (4) the graph-based approach based on a similarity graph and a min-cut criterion.

The Spherical K-Means algorithm [6] is an extension of the well-known Euclidian K-Means algorithm. This algorithm partitions the dimension using great hyper-circles.

V. RESULTS

To analyze the results we started by computing the ranking of each algorithm according to each criterion and for each dataset. Table 3 reports average ranking over different datasets of a subset of algorithms. It also reports the aggregated ranking over all criteria and computed as the average of all criteria.

We further computed the Spearman correlation for each pair of criteria on each dataset and computed an average of all Spearman measures over all datasets. The results are reported in the diagram of Fig. 4 where average Spearman values between two criteria are written next to the edge connecting the two criteria on the diagram. For example, the average value found for Spearman correlation between *cov* and *mod* is 0.43. Overall, with the exception of *1/cut* and *cov*, most criteria are not strictly correlated.

The variability of the results was an incentive to use a visual analysis approach to better understand the cause of variability. Therefore we used parallel coordinate diagrams [9] to interpret the results of the experiments. Figure 2, 3, 5, 6, 7 and 8 show six such diagrams corresponding to the six different datasets. In these diagrams several vertical axes are used to embody different dimensions of the data explored. In our case, the vertical axes were used to embody the different quality criteria. Each item, e.g each algorithm is further represented by a polyline that joins the values corresponding to that algorithm for all criteria. These diagrams offer well-known benefits: (1) covariance/contravariance of ranking between two adjacent criteria is visually obvious, (2) the distribution of the values for each criteria is also visually obvious for all criteria and (3) it is very easy to select an algorithm that performs best for one criteria and to see how it compares to other algorithms in the other criteria.

TABLE III.   AVERAGE OF RANKINGS

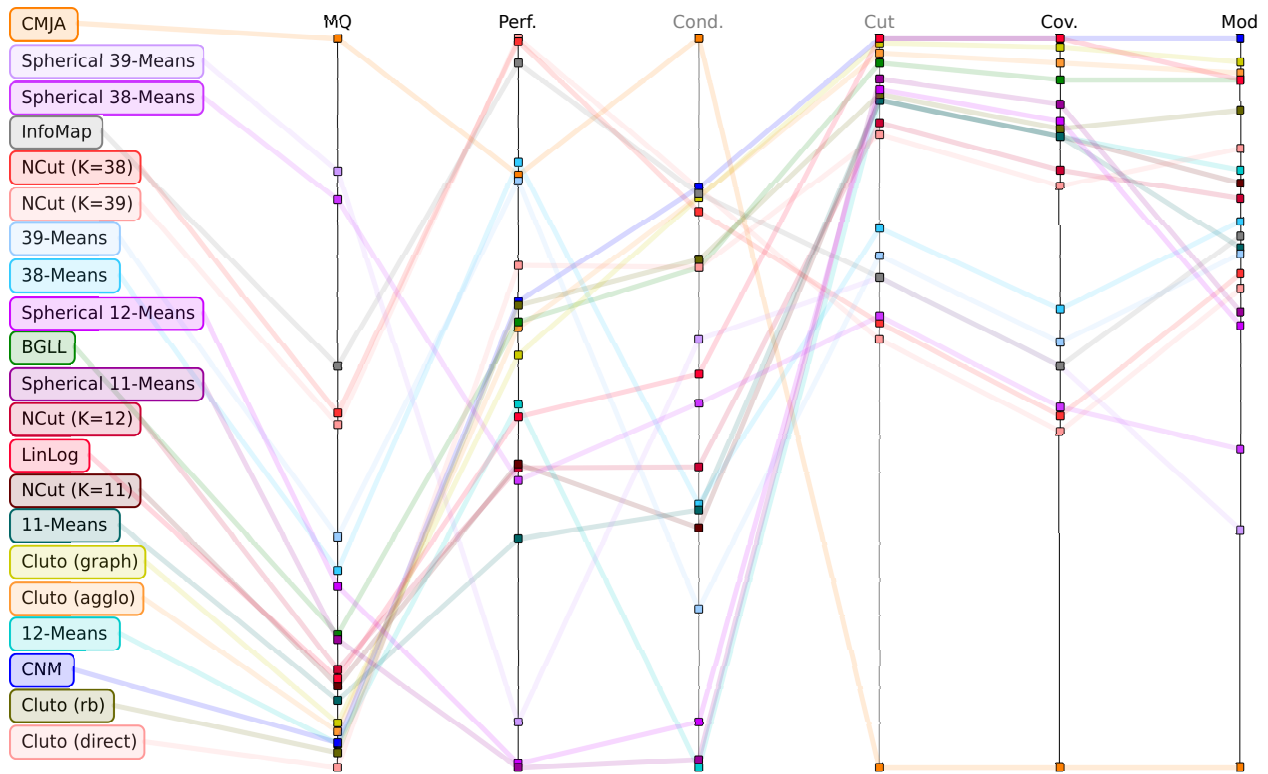| Algorithm	MQ	PERF	COV	CUT	COND	MOD	Total Average
CNM	4.5	4.3333	1.6667	4.5	3.1667	2.3333	3,4167
BGLL	4.1667	3.1667	3.5	3	3.3333	2	3,1933
CMJA	1	4.5	2.8333	3.5	3.1667	6	3,5
InfoMap	2.1667	1.3333	5	1.8333	4.3333	4	3.1111
LinLog	4.3333	3.6666	2.5	3.8333	3.3333	1.5	3.1944
K-Means	3.8333	2.8333	3.6666	2.5	2.1667	4.1667	3.1944

Figure 2. Usage of parallel coordinate on quality Criteria for JDM 200
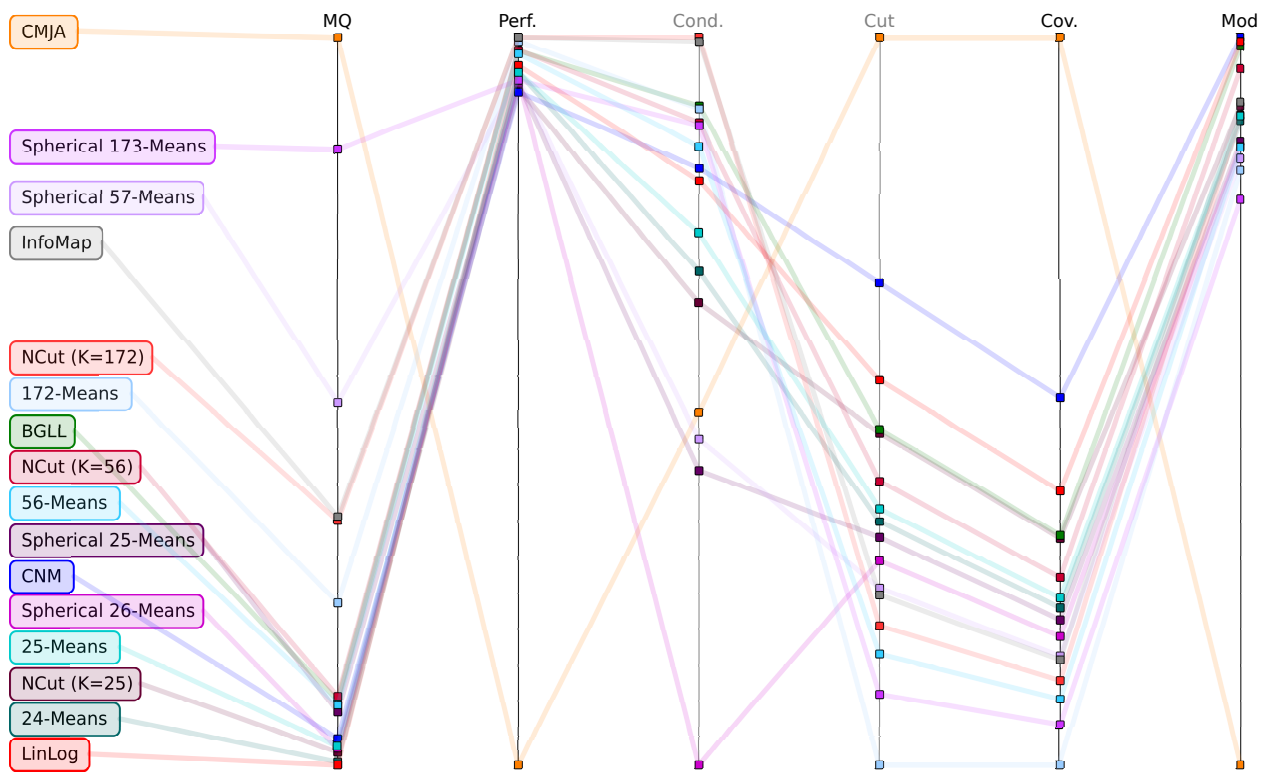

Figure 3. Usage of parallel coordinate on quality criteria for JDM 2000

VI. DISCUSSION

A. General trends : important variations of ranking over criteria and datasets.

Table 2 shows the average ranking for a subset of datasets and algorithms used in the experiment. It illustrates (1) important variations over different criteria and (2) very average ranking for all algorithms when all rankings are combined. These results tend to corroborate the position of both Jain and Buxton. Indeed W. Buxton used to say: "Everything that is best for something is worse for something else". And Jain, recently wrote: "While numerous clustering algorithms have been published and new ones continue to appear, there is no single clustering algorithm that has been shown to dominate other algorithms across all application domains [...] with the emergence of new applications, it has become increasingly clear that the task of seeking the best clustering principle might indeed be futile". However, that said, it is important to better understand the variations of the quality of algorithms measured by different criteria over varying datasets, and how the three interact.
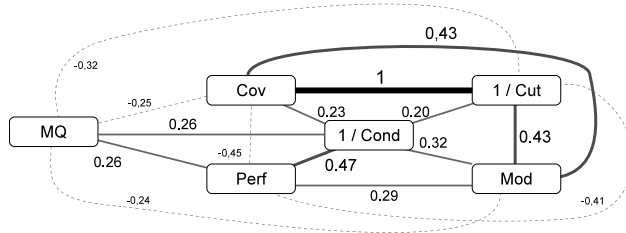

Figure 4. Spearman Rank Correlation on Quality Measures

B. What are the dependencies between Cut, Cov, Cond and Mod ?

All criteria used in the experiment try to capture how similar the elements inside clusters are and how dissimilar the clusters are one from another. Ratio of between-edges over within-edges is used in the definition of four (e.g. *Cut*, *Cov*, *Cond* and *Mod*) out of the six criteria used in the experiment. However these ratio are not exactly computed the same way and small differences in their definition sometimes has huge impact on the results.

Cut and 1/*Cov* are strictly covariant, because *Cov* can be considered as the normalized version of 1/*Cut*. Therefore Spearman's correlation between 1/*Cut* and *Cov* is 1 and parallel coordinates shows no crossing between 1/*Cut* and *Cov* polylines.

Both empirical and analytical results further suggest that *Cov* and *Mod* are partially correlated. Both Spearman and parallel coordinate diagrams clearly show that the two criteria are related but not strictly correlated. As mentioned earlier, *Mod* can be considered as a measure of *Cov* corrected by random, however, more work is needed

to characterize the relationship between the two criteria more accurately.

Cut and *Cond* use ratio of between-edges over within-edges. *Cut* has a global computation of the ratio, whereas *Cond* not only computes the ratio at the cluster level but also considers only the minimum number of within-edges in each cluster. This difference between the two criteria has an important impact on the final results. Spearman's average correlation between *Cond* and *Cut* is 0.20. Most parallel coordinate diagrams show that there are crossings between *Cond* and *Cut* but not too many, confirming a partial relation between the criteria. It seems that *Cond* is preferable over *Cut* when comparing highly variable numbers clustering results. Note that *Cond* and *Cut* are the only two criteria that have to be minimized and not maximized. Therefore, Spearman's correlations have been computed with 1/*Cut* and 1/*Cond* instead of *Cut* and Cond. It is also the reason why we have reversed their axis in the parallel coordinate diagrams so that for all criteria best values are on top, worst values at the bottom.

C. What makes Perf and MQ different from Cov, Cut, Cond, Mod ?

The particularity of *MQ*, is that it explicitly accounts for the number of clusters. The number of clusters clearly impacts the number of possible between-edges and therefore the overall values of other criteria. When comparing clustering with very different numbers of clusters, *MQ* is very useful. Other criteria can exhibit severe bias. For example, in the extreme case where a clustering results in a single cluster, and is compared to a much better clustering that provides 10 clusters, no between-edge will be found in the first clustering and most criteria will compute a high quality measure despite the fact that the clustering is really doing a poor job compared to the second. The fact that *MQ* accounts for the number of clusters prevents it from that bias. Also, experiments showed no correlation at all with criteria such as *Cov* or mod and these results suggest that using *MQ* capture significantly different aspect of the quality of clustering and that using it in conjunction with *Cov* can be useful to balance other biases such as, for example, the bias coming from varying numbers of cluster.

Perf is probably the most debatable criteria amongst those reported in this paper. It surely tries to capture a different aspect of the quality of the results than the others. *Perf* captures the number of errors compared to an clustering that would ideally lead to a disconnected set of cliques. However, the fact that the computation of *Perf* computes a ratio of the number of errors (between edges and missing within edges) over the total number of possible edges can lead to very misleading interpretations in many real situations. For example, previous experiments showed that random clustering can get better *Perf* ratings than other clustering.
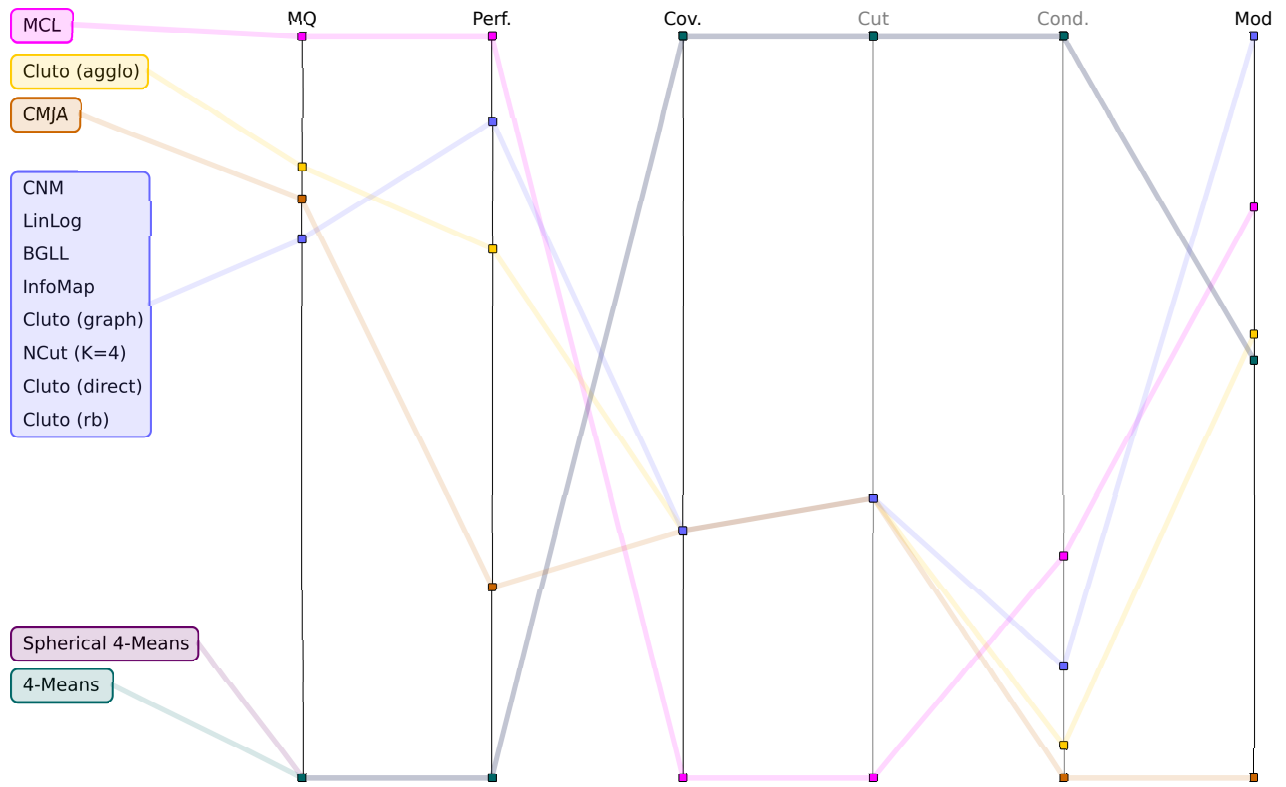
Figure 5. Usage of parrallel coordinate on quality criteria for JDM 20
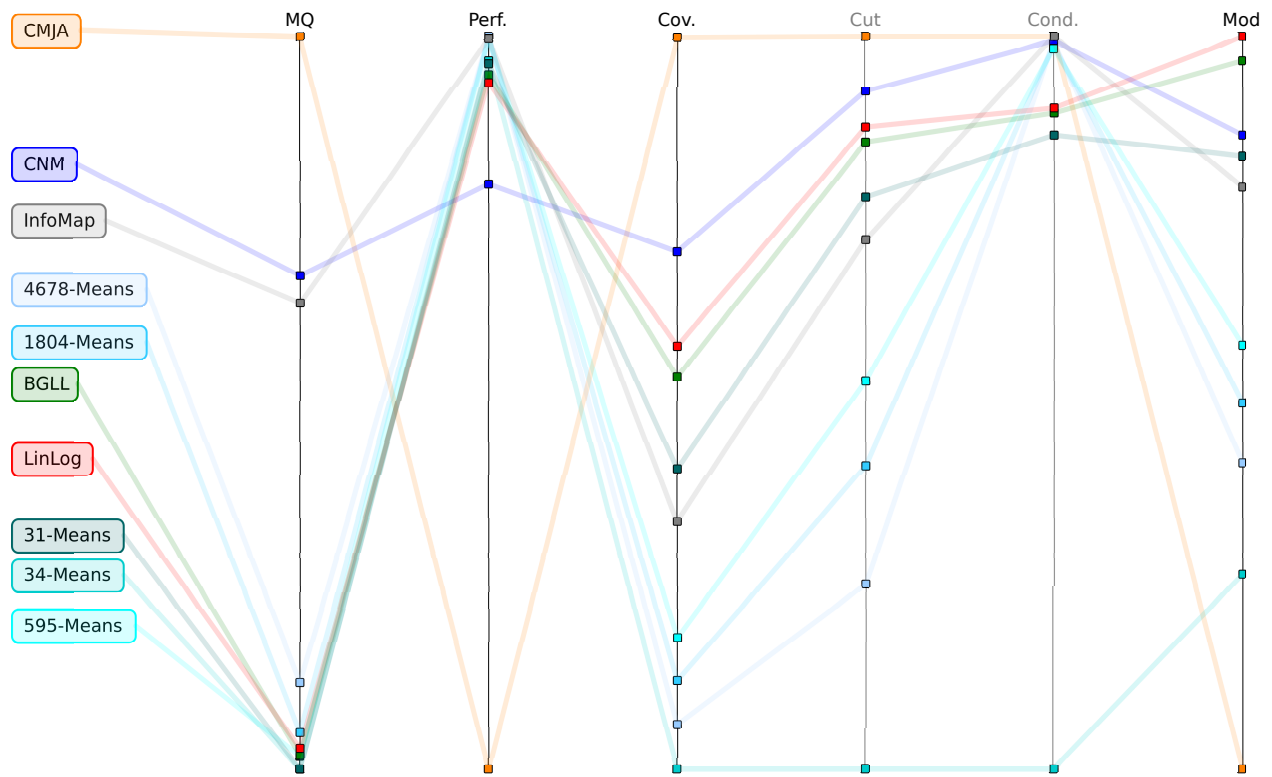

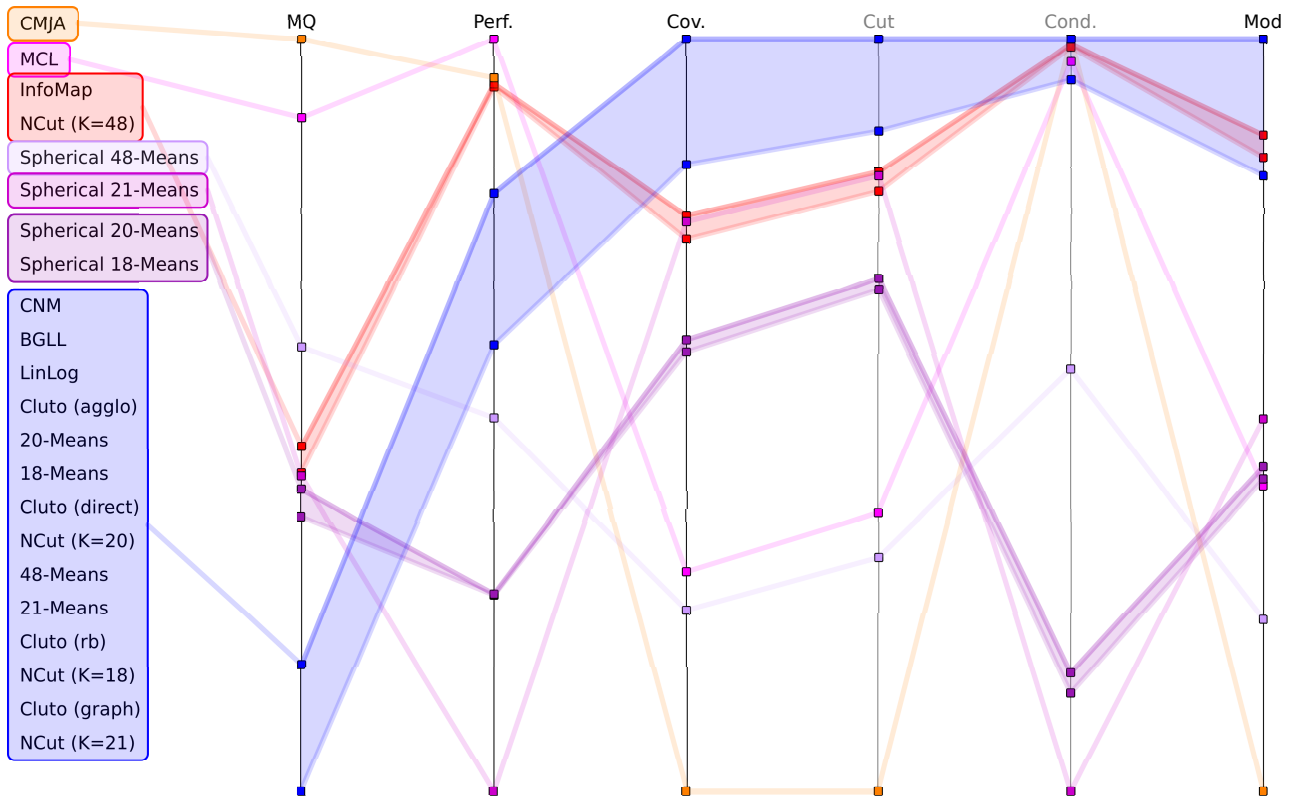Figure 6. Usage of parallel coordinate on quality criteria for JDM ALL

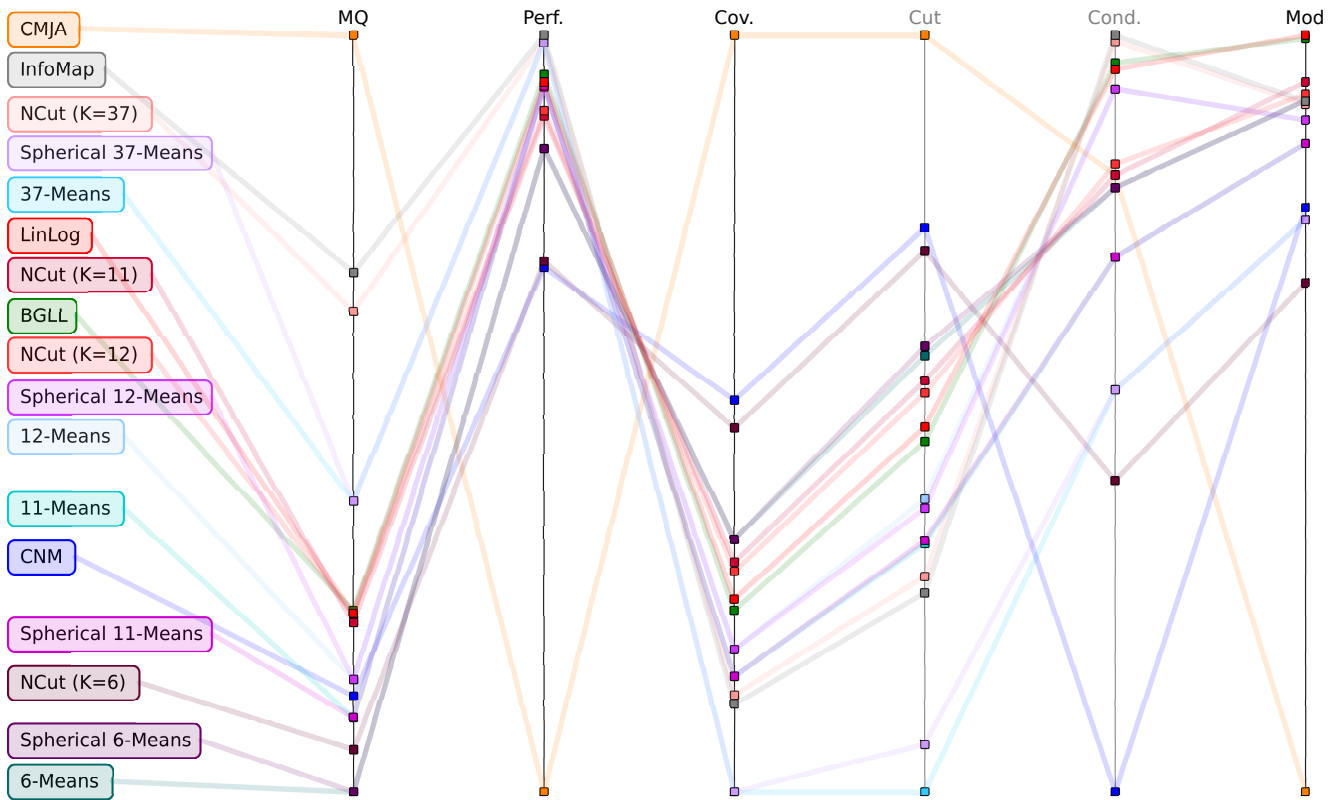Figure 8. Usage of Parrallel Coordinate Merged on Quality Criterion for SIG 1000

Figure 7. Usage of Parrallel Coordinate on Quality Criterion for SIG 10000

VII. Conclusion

In order to compare clustering results, most existing numerical criteria found in the literature focus on evaluating the quality of the compromise between intra-cluster cohesion and inter-cluster differentiation. In this paper we report the results of several experiments with clustering algorithms over different networks. We have combined different criteria and analyzed the results using different approaches. The lessons learned are : (1) there is a lot of variation in the quality of the same clustering technique depending on the criteria / the datasets / the parameters used in the algorithm and (2) out of six different quality criteria found in the literature, *Cov* and *MQ* used in conjunction can probably capture most of what the others can capture and (3) a lot of different aspects of the quality of the results cannot be captured at all with existing criteria. This experiment suggests that a lot of work is needed to better understand the quality and characteristics of automatic clustering results before using them to manage and control large and complex networks .

References

[1] James Abello, Frank van Ham, and Neeraj Krishnan. 2006. ASK-GraphView: A Large Scale Graph Visualization System. IEEE Transactions on Visualization and Computer Graphics 12, 5 (September 2006), 669-676.

[2] David Auber, Yves Chiricota, Fabien Jourdan, and Guy Melançon. 2003. Multiscale visualization of small world networks. In Proceedings of the Ninth annual IEEE conference on Information visualization (INFOVIS'03), Tamara Munzner and Stephen North (Eds.). IEEE Computer Society, Washington, DC, USA, 75-81.

[3] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008(10):P10008+, July 2008.

[4] Spiros Mancoridis, Brian S. Mitchell, C. Rorres, Yih-Farn Chen, Emden R. Gansner: Using Automatic Clustering to Produce High-Level System Organizations of Source Code. IWPC 1998.

[5] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. August 2004.

[6] Inderjit S. Dhillon and Dharmendra S. Modha. 2001. Concept decompositions for large sparse text data using clustering. Mach. Learn. 42, 1/2 (January 2001), 143-175.

[7] Stijn van Dongen, Graph Clustering by Flow Simulation. PhD thesis, University of Utrecht, May 2000.

[8] Huffman, D. A. "A Method for the Construction of Minimum-Redundancy Codes." Proc. Inst. Radio Eng. 40, 1098-1101, 1952.

[9] Alfred Inselberg and Bernard Dimsdale. 1990. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In Proceedings of the 1st conference on Visualization '90 (VIS '90), Arie Kaufman (Ed.). IEEE Computer Society Press, Los Alamitos, CA, USA, 361-378.

[10] Anil K. Jain. 2010. Data clustering: 50 years beyond K-means. *Pattern Recogn. Lett.* 31, 8 (June 2010), 651-666.

[11] Lafourcade, M.: Making People Play for Lexical Acquisition with the JeuxDeMots prototype. In: SNLP 2007: 7th International Symposium on Natural Language Processing, Pattaya, Thailand (2007).

[12] J. B. Macqueen. Some methods of classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967.

[13] Noack. A. 2004. An Energy Model for Visual Graph Clustering. In Proceedings of the 11th International Symposium on Graph Drawing (Perugia, Italy, Sep. 21--24), GD 2003, Springer-Verlag, Berlin, LNCS 2912, 425--436.

[14] Martin Rosvall and Carl T. Bergstrom. Maps of random walks on complex networks reveal community structure. Proceedings of the National Academy of Sciences, 105(4):1118-1123, January 2008.

[15] Gerard Salton and Chris Buckley. 1987. *Term Weighting Approaches in Automatic Text Retrieval*. Technical Report. Cornell University, Ithaca, NY, USA.

[16] Satu Elisa Schaeffer. Graph Clustering. Computer Science Review 1(1): 27-64, 2007.

[17] J. Seberry and M. Yamada, Hadamard matrices, sequences and block designs, Contemporary Design Theory:A Collection of Surveys , eds. J.Dinitz and D.Stinson, J.Wiley, New York, (1992), 431-560.

[18] Jianbo Shi and Jitendra Malik. 1997. Normalized Cuts and Image Segmentation. In Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97) (CVPR '97). IEEE Computer Society, Washington, DC, USA, 731-.

[19] Fei Wang, Changshui Zhang, and Tao Li. 2007. Regularized clustering for documents. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '07). ACM, New York, NY, USA, 95-102.

[20] Rui Xu and Donald C. Wunsch II, "Survey of clustering algorithms," IEEE Transactions on Neural Networks, vol. 16, no. 3, pp. 645-678, 2005.

[21] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis, 2001.

[22] Ulrik Brandes, Marco Gaertler, Dorothea Wagner: Experiments on Graph Clustering Algorithms. ESA 2003: 568-579

[23] Boutin, F.: Filtrage, partitionnement et visualisation multi-échelles de graphes d'interactions à partir d'un focus. Phd. Thesis (2005)