Supplementary Information

PEPPER: Cytoscape app for Protein complex Expansion using Protein-Protein intERaction networks

C. Winterhalter, R. Nicolle, A. Louis, C. To, F. Radvanyi, M. Elati

1 METHODS

1.1 Problem Formulation

Given an unweighted and undirected graph $G = (V, E)$ (PPI network) with V the set of graph vertices (*i.e* proteins), $E \subseteq V \times V$ the set of edges (*i.e* protein interactions) and a list of interesting vertices P (*seed* list of proteins), the problem to be solved is to identify subgraphs $G' = (V', E')$ that are densely connected and include proteins in P.

Graph based objective. The first objective is formulated as a maximisation of the subgraph density $f_{density}(G')$. Computed as the ratio between the number of interactions found between proteins of the subgraph over the total of all possible interactions, it is defined as follows:

$$f_{density}(G') = \frac{2|E'|}{|V'|(|V'| - 1)}$$

where V' is the set of proteins in a given solution G' and E' is a subset of E containing only interactions between proteins in V'.

Seed list based objective. The second objective seeks to include as many *seed* vertices (proteins of interest) of P as possible in G' and is referred to as the coverage:

$$f_{coverage}(G') = \frac{|P \bigcap V'|}{|P|}$$

which is maximised whenever all proteins of the *seed* $|P|$ are chosen.

Both $f_{density}(G')$ and $f_{coverage}(G')$ functions are ranged in $[0, 1]$. Adding a large number of irrelevant proteins, up to an extreme solution in which all proteins are chosen ($V' = V$), will not degrade the coverage function. However, in practice increasing the number of irrelevant proteins in V' rapidly degrades the density ($f_{density}$) of the solution G'. In the same manner, small solutions may have high density (*e.g.* local cliques) but will rarely include many *seed* proteins from P.

1.2 Multi-objective optimisation for relevant subgraph extraction

Finding dense subgraphs is an NP-hard task (Feige *et al.*, 2001). Optimisation of multiple objectives makes the problem become even more intractable. To address the problem of extracting subgraphs satisfying the density and coverage criteria, PEPPER uses a Multi-Objective Genetic Algorithm (MOGA) approach to extract a set of solutions maximising both objective functions. MOGA belong to a family of meta-heuristic optimisation algorithms that mimic biological evolution and natural selection to evolve candidate solutions then determine the *fittest* individual - representing a solution - relatively to defined fitness functions. The Strength Pareto Evolutionary Algorithm 2 (*SPEA2*, Zitzler *et al.* (2001)) was used to optimise simultaneously the Graph- and *seed* list-based objective functions. The MOGA components and operators are described in the following subsections.

Solutions representation and fitness function. Given a PPI network $G = (V, E)$, a candidate solution is encoded into a binary chromosome of size $|V|$ representing the indexed set of V. A 1 value at position i corresponds to the presence of the ith protein. PEPPER uses the *SPEA2* implemented in the open-source *JMetal* platform (Durillo and Nebro, 2011). Based on the notion of non-dominance for fitness evaluation, the algorithm searches a set of *Pareto* optimal solutions. A solution is Pareto optimal when no other solution is better in all fitness functions and therefore any of the objectives cannot be improved without degrading another. For instance in our problem, a set of proteins is a Pareto optimal solution when no other set has both a higher density ($f_{density}$) and higher coverage ($f_{coverage}$) and any other Pareto optimal solution with a higher value in one of these function will necessarily have a lower value in the other. The output of the MOGA is a set S of m solutions $S = G^1, G^2, ..., G^m$, which represents an estimation of the whole set of Pareto optimal solutions, also called Pareto front. All the solutions in S are available for custom visualisation and post-processing in the Cytoscape app.

Initialisation. MOGA requires the initialisation of a predefined number of chromosomes (population). The initial population is constructed with chromosomes composed of random proteins in P and as many proteins randomly picked in the neighbourhood of P in G within a radius of 2.

Genetic operators and parameter settings. PEPPER's MOGA optimises $f_{density}(G')$ and $f_{coverage}(G')$ objectives by performing changes driven by mutations (adding or removing a protein) and crossing-over (interchanging sets of proteins in two independent solutions) operators among chromosomes. At each iteration of the algorithm, random chromosome pairs are exposed to these operations and generate offspring sets. Given the two objectives, *fittest* chromosomes are then selected by binary tournament to evolve the population.
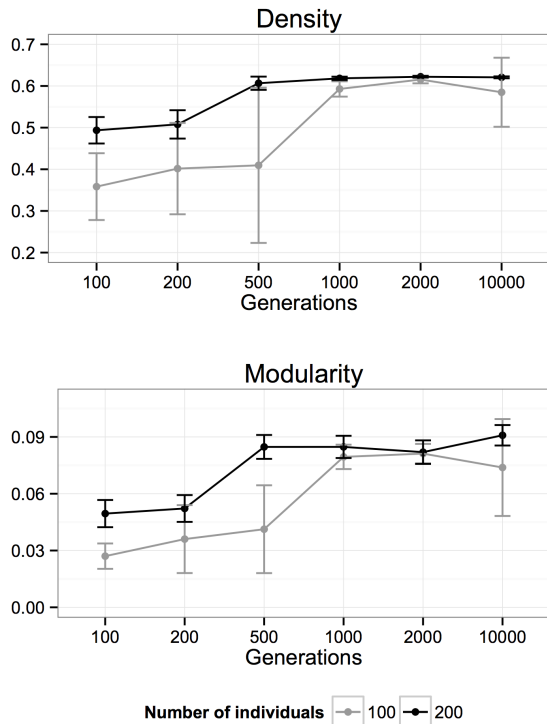
Fig. 1. Density and modularity of the merged optimal Pareto solutions as a function of the number of generations and size of the population. Results obtained on the ATG10 protein of the autophagy proteomic study. The error bar correspond to the standard deviation of 5 replicates.

Parameter	Value
Population size	200
Crossover	$P_C = 0.90$
Mutation	$P_M = 0.10$
Number of generations	1000
Pareto front	10 solutions

Table 1. Parameters of the multi-objective evolutionary algorithm. Crossover and Mutation, the genetic algorithm operators, are expressed as probabilities.

The MOGA requires several parameters which mainly impact the rate of convergence. These parameters include size of the population (number of individuals in a generation), number of generations (iterations), mutation rate, crossover rate and size of the Pareto front to return. In order to define a set of default parameters, we tested the MOGA on a Human proteomics dataset of the autophagy system (Behrends *et al.*, 2010) using the Hippie protein network (Schaefer *et al.*, 2012b) after removing protein interactions originating from the proteomics study. Figure 1 shows density and modularity values of the merged pareto solutions (see next section) as a function of their size and the number of generations. Using 200 individuals per generation allowed to converge rapidly and with lower variance, as early as 500 generations for the example given in Fig. 1. Based on these results and on general observations of other proteins from the same dataset, in PEPPER , default number of generations was set to 1,000 and number of individuals to 200. Maximum coverage is virtually always obtained independently of the parameters and is therefore not shown. The other parameters of the MOGA were set to the standard SPEA2 in the JMetal platform and are reported in table 1.

Merging Pareto optimal solutions. From the union of Pareto optimal solutions $\cup_S$, we devised a simple algorithm to build a consensus *modular* subnetwork noted S_f. The use of modularity is based on a common observation that functional processes are often found in modular subparts of biological networks. This inspired clustering algorithms to use this measure which has led to successful application in particular in protein complex discovery problems (Nepusz *et al.*, 2012). The modularity is defined here for a subset of proteins as the ratio between the number of interactions that occur only between these proteins against the number of other interactions involving these proteins in the whole PPI (G). It can be computed for a subgraph $G' = (V', E')$ as follow :

$$f_{modularity}(V', E') = \frac{\sum_{i,j \in V', i \neq j} E'(i,j)}{2 \times \sum_{i \in V', j \notin V'} E(i,j)}$$

with $G = (V, E)$ and $G' \subseteq G$.

The merge algorithm starts from all the proteins in P that were found in at least one solution ($\cup_S \cap P$) and iteratively tries to add one of the remaining *expansion* proteins ($\cup_S \setminus P$). At each step the *expansion* protein which maximises the overall modularity is kept until it cannot be increased anymore. This greedy algorithm has two characteristics. First, it keeps all the proteins from the initial set P which were identified in at least one of the Pareto optimal solutions of S. Second, it helps removing non-specific proteins which could have been added because of their high connectivity (which will increase density) but low specificity to the subnetwork of interest (which will decrease modularity), typically hub proteins of the network.

The final consensus network S_f is the protein complex predicted by PEPPER. Union of all optimal solutions ($\cup_S$) and predicted complex are the networks that are first generated by PEPPER in Cytoscape, showing a mixture of expanded and initial proteins as well as a *bait* protein if provided. All solutions in S are also available for visualisation and analysis in Cytoscape Results panel.

1.3 Assessment of *predicted* protein complexes

A set of methods are used to analyse the specificity of the predicted *expansions* to the solution and the initial list of proteins of interest (P). This allows indicating the relevance of the overall predicted protein complex and of each of the *expansion* proteins. Four scoring measures are browsable in Cytoscape Results panel for a given predicted complex and viewable as a color code for the *expansion* proteins based on:

- topological connectivity to assess the importance of a protein in connecting the predicted complex
- co-occurrence in a repository of hand-curated protein complexes

- similar functional annotation, particularly in terms of cellular localisation and function

The aforementioned analysis is proposed as an integrated pipeline automatically performed following the evolutionary-base network extraction and merge steps.

Topological considerations. Four topological properties are used: degree and clustering coefficient, known to be good assessment factors in cellular biology and proteomics studies (Glaab *et al.*, 2010; Aittokallio and Schwikowski, 2006; Özgür *et al.*, 2008); modularity, which is used in the merge algorithm and more generally used for protein complex discovery (Nepusz *et al.*, 2012); and closeness centrality, a measure used as an indicator of the overall similarity of a network nodes (Özgür *et al.*, 2008). These measures were calculated for:

1. The whole predicted subnetwork, *i.e.* the solution given by PEPPER noted S_f

2. Only proteins of interest used as an input to PEPPER present in the final solution ($P \cap S_f$)

These measures are reported in Cytoscape Results panel, in which differences between the original list and the final solution serves as a first indicator of PEPPER's predictions importance. Then, each of these topological measures are computed for each of the *expansion* proteins and are summarised in a global topological score ranged in $[0, 1]$ using the following formula:

$$Score_{topology}(X) = \frac{\sum_{\pi \in \Pi} \frac{X_\pi}{max(\pi)}}{|\Pi|}$$

with X a given protein, X_π the measure π associated to the protein X, $max(\pi)$ the maximum observed for measure π in the subnetwork and Π the set of all the topological measures used: degree, clustering coefficient, modularity or closeness centrality.

Overlap with known protein complexes. PEPPER was developed to solve problems encountered in proteomics studies, in particular for protein complex discovery. Therefore, the second measure of similarity takes into account the co-occurrence of predicted proteins in large collections of hand-curated protein complexes. These are available inside the plugin and were retrieved from the CYC2008 database for *S. cerevisiae* (Pu *et al.*, 2009) and the CORUM database (Ruepp *et al.*, 2008) for mammals.

Predicted complexes are evaluated using the matching score (also known as overlap score in Bader and Hogue (2003)):

$$MS(S, R) = \frac{|S \bigcap R|^2}{|S| * |R|}$$

where S and R respectively correspond to the sets of proteins in the predicted and reference complexes. The latter matching score is computed for any reference protein complex that presents at least one proteins in the predicted subnetwork. For each match between a predicted and reference complex, PEPPER also generates and displays its associated performances in terms of sensitivity, precision and geometric accuracy (*cf.* section 2.2).

In order to evaluate and rank *expansion* proteins, each *expansion* is scored based on its occurrence in reference complexes associated to the solution given by PEPPER. This score is weighted by the matching score to give higher ranks to proteins that occur in reference complexes, which are more relevant to the solution. It is computed as follows:

$$Score_{complex}(S, X) = \frac{\sum_{r \in R} |X \cap r| \times MS(P, r)}{|R|}$$

where S is a PEPPER predicted complex, R is the set of reference complexes with at least one protein shared in S and X is one of the *expansions* in S. Known protein complexes matching thus results in a detailed list of overlapping complexes with PEPPER predictions but also provides a score translating *expansions* importance in those complexes.

Common functions and co-localisation. Proteins associating in a complex are necessarily co-localised in the cell and are likely to share a given biological function. Based on this, gene annotations of cellular function and localisation were used to estimate the relevance of the predicted complex and each of the *expansion* proteins. This was computed based on the Gene Ontology (GO) annotations. A hypergeometric test is used to identify Biological Process and Cellular Component annotations that are significantly associated with the predicted protein complex (with $\alpha < 5\%$).

To evaluate *expansion* proteins individually, each of them are scored by the number of annotations they share with those found to be specific to the overall predicted complex as follows:

$$Score_{GO}(X) = \frac{|X_{GO} \cap S_{GO}|}{|S_{GO}|}$$

where S_{GO} is the set of GO terms (Biological Process: *gobp*; or Cellular Component: *gocc*) associated with a solution of PEPPER S (hypergeometric test $\alpha < 5\%$), X is the protein contained in S that is scored and X_{GO} is the set of annotations of X. Each *expansion* protein is scored by the number of functional and localisation terms it shares with the overall predicted complex.

Protein *expansions* global score. The integrated post-processing pipeline provides four distinct scores related to: (i) topology, (ii) reference complexes, (iii) Biological Process GO terms and (iv) Cellular Component GO terms. To summarise the information at a higher level, *expansion* proteins must be characterised by an integrated post-processing score. Several aggregation methods can be used to merge and normalise scores in a $[0, 1]$ range: mean, max or min. In order to identify the best aggregation method, we compared the ranking of the *expansion* proteins using these methods with an approximation of the optimal ranking. The reference ranking is defined by the ranked list that minimises the distance with the rank given by each of the individual scores. This *optimal ranked list* was obtained using the R package RankAggreg Pihur *et al.* (2009), which uses a Cross-Entropy method to identify the ranking minimising the sum of absolute differences with the ranks of each individual score. Because of the computational time required to obtain this optimal ranking, RankAggreg was used only for comparison and was not directly integrated in the pipeline.

Figure 2 shows the distribution of Spearman correlations of several score aggregation methods with the optimal ranking from RankAggreg for 10 sets of *expansions* from 10 pull-down assays led in the Human autophagy system (Behrends *et al.*, 2010). Besides
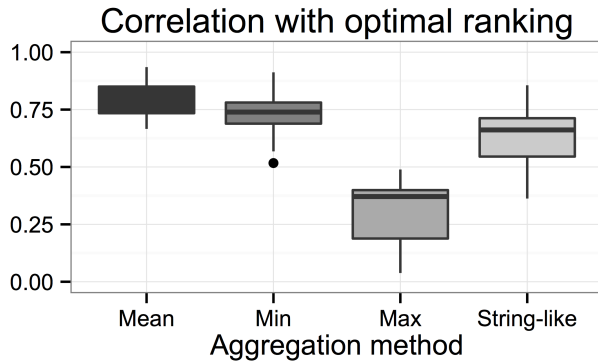
Correlation with optimal ranking

Fig. 2. Distribution of Spearman correlations between optimal rankings of expansion proteins identified RankAggreg and ranking obtain by several score aggregation methods.

mean, max and min, we also used the score integration function introduced in the String database (von Mering, 2004) $(1-\prod_i(1-S_i))$, with S_i corresponding to each individual scores). These results show that the mean of the scores is the closest to the optimal ranking of the expansion proteins. The min function also appears to be an efficient aggregation method; however, a large set of *expansions* present at least one null score (approximately 30%).

In order to bring more flexibility to the global score calculation, PEPPER gives the possibility to add weights to each score and compute a weighted arithmetic mean such as:

$$Score_{postProcess}(X) = \frac{\sum_{i\in\zeta}\omega_i x_i}{\sum_{i\in\zeta}\omega_i}$$

where X represents a specific protein expansion, x its associated score to a post-processing feature, ω the weight given to the latter feature within ζ the set of post-processing assessment criteria:

$$\zeta = \{Score_{topology}, Score_{complex}, Score_{gobp}, Score_{gocc}\}$$

Each score weight has a default value equal to 1, which summarises equally the information from each post-processing feature into a common mean $\frac{1}{n}\sum_{i=1}^{n}x_i$, where n is the overall number of assessment criteria. Choice to modify weights individually is left to users in PEPPER post-processing panel "Overview" tab. Tuning those parameters is a way to make abstraction of certain properties, which may be helpful for results visualisation and interpretation since the plugin automatically updates the overall post-processing score as users modify weights. PEPPER dynamically translates each *expansion* overall score into a red color gradient (the darker the higher) in Cytoscape graphs.

2 PERFORMANCE COMPARISON

2.1 Comparison principles

To evaluate the ability of PEPPER to find relevant protein complexes, we applied it to real Affinity-Purification followed by Mass-Spectrometry (AP-MS) data and compared the results to gold

standard sets of hand-curated reference protein complexes. Each of the AP-MS experiments performed on a single *bait* protein resulted in a list of *preys* and the union of both was used as a *seed* list of proteins. For each *seed*, the best matching reference complex from the gold standard was considered as the complex to be predicted. Therefore, only AP-MS with high matching reference complexes and for which no ambiguity was possible (only one highly matching reference complex) were selected. PEPPER was directly applied to each of these selected *seeds*. State-of-the-art protein complex discovery algorithms based on graph clustering, namely MCODE (Bader and Hogue, 2003) and ClusterONE (Nepusz *et al.*, 2012), were used for performances comparison. Network clustering methods do not aim at finding protein complexes from a *seed* protein list of interest but rather enumerate all protein complexes in a PPI network. Therefore, these two methods were applied to the PPI network also used in PEPPER and the extracted complex with the highest overlap with the *seed* was considered as its associated prediction. For fair comparison, we tested several overlapping measures (intersection, Jaccard and Matching-Score) and reported only the results of the measure with highest performance which was obtained with the absolute size of the intersection.

2.2 Assessing performances

For each of the protein complexes predicted by PEPPER, ClusterONE or MCODE, the overlap of the set of predicted proteins with the known complex was computed as well as four common prediction performance measures:

- True Positive, TP: the number of proteins of the predicted complex that are found in the reference complex.

- True Negative, TN: the number of proteins that are not in the predicted complex and that are not found in the reference complex.

- False Positives, FP: the number of proteins of the predicted complex that are not found in the reference complex.

- False Negative, FN: the number of proteins that are not in the predicted complex that are in the reference complex.

Because the total number of proteins in the PPI network is several orders of magnitude higher than the number of proteins in the predicted or reference complexes, the number of TN provides little information. Therefore, we chose the following measures commonly used in information retrieval:

- sensitivity, also called True Positive Rate (TPR), which evaluates how well positives are predicted,

$$Sn = \frac{TP}{TP + FN} \tag{1}$$

- precision, also called Positive Predictive Value (PPV),

$$Prec = \frac{TP}{TP + FP} \tag{2}$$

- geometric accuracy,

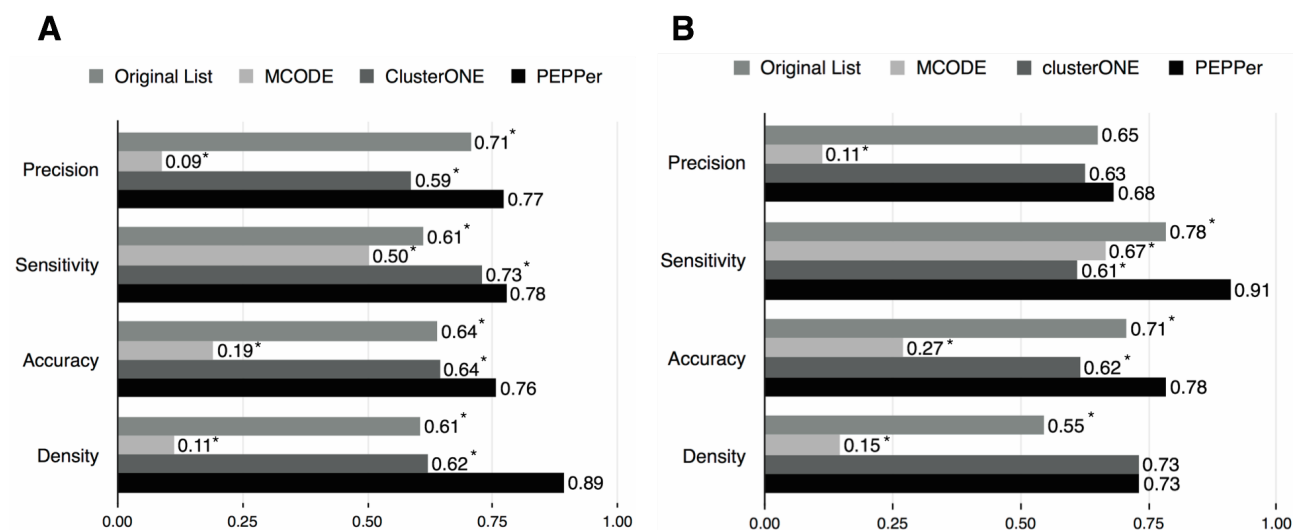$$Acc = \sqrt{Sn * Prec} \tag{3}$$

A

B

Fig. 3. PEPPER, MCODE and ClusterONE performances. (A) Protein complexes predicted from 135 single *Bait* AP-MS experiments in Yeast. (B) Protein complexes predicted from 9 single *Bait* AP-MS experiments in Human. The statistical significance is shown for comparison between MCODE and PEPPER as well as ClusterONE and PEPPER (* : $\alpha = 5\%$)

- density which measures subgraphs connectivity degree

$$Density(V) = \frac{2|E|}{|V|(|V|-1)} \tag{4}$$

where V and E respectively stand for the set of vertices (proteins) and edges (interactions) in a graph.

2.3 Results

A gold standard of manually curated protein complexes was used as a reference for *Saccharomyces cerevisiae* (Pu *et al.*, 2009) and *Homo sapiens* (Ruepp *et al.*, 2008). Single *bait* AP-MS experiments were obtained from a large-scale study in Yeast (Gavin *et al*, 2006) and Human (Choi *et al.*, 2010). For each experiment, the *bait* and its associated set of *preys* were used as the *seed* list of proteins. Data for Yeast was already a set of curated proteins. In Human, only high-confidence proteins (SAINT score greater or equal to 99%) were kept as a list of *preys*. In order to assess the quality of predictions, only experiments for which a reference gold standard is available were selected. To this end, *seed* lists were selected based on the overlap with one of the complexes in the gold standard according to two criteria:

- the *seed* should contain more than 5 proteins in the same gold reference protein complex
- more than 50% of the *seed* should be contained in the same reference complex

From this filtering, 135 and 9 lists of *seeds* were selected for Yeast and Human respectively. The PPI networks used for the analysis were the default Yeast Biogrid network (Stark *et al.*, 2010) and the HIPPIE database (Schaefer *et al.*, 2012a) for Human. Performances of our method are reported in Fig.3 alongside with those of MCODE, ClusterONE and of the original list of proteins used as seeds. Significant differences between PEPPER and

MCODE or ClusterONE were computed using Student's two-sided t-test with $\alpha = 5\%$. The higher performance of PEPPER is statistically significant except for density and precision in Human species. Moreover, unlike ClusterONE, MCODE identifies protein complexes for only a small portion of the total number of proteins. In Yeast, only approximately 40% of the proteins of the PPI network (2,449 out of 5,968 proteins) where assigned to a predicted complex whereas ClusterONE predicted a complex for nearly 80% of the proteins (4,742). The Human PPI networks being less connected, these proportions drop to 17% for MCODE and 36% for ClusterONE. Therefore, many seed lists of proteins that can be assigned to a known protein complex cannot however be mapped to an MCODE predicted complex. The results obtained with PEPPER in Yeast or Human always showed an increase in all of the classification performance measures as compared to the original list of proteins or to the two tested methods. Interestingly, this increase in performance is associated with an increase of the density in Yeast. In Human, however, ClusterONE and PEPPER find protein complexes with very similar densities. Yet, PEPPER significantly outperforms ClusterONE with an average increase of 16% in accuracy and of 30% in sensitivity. These results suggest that extracting solutions solely based on optimising topological measures can be improved by integrating the context specificity of real experimental data.

3 CASE STUDY

An example of usage of PEPPER is shown in Fig. 4 for a particular application on the Human protein *WDR92*.

3.1 Input data

The experimental results of an AP-MS assay performed using *WDR92* as a *bait* protein were obtained from a previously published study (Choi *et al.*, 2010). From the raw list of proteins identified in
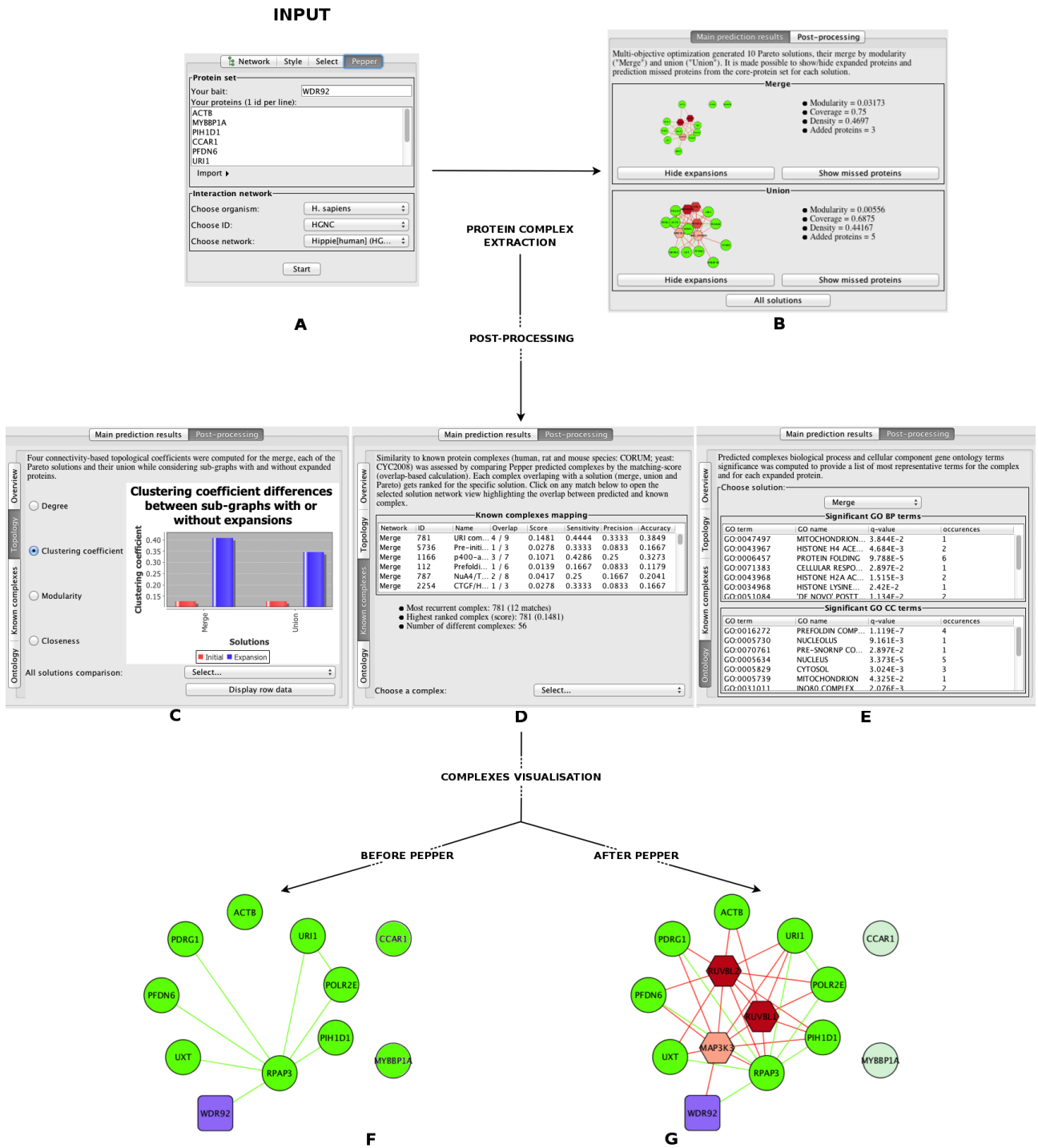
Fig. 4. A view of the user interface of PEPPER in Cytoscape and the application to the *WDR92* case study. (A) Necessary inputs are the organism (with default PPI networks for Human, Yeast and Mouse) and the list of *seed* proteins. (B) The predicted protein complexes, including first the final merged solution then all the Pareto optimal solutions and its union, are directly visible in the first tab of the result panel. The second tab shows the results of the post-processing scores including (C) topology feature differences when considering extracted subgraphs without or with PEPPER *expansions*, (D) occurrence of proteins of the solutions in reference protein complexes and (E) enriched GO terms. The set of proteins of a given solution that co-occur in a particular reference protein complex or are annotated with a specific GO term, are highlighted clicking on the annotation of interest in the result panel. The network formed by adding known interactions between proteins of the *seed* list (F) and between the proteins predicted by PEPPER to form a complex (G) are visible at the end of the run. Green nodes are *prey* proteins, the squared purple node is the *bait* and hexagonal nodes are *expansions* predicted by PEPPER with a colour code (light to dark red) increasing with significance. Green edges represent interactions between *seed* proteins whereas red edges connect *expansion* proteins added by PEPPER, both originating from the input PPI network.

the assay, 10 high confidence *prey* proteins (SAINT score greater or equal to 99%) were selected (list available as Supplementary File). In all, 8 interactions were found between all the proteins identified by the assay and three of the preys (*ACTB*, *CCAR1* and *MYBBP1A*) were not connected to any of the other preys.

3.2   Main prediction results

From the *seed* list, PEPPER generated a consensus subgraph covering 75% of the initial *seed* proteins with density and modularity values of respectively 0.47 and 0.032 (Fig. 4B). The complex predicted by PEPPER and the original network built from the *seed* list are shown in Figure 4F-G. PEPPER predicted three *expansions*: *RUVBL1*, *RUVBL2* and *MAP3K3*.

The *expansion* proteins predicted by PEPPER greatly increased the connectivity of the initial solutions, which was measurable for several topological features described in Section 1.3 (example of the clustering coefficient in Fig. 4C). For instance, degree and modularity values showed more than a two-fold increase. The *expansions* also slightly increased the overall subnetwork closeness centrality. Finally, the clustering coefficient was more than three times higher with than without *expansions*. Proteins added by PEPPER significantly affected the apparition of cliques (fully connected components) in the subnetwork, thus increasing the complex connectivity.

No protein complex was found to be associated with the original list of *seed* proteins. Matching PEPPER's predictions to the reference protein complexes resulted in 56 mapped known complexes when considering all solutions (Fig. 4D). Among these, 43 were found to overlap with at least one protein of the final solution. The best matching complex was the *URI complex (Unconventional prefolding RPB5 Interactor)* with a matching score of 0.1481, associated to two of the *seed* proteins and two of the *expansions*. While the interpretation necessitates further experiments to validate whether this complex is formed in the studied system, PEPPER provides directions for validation by prioritizing the candidates.

Among the significantly enriched cellular processes, several functions related to histones acetylation and methylation were found and respectively linked to *RUVBL1/2* (*expansions*) and *WDR92* (*bait*) proteins. Annotation of proteins and their cellular localisation provided information about the possible localisation of the complex in the nucleus, which is consistent with the putative association with the *URI complex* and the possible function in histone modifications (Fig. 4E).

Finally, *expansion* proteins were found with post-processing scores of 0.21 for *MAP3K3* and 0.42 and 0.44 respectively for *RUVBL2* and *RUVBL1* proteins. *RUVBL1* actually appears as a *prey* protein of *WDR92* with a very low number of unique peptide (only one) and a low SAINT score (0%) and therefore did not pass the detection threshold. Furthermore, AP-MS using both *RUVBL1* and *RUVBL2* also identified *WDR92* as a *prey*. Altogether, these results strongly suggest that *WDR92* forms a complex with both of these proteins predicted as *expansions* by PEPPER.

REFERENCES

Aittokallio, T. and Schwikowski, B. (2006). Graph-based methods for analysing networks in cell biology. *Briefings in Bioinformatics*, **7**(3), 243–255.

Bader, G. D. and Hogue, C. W. V. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, **4**, 2.

Behrends, C., *et al.* (2010). Network organization of the human autophagy system. *Nature*, **466**(7302), 68–76.

Choi, H., *et al.* (2010). Saint: probabilistic scoring of affinity purification-mass spectrometry data. *Nature methods*, **8**(1), 70–73.

Durillo, J. J. and Nebro, A. J. (2011). jmetal: A java framework for multi-objective optimization. *Advances in Engineering Software*, **42**, 760–771.

Feige, U., *et al.* (2001). The dense k-subgraph problem. *Algorithmica*, **29**(3), 410–421.

Gavin *et al*, A. C. (2006). Proteome survey reveals modularity of the yeast cell machinery. *Nature*, **440**(7084), 631–636.

Glaab, E., *et al.* (2010). TopoGSA: network topological gene set analysis. *Bioinformatics*, **26**(9), 1271–1272.

Nepusz, T., *et al.* (2012). Detecting overlapping protein complexes in protein-protein interaction networks. *Nature Methods*, **9**, 471–472.

Özgür, A., *et al.* (2008). Identifying gene-disease associations using centrality on a literature mined gene-interaction network. *Bioinformatics*, **24**(13), i277–i285.

Pihur, V., *et al.* (2009). RankAggreg, an R package for weighted rank aggregation. *BMC Bioinformatics*, **10**(1), 62.

Pu, S., *et al.* (2009). Up-to-date catalogues of yeast protein complexes. *Nucleic Acids Research*, **37**(3), 825–831.

Ruepp, A., *et al.* (2008). Corum: the comprehensive resource of mammalian protein complexes. *Nucleic Acids Research*, **36**(Database-Issue), 646–650.

Schaefer, M. H., *et al.* (2012a). : Integrating Protein Interaction Networks with Experiment Based Quality Scores. *PLoS ONE*, **7**(2), e31826.

Schaefer, M. H., *et al.* (2012b). HIPPIE: Integrating Protein Interaction Networks with Experiment Based Quality Scores. *PLoS ONE*, **7**(2), e31826.

Stark, C., *et al.* (2010). The BioGRID Interaction Database: 2011 update. *Nucleic Acids Research*, **39**(Database), D698–D704.

von Mering, C. (2004). STRING: known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, **33**(Database issue), D433–D437.

Zitzler, E., *et al.* (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. Technical report.