



HAL
open science

Clust&See: A Cytoscape plugin for the identification, visualization and manipulation of network clusters

Lionel Spinelli, Philippe Gambette, Charles E. Chapple, Benoît Robisson, Anaïs Baudot, Henri Garreta, Laurent Tichit, Alain Guénoche, Christine Brun

► **To cite this version:**

Lionel Spinelli, Philippe Gambette, Charles E. Chapple, Benoît Robisson, Anaïs Baudot, et al.. Clust&See: A Cytoscape plugin for the identification, visualization and manipulation of network clusters. *BioSystems*, 2013, - (-), 10.1016/j.biosystems.2013.05.010 . hal-00832028v1

HAL Id: hal-00832028

<https://hal.science/hal-00832028v1>

Submitted on 10 Jun 2013 (v1), last revised 30 Jun 2013 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Clust&See: a Cytoscape plugin for the identification, visualization and manipulation of network clusters

Lionel Spinelli ^{a, d}, Philippe Gambette ^{a, d, 1}, Charles E. Chapple ^{b, d}, Benoît Robisson ^{b, d}, Anaïs Baudot ^{a, d}, Henri Garreta ^{c, d}, Laurent Tichit ^{a, d}, Alain Guénoche ^{a, d}, Christine Brun ^{b, d, e}

^a Institut de Mathématiques de Luminy, CNRS - FRE3529, Avenue de Luminy, 13288 Marseille Cedex 9, France

^b Techniques Avancées pour le Génome et Clinique, INSERM - U1090, Avenue de Luminy, 13288 Marseille Cedex 9, France

^c Laboratoire d'Informatique Fondamentale de Marseille, CNRS – UMR7279, Avenue de Luminy, 13288 Marseille Cedex 9, France

^d Aix-Marseille Université, Campus de Luminy, 13288 Marseille Cedex 9, France

^e CNRS, France

¹ Current address : Laboratoire d'Informatique Gaspard-Monge, CNRS - UMR 8049, Cité Descartes, Bât Copernic --5, bd Descartes, Champs sur Marne 77454 Marne-la-Vallée Cedex 2 France

Corresponding author:

Christine Brun
TAGC, U1090 Inserm-AMU
Parc Scientifique de Luminy case 928
163, avenue de Luminy
13288 MARSEILLE cedex 09
FRANCE

Email : brun@tagc.univ-mrs.fr
Tel : +33491828712

ABSTRACT

Background and Scope

Large networks, such as protein interaction networks, are extremely difficult to analyze as a whole. We developed Clust&See, a Cytoscape plugin dedicated to the identification, visualization and analysis of clusters extracted from such networks.

Implementation and performance

Clust&See provides the ability to apply three different, recently developed graph clustering algorithms to networks and to visualize (i) the obtained partition as a quotient graph in which nodes correspond to clusters, (ii) the obtained clusters as their corresponding subnetworks. Importantly, tools for investigating the relationships between clusters and vertices as well as their organization within the whole graph are supplied.

Availability

<http://tagc.univ-mrs.fr/tagc/index.php/clustnsee>

Keywords: interaction networks, graph partitioning, clustering, visualization

1. INTRODUCTION

The field of functional genomics is producing a large amount of data, often represented as interaction networks - or undirected graphs. These graphs typically contain thousands of vertices, rendering the extraction of pertinent biological information a daunting task. Graph partitioning or clustering methods have been used to highlight groups of densely connected vertices (Aittokallio and Schwikowski, 2006) which, in the field of protein interactions, often correspond to clusters of proteins involved in the same cellular process(es).

Cytoscape is a popular and versatile software platform (Shannon et al., 2003) for network visualization and analysis. While a number of Cytoscape plugins such as ClusterMaker (Morris et al., 2011) or ClusterOne (Nepusz et al., 2012) can identify clusters from graphs, they mainly focus on visualizing the obtained clusters individually and independently as subnetworks to further investigate their node composition. However, exploring the relationships between clusters in detail is as important as studying their internal composition. Indeed, while proteins involved in the same process(es) interact within clusters, links between clusters correspond to crosstalk between processes. Communication between processes can also be performed by proteins belonging to several clusters. Consequently, considering the links between network clusters permits a better understanding of the modularity of biological networks and the functional transitions imposed by the integrative organization levels, from proteins to functional modules to entire systems.

To fill this gap, we have developed Clust&See, a truly interactive tool that can (i) automatically decompose a network into clusters, (ii) visualize those clusters as metanodes linked by several types of edges/relationships, (iii) manipulate the clusters for further detailed visualization, analyses and comparisons.

2. SOFTWARE DESCRIPTION

Clust&See is a Cytoscape plugin developed for Cytoscape version 2.8. Some GUI elements have been reused from code from the MCODE (Bader and Hogue, 2003) and ClusterViz (<http://apps.cytoscape.org/apps/clusterviz>) plugins.

2.1. Implemented graph clustering algorithms

To date, three clustering algorithms based on the optimization of Newman's modularity (Newman, 2004) have been implemented in Clust&See. While TFit and FT lead to disjoint clusters, OCG leads to overlapping ones:

(1) FT (for Fusion-Transfer) (Guénoche, 2011) is an ascending hierarchical method fusing two clusters iteratively if the fusion results in a modularity gain. The algorithm starts with singletons and stops when further fusions lead to a loss in modularity. Modularity is then further optimized by transferring vertices from one cluster to another.

(2) TFit (for iterated Transfer-Fusion) (Gambette and Guénoche, 2012) is a multi-level algorithm in which a vertex transfer procedure is performed at every level. Level one corresponds to the network. While modularity increases, each node is associated to its best adjacent cluster. Classical transfers are then performed and a quotient graph is computed; clusters then become the nodes of the next level to be further optimized.

(3) OCG (for Overlapping Cluster Generator) (Becker et al., 2012) is an ascending hierarchical method fusing two clusters at each step. Initially, an overlapping class system formed by either (i) maximal cliques, or (ii) edges or (iii) centered cliques is built. These classes are then merged, while modularity increases, resulting in overlapping clusters.

Performance values in terms of time and memory of the three algorithms are provided as Supplementary Material. Clustering results produced by Clust&See can be exported as text files, and subsequently re-imported and re-mapped to the original network, avoiding repetitive computation. Importantly, results from external clustering tools can be analyzed with Clust&See. Currently, the R package "Linkcomm" (Kalinka and Tomancak, 2011), in which the LinkCommunities (Ahn et al., 2010) and OCG (Becker et al., 2012) algorithms are implemented, provides output files that are compatible with Clust&See (for further information on supported formats, see the online Documentation, <http://tagc.univ-mrs.fr/tagc/index.php/software/clustnsee/clustnseedocumentation>). Finally, the modular structure of Clust&See makes it easy to implement other clustering algorithms directly in Java.

2.2. Visualization and analysis

The clustering results can be visualized in Clust&See as a quotient graph in which clusters are represented as metanodes whose width is proportional to the number of their constituent vertices (Figure 1). Metanodes can be linked by two types of "metaedges", one (black) whose width is proportional to the number of interactions between their vertices and, most importantly, one (green) whose width is proportional to the number of vertices shared by overlapping clusters computed by algorithms such as OCG. A docked Result Panel provides a sortable list of the clusters in which each cluster's subnetwork is displayed along with its relevant features, such as size or edge density.

Novel views, into which clusters of interest can be successively loaded, can be created on demand. An "Expand/collapse nodes" function allows the user to switch from the cluster/metanode representation to the corresponding subnetwork of vertices and vice versa (Figure 2). Details provided in the Data Panel upon selection of the different objects (vertices,

edges) facilitate the study of the relationships between clusters. The composition of each metanode is provided as well as, importantly, the composition of the metaedges representing the shared objects (nodes or edges) between cluster pairs.

When two different partitions are computed on the same network (using different algorithms or different parameters), they are compared using the Jaccard index (Jaccard, 1901) which provides a measure of the partitions' similarity. A contingency table listing the number of shared nodes between the clusters of each partition is provided for further analysis.

In addition, because launching an analysis on a very large network or selection may lead, at best, unmanageable results (too many clusters) and, at worst, to memory issues and very long computation times, Clust&See offers the user the choice of extracting sub-networks of interest on which to continue the analysis by using the “Build neighborhood network” functionality. Finally, the provided search function can identify a specific node among the clusters of all partitions under investigation.

3. APPLICATION

3.1. From clusters to nodes

Figure 1 shows the results obtained when applying the 3 algorithms currently implemented in Clust&See to the PI3K interactome network (Pilot-Storck et al., 2010). A global view of each partition as a quotient graph, in which the obtained clusters are represented as metanodes, is given. Note that the default view is shown when the partition contains no more than 15 clusters/metanodes, but can always be displayed on demand for larger partitions. These views are one of the original features provided by the plug-in.

The PI3K pathway transmits signals from receptors located at the cell surface to transcription factors in the nucleus, *via* an intracellular signaling cascade involving several kinases. To illustrate the value of a local analysis using Clust&See, we have chosen to explore the connections between 3 overlapping clusters generated by the OCG algorithm, containing a majority of receptor-binding proteins (Cluster 12), serine/threonine kinases (Cluster 25) and nuclear acid binding proteins (Cluster 5) respectively. The 3 clusters are represented as metanodes in Figure 2, which shows a “New Cluster View” created on the fly. In addition, Cluster 25, formed by 13 nodes linked by 16 intra-cluster edges, is shown in the Cluster Browser of the Results panel. Details on the mono/multi-clustered status of vertices are given in the Data Panel. Clusters sharing vertices (like Clusters 5 and 25), are easily identifiable since they are linked by a green metaedge whose details are shown in the Data Panel upon selection: two kinases, KS6B1 and PK3CA, belong to both clusters, suggesting a possible functional link between those proteins. Interestingly, particular variants of the genes encoding these proteins have been found to interact genetically in a case-control study for colorectal cancers (Slattery et al., 2011). A visual exploration of the organization of the clusters with Clust&See can therefore help building hypothesis and pointing toward relevant functional objects (represented as nodes, metanodes, edges or metaedges) and their relationships. Finally, metanodes can be expanded (and subsequently collapsed) in order to visualize the underlying subnetwork. The combination of the details shown in the Data Panel and the visualization of the composite subnetwork greatly facilitates the study of the identified clusters and their connections.

3.2. From nodes to clusters

Figure 3 illustrates the search for a particular node, PARP1, across the different partitions obtained when using all three algorithms on the same network. PARP1 belongs to two OCG clusters (P3 partition in the Results Panel). Interestingly, when both clusters are loaded in a “New Cluster View” for further analysis, the Data Panel shows that PDK1 is also shared by the same clusters, suggesting a possible functional link between PARP1 and PDK1. This is further confirmed when the expanded cluster view is generated, showing that both proteins interact directly. This interaction could in part explain the fact that co-targeting the PI3K pathway improves the response of cancer cells to PARP1 inhibition (Kimbung et al., 2012).

4. CONCLUSION

The Cytoscape plug-in Clust&See aims to facilitate network clustering and analysis for biologists not only by providing several original functionalities but also by providing them within a single analysis framework. While ClusterMaker (Morris et al., 2011) can represent clusters as metanodes, it provides neither intra/extra-edge visualization nor the possibility to expand/collapse the metanodes. Similarly, while ClusterOne (Nepusz et al., 2012) can identify overlapping clusters, studying the relationships between these clusters is not possible because their combined representation is not supported. In addition, Clust&See enables (i) better evaluation of the biological meaning of network clustering, (ii) better understanding of the underlying reasons for a particular node classification, (iii) better estimation of the quality of the network under scrutiny and (iv) adjusting the clustering algorithm choice to the studied network. In summary, the originality of Clust&See lies in its providing users with a complete tool for the creation and analysis of network clusters and the relationships between them.

ACKNOWLEDGEMENTS

We thank Marine Veyssi re for the update of the PI3K interactome. This work is supported by the “Agence Nationale de la Recherche” as a PIRIBIO grant (09-PIRI-0028, Moonlight project to C.B). B.R. is a PhD fellow of the AXA Research Fund. C.C. is a postdoctoral fellow of the “Fondation pour la Recherche M dicale”.

REFERENCES

- Ahn, Y.-Y., Bagrow, J.P., and Lehmann, S. (2010). Link communities reveal multiscale complexity in networks. *Nature* 466, 761–764.
- Aittokallio, T., and Schwikowski, B. (2006). Graph-based methods for analysing networks in cell biology. *Brief Bioinform* 7, 243–255.
- Bader, G.D., and Hogue, C.W. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* 4, 2.
- Becker, E., Robisson, B., Chapple, C.E., Gu noche, A., and Brun, C. (2012). Multifunctional proteins revealed by overlapping clustering in protein interaction network. *Bioinformatics* 28, 84–90.
- Gambette, P., and Gu noche, A. (2012). Bootstrap clustering for graph partitioning. *RAIRO - Operations Research* 45, 339–352.

Guénoche, A. (2011). Consensus of partitions: a constructive approach. *Advances in Data Analysis and Classification* 5, 215–229.

Jaccard, P. (1901). Distribution de la flore alpine dans le bassin des Dranses et de quelques régions voisines. *Bulletin de la Société Vaudoise de Sciences Naturelles* 37, 241–272.

Kalinka, A.T., and Tomancak, P. (2011). linkcomm: an R package for the generation, visualization, and analysis of link communities in networks of arbitrary size and type. *Bioinformatics* 27, 2011–2012.

Kimbung, S., Biskup, E., Johansson, I., Aaltonen, K., Ottosson-Wadlund, A., Gruvberger-Saal, S., Cunliffe, H., Fadeel, B., Loman, N., Berglund, P., et al. (2012). Co-targeting of the PI3K pathway improves the response of BRCA1 deficient breast cancer cells to PARP1 inhibition. *Cancer Lett.* 319, 232–241.

Morris, J.H., Apeltsin, L., Newman, A.M., Baumbach, J., Wittkop, T., Su, G., Bader, G.D., and Ferrin, T.E. (2011). clusterMaker: a multi-algorithm clustering plugin for Cytoscape. *BMC Bioinformatics* 12, 436.

Nepusz, T., Yu, H., and Paccanaro, A. (2012). Detecting overlapping protein complexes in protein-protein interaction networks. *Nat. Methods* 9, 471–472.

Newman, M.E. (2004). Fast algorithm for detecting community structure in networks. *Phys Rev E Stat Nonlin Soft Matter Phys* 69, 066133.

Pilot-Storck, F., Chopin, E., Rual, J.-F., Baudot, A., Dobrokhotov, P., Robinson-Rechavi, M., Brun, C., Cusick, M.E., Hill, D.E., Schaeffer, L., et al. (2010). Interactome mapping of the phosphatidylinositol 3-kinase-mammalian target of rapamycin pathway identifies deformed epidermal autoregulatory factor-1 as a new glycogen synthase kinase-3 interactor. *Mol. Cell Proteomics* 9, 1578–1593.

Shannon, P., Markiel, A., Ozier, O., Baliga, N.S., Wang, J.T., Ramage, D., Amin, N., Schwikowski, B., and Ideker, T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* 13, 2498–2504.

Slattery, M.L., Lundgreen, A., Herrick, J.S., and Wolff, R.K. (2011). Genetic variation in RPS6KA1, RPS6KA2, RPS6KB1, RPS6KB2, and PDK1 and risk of colon or rectal cancer. *Mutat. Res.* 706, 13–20.

FIGURE LEGENDS

Figure 1: Quotient graphs obtained when clustering the PI3K network with the 3 implemented algorithms.

Figure 2: A 'New Cluster View' showing the clusters 5, 12 and 25 generated by the OCG algorithm as metanodes and the two edge types linking them. Black metaedges connect vertices of one cluster to those of another. Green metaedges represent the nodes shared between the clusters. Details are shown in the Data Panel when selecting a green or a black metagedge. An example of an expanded metanode is also given.

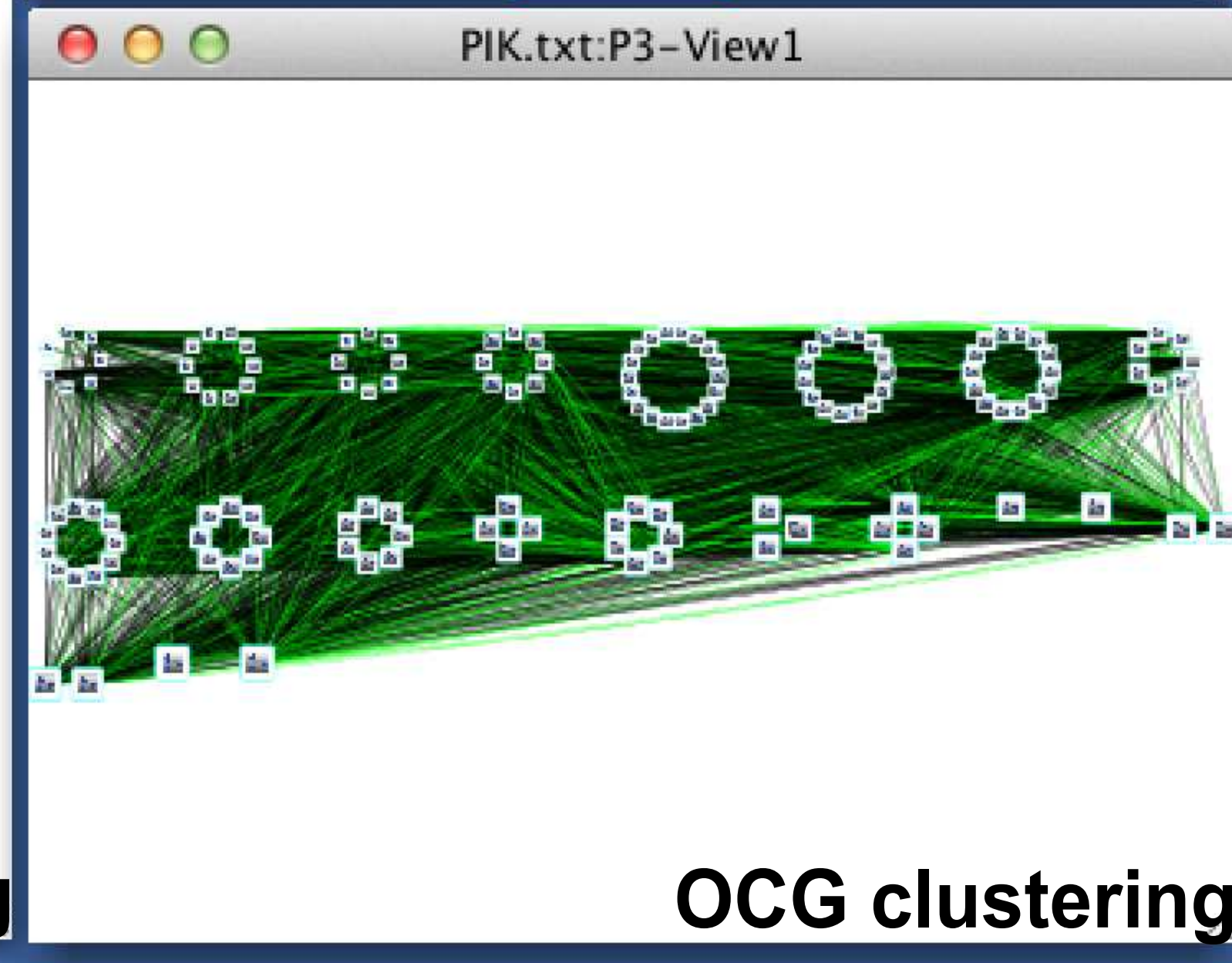
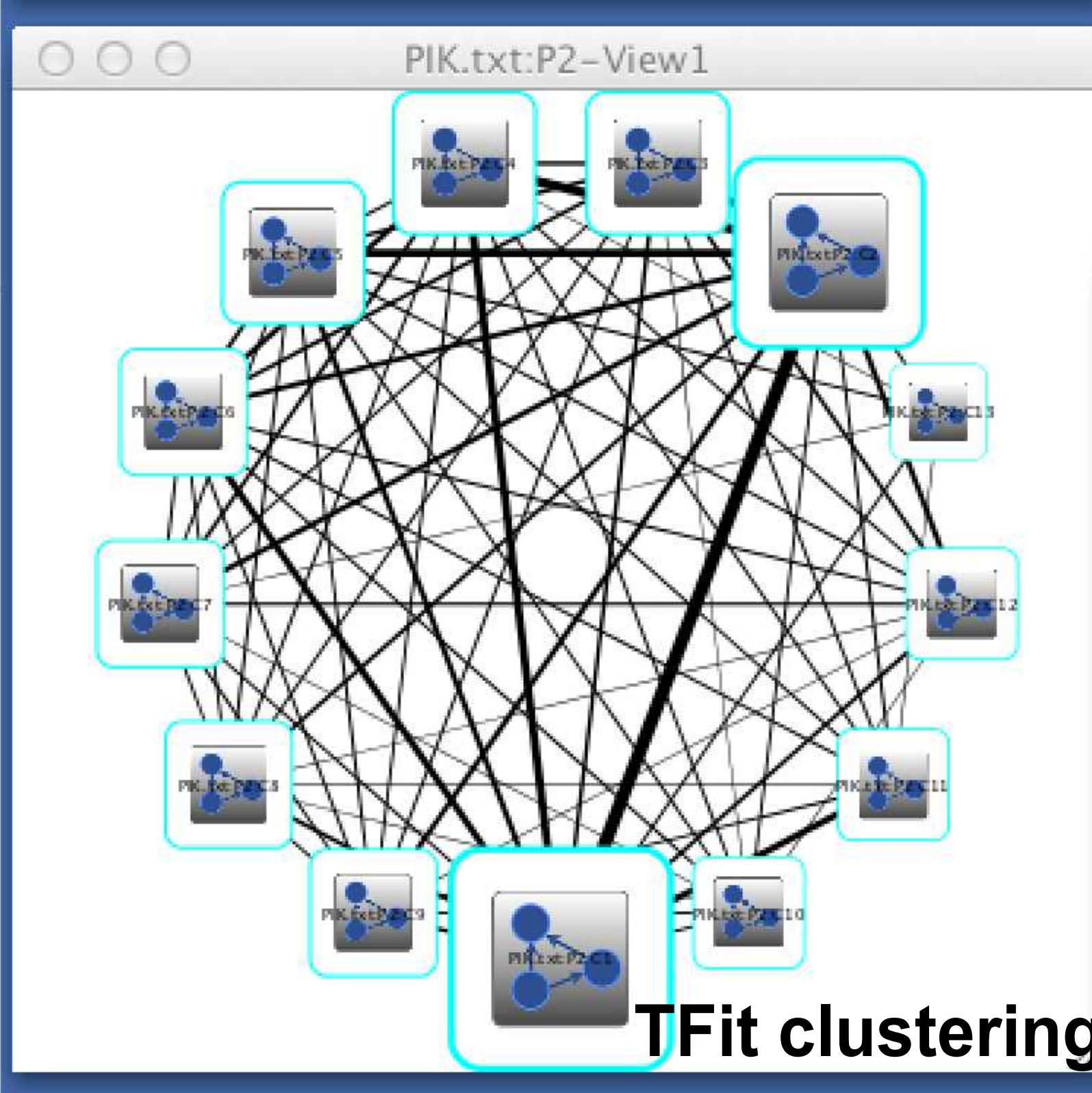
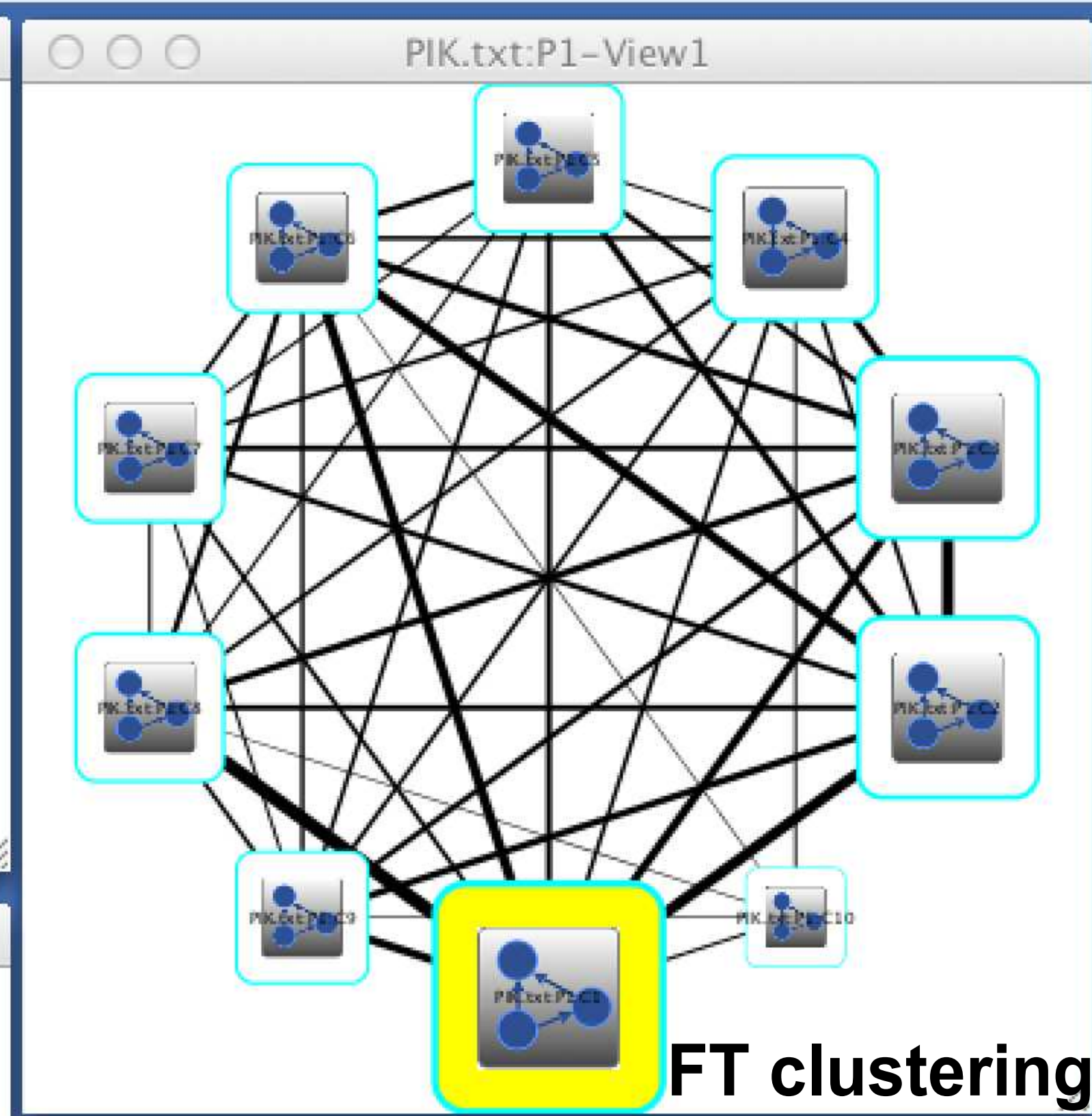
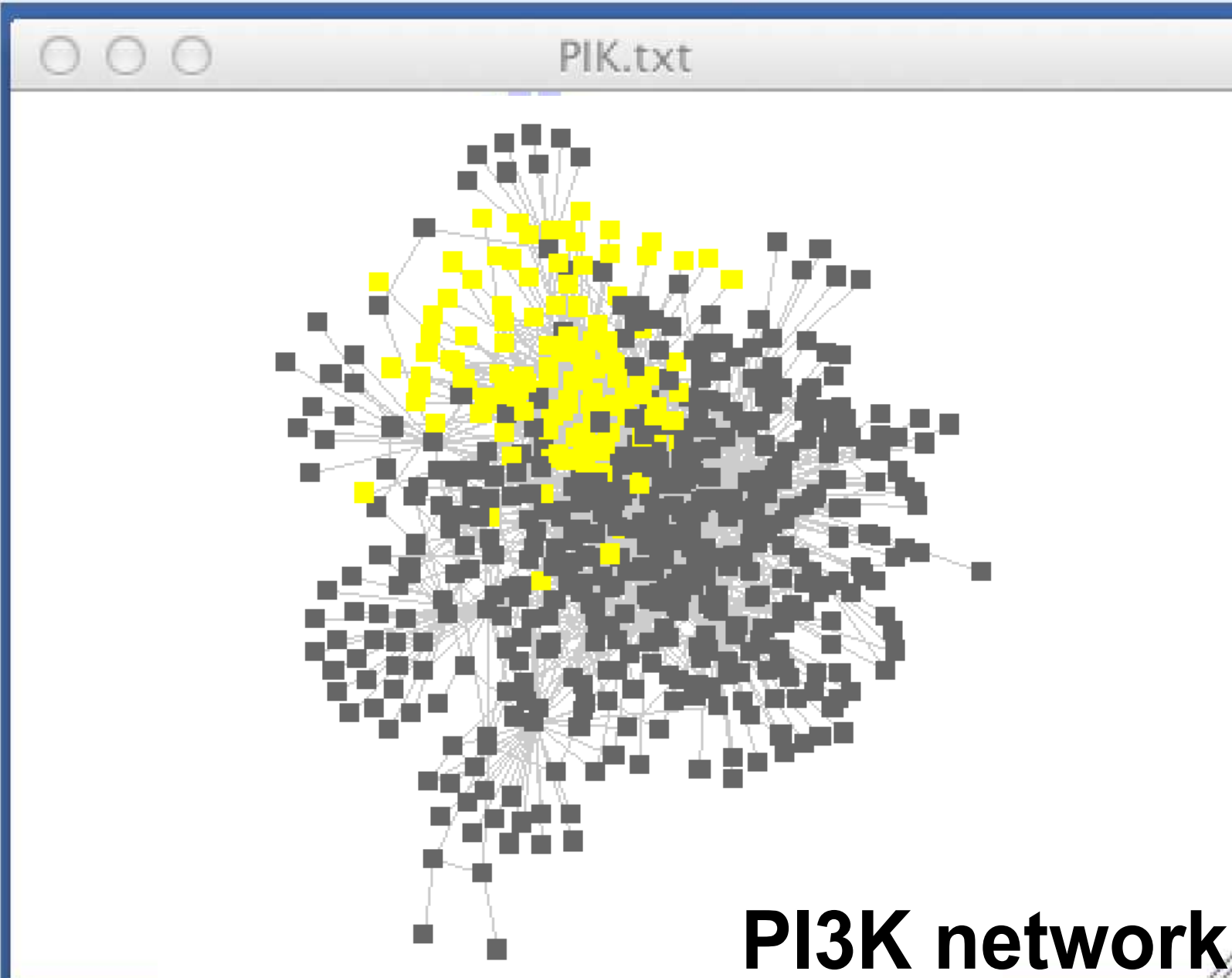
Figure 3: The result of a search for PARP1 in the clusters obtained from the different

algorithms is shown. The two clusters containing PARP1 according to OCG are provided as a 'New Cluster View' and the expanded metanodes are shown below them.



Control Panel Clust&See

- Scope
- Network
 - Selection
- Algorithm
- FT
 - TFit
 - OCG
- OCG Options
- Default options
 - Expert options



Results Panel

PIK.txt:P1 | PIK.txt:P2 | PIK.txt:P3

Results | Information

Cluster Browser(10 in total)

Sort Clusters by (descend): Size in/ext edges ratio

Snapshot	Details
	Name : PIK.txt:P1:C1 Nodes : 122 Intra Cluster Edges : 458 Extra Cluster Edges : 250 Intra/Extra edges ratio : 1,832
	Name : PIK.txt:P1:C2 Nodes : 79 Intra Cluster Edges : 133 Extra Cluster Edges : 224 Intra/Extra edges ratio : 0,594
	Name : PIK.txt:P1:C3 Nodes : 72 Intra Cluster Edges : 156 Extra Cluster Edges : 198 Intra/Extra edges ratio : 0,788
	Name : PIK.txt:P1:C4 Nodes : 58

Current: PIK.txt:P1:C1

New Cluster View | New Cluster Network View

Add Cluster to View | Add Cluster Network to View

Partition View | Export Partition | Discard Partition

Analyze

Data Panel

PIK.txt:P1:C1

Nodes in cluster :122

Node List | Annotation List

Node Attribute Browser | Edge Attribute Browser | Network Attribute Browser | Partition Details

Control Panel

Clust&See

Scope

- Network
- Selection

Algorithm

- FT
- TFit
- OCG

OCG Options

- Default options
- Expert options

Analyze

PIK.txt:P3-View2

New Cluster View

PIK.txt:P3-View3

When a cluster is expanded :

Results Panel

PIK.txt:P1 | PIK.txt:P2 | PIK.txt:P3

Results | Information

Cluster Browser (139 in total)

Sort Clusters by (descend): Size

Snapshot | Details

Name : PIK.txt:P3:C25
Nodes : 13
Intra Cluster Edges : 16
Extra Cluster Edges : 102
Intra/Extra edges ratio : 0,157

Name : PIK.txt:P3:C26

Current: PIK.txt:P3:C27

New Cluster View | New Cluster Network View

Add Cluster to View | Add Cluster Network to View

Partition View | Export Partition

Data Panel

When a green edge is selected :

PIK.txt:P3:C5 -- PIK.txt:P3:C25

PIK.txt:P3:C5

Nodes in cluster :20

Mono-clustered node... 20

Multi-clustered nodes :10

Intra cluster edges :29

Extra cluster edges :207

Node name	# clusters	Degree
4EBP1	1	6
BDP1	1	1
CLIP1	1	1
CSK21	15	54
DNJB6	1	1
FOS	3	8
HCL1	1	1

Multiclassed nodes : 2

Node name
KS6B1
PK3CA

PIK.txt:P3:C25

Nodes in cluster :13

Mono-clustered node... 13

Multi-clustered nodes :7

Intra cluster edges :16

Extra cluster edges :102

Node name	# clusters	Degree
AAKB1	2	4
AAKG1	2	4
AAPK1	14	36
AAPK2	1	5
APC7	1	1
BRSK2	1	1
CASI	2	7

Node Attribute Browser | Edge Attribute Browser | Network Attribute Browser | Partition Details

Data Panel

When a black edge is selected :

PIK.txt:P3:C12 -- PIK.txt:P3:C5

PIK.txt:P3:C12

Nodes in cluster :17

Mono-clustered node... 17

Multi-clustered nodes :10

Intra cluster edges :33

Extra cluster edges :241

Node name	# clusters	Degree
FABP9	1	1
FYN	8	28
GAB2	2	10
IBP2	1	1
IBP4	1	1
IBP5	1	1
IGF1	8	14

Number of Interactions : 4

Node name	Node name
GAB2	MK01
FYN	KPCZ
LCK	PK3CA
P85A	PK3CA

PIK.txt:P3:C5

Nodes in cluster :20

Mono-clustered node... 20

Multi-clustered nodes :10

Intra cluster edges :29

Extra cluster edges :207

Node name	# clusters	Degree
4EBP1	1	6
BDP1	1	1
CLIP1	1	1
CSK21	15	54
DNJB6	1	1
FOS	3	8
HCL1	1	1

Node Attribute Browser | Edge Attribute Browser | Network Attribute Browser | Partition Details



Control Panel

Clust&See

Scope

- Network
- Selection

Algorithm

- FT
- TFit
- OCG

OCG Options

- Default options
- Expert options

Analyze

Search Node Clusters

Node information

Node name :

Node clusters

Cluster name

- PIK.txt:P1:C5
- PIK.txt:P2:C5
- PIK.txt:P3:C11
- PIK.txt:P3:C22

OCG Result

PIK.txt:P3-View4

New Cluster View

PIK.txt:P3-View5

Results Panel

PIK.txt:P1 | PIK.txt:P2 | PIK.txt:P3

Results | Information

Cluster Browser (139 in total)

Sort Clusters by (descend): Size

Snapshot	Details
	Name : PIK.txt:P3:C21 Nodes : 14 Intra Cluster Edges : 18 Extra Cluster Edges : 207 Intra/Extra edges ratio : 0,087
	Name : PIK.txt:P3:C22 Nodes : 14 Intra Cluster Edges : 20 Extra Cluster Edges : 115 Intra/Extra edges ratio : 0,174

Current: PIK.txt:P3:C22

PIK.txt:P3:C22 -- PIK.txt:P3:C11

PIK.txt:P3:C22

Nodes in cluster :14

Mono-clustered node... 14

Multi-clustered nodes :11

Intra cluster edges :20

Node name	# clusters	Degree
IRK1	1	4
KAPCA	5	21
M3K3	2	7
NHRF2	3	6
PAF1	1	1

Multiclassed nodes : 2

Node name
PARP1
PDPK1

PIK.txt:P3:C11

Nodes in cluster :17

Mono-clustered node... 17

Multi-clustered nodes :11

Intra cluster edges :29

Node name	# clusters	Degree
AKT1	25	109
CASP3	10	29
DLG1	1	1
EP300	3	21
GATA2	1	1

Supplementary information

1- Three different algorithms are currently implemented in Clust&See. They are all based on the optimisation of Newman's modularity (Newman, 2004).

- FT (for Fusion-Transfer) (Guénoche, 2011) is an ascending hierarchical method fusing two clusters at each step if the fusion results in a modularity gain. The algorithm stops when further fusions lead to a loss in modularity. Finally, the modularity is further optimized by transfer of vertices from one cluster to another.

- TFit (for Transfer-Fusion iterated) (Gambette and Guénoche, 2011) is a multi-level method of transfers. Level one corresponds to the network. While modularity increases, each node is associated to its best adjacent cluster. Classical transfers are then performed and a quotient graph, in which clusters become the nodes of the next level, is computed.

- OCG (for Overlapping Cluster Generator) (Becker *et al*, 2012) is an ascending hierarchical method fusing two clusters at each step. The initial clusters make an overlapping class system that can be either (i) maximal cliques, (ii) edges or (iii) centered cliques. These are then merged, while modularity increases, resulting in overlapping clusters.

Algorithms	Initial classes	Stop criterion	Results	Speed	Network Size*
FT (Guénoche, 2011)	singletons (N^\dagger)	maximum modularity value and local transfer optimization	disjoint clusters	slow	$N < 8\ 000$
TFit (Gambette & Guénoche, 2011)	singletons (N)	maximum modularity value	disjoint clusters	fast	$N < 30\ 000$
OCG (Becker <i>et al.</i> , 2012)	max. cliques ($\gg N$) or edges ($< N(N-1/2)$) or centered cliques ($< N$)	# of class or max. class cardinality or max. modularity value	overlapping clusters and list of multi-clustered nodes	very slow slow fairly slow	$N < 10\ 000$

* These limits are just recommendations. Exceeding these limits when using the Clust&See plugin may result in very long computation times and high memory usage. Therefore, we recommend a native implementation of the clustering algorithm for larger networks.

\dagger N is the number of nodes in the network.

2- Performance of the three algorithms when applied to different subnetworks of the human interactome (tests performed on an Intel core i7 cpu M640, 2.8GHz, 4GB RAM).

Network name	# nodes	# edges
1	1135	2010
2	1902	4148
3	3906	10430
4	7051	20883

time

	1	2	3	4
FT	0'06"	0'25"	4'25"	25'
TFit	0'01"	0'04"	0'10"	0'30"
OCG*	0'19"	1'47"	17'32"	120'

memory use (estimate)

	1	2	3	4
FT	22M	55M	100M	800M
TFit	20M	50M	270M	920M
OCG*	17M	77M	286M	612M

* The centered clique system was used for OCG.

3- OCG algorithm performance according to implementation (in C as in Becker et al., 2012; in Java as in Clust&See (this work); in R as in the LinkComm package (Kalinka and Tomancak, 2011)), when applied to different subnetworks of the human interactome (tests performed on an Intel core i7 cpu M640, 2.8GHz, 4GB RAM).

time

	1	2	3	4
C	0'14"	1'07"	11'35"	83'06"
Java	0'19"	1'47"	17'32"	120'
R	0'05"	0'26"	5'50"	40'02"

memory (estimations)

	1	2	3	4
C	23M	65M	220M	640M
Java	17M	77M	286M	612M
R	15M	50M	100M	350M

4- References :

Becker, E., Robisson, B., Chapple, C.E., Guénoche, A. Brun, C. (2012) Multifunctional proteins revealed by overlapping clustering in protein interaction network. *Bioinformatics*, 28(1), 84-90.

Gambette, P. and Guénoche, A. (2011) Bootstrap Clustering for Graph Partitioning. *RAIRO-Operations Research*, 45(4), 339-352.

Guénoche, A. (2011) Consensus of partitions : a constructive approach. *Advances in Data Analysis and Classification*, 5(3), 215-229.

Kalinka, A.T., Tomancak, P. (2011) Linkcomm: an R package for the generation, visualization, and analysis of link communities in networks of arbitrary size and type. *Bioinformatics*, 27(14), 2011-2012.

Newman, M.E. (2004) Fast algorithm for detecting community structure in networks. *Phys. Rev. E Stat. Nonlin. Soft. Matter Phys.*, 69, 066133.