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[†])	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 et al., 2012)	max. cliques (>>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.

[†] 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 at 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.