Decipher the essentials and sub-networks from complex biological networks for drug target selection

複雜生物網路的解析

Reporter: Chia-Hao Chin

Outline

- Prediction of Essential Proteins
- Prediction of Functional Modules

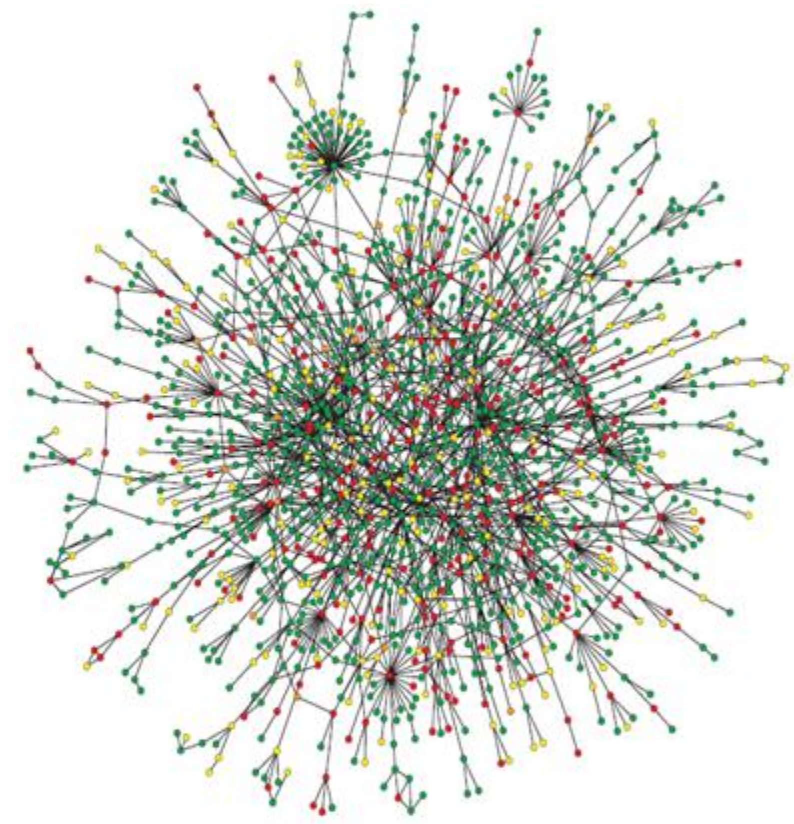
Prediction of Essential Proteins

- A gene (or its associated protein) to be essential if its deletion leads to the loss of cell viability.

- H. Jeong, Z. N. Oltvai, and A. L. Barabsi, "Prediction of protein essentiality based on genomic data," ComPlexUs, vol. 1, pp. 19-28, 2003.

A Yeast Protein Interaction Network

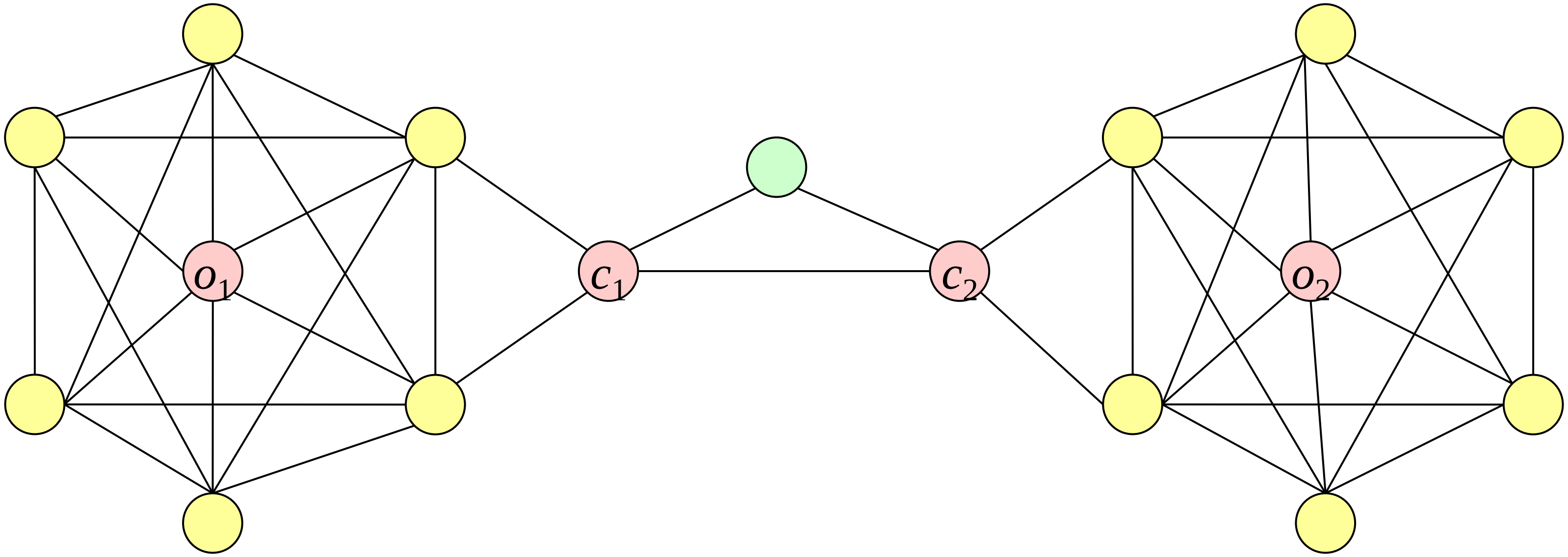
- 1870 proteins,2240 interactions.
  - Red: **lethal**.
  - Green: non-lethal.
  - Orange: slow growth.
  - **Yellow:** unknown..



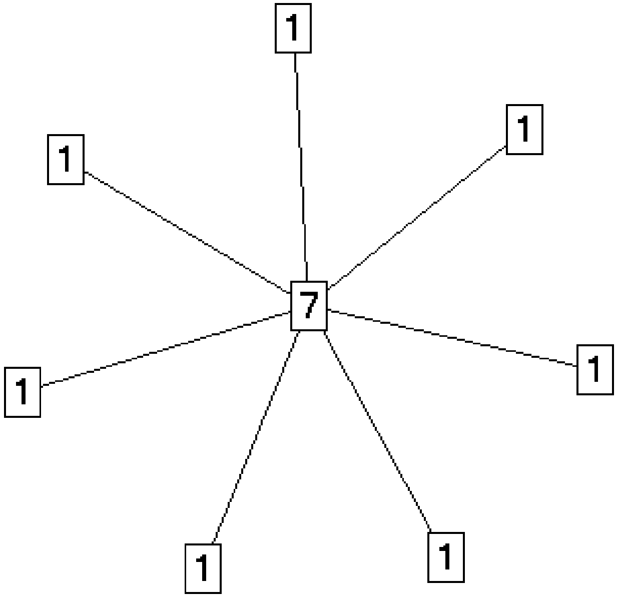
- Jeong H., Mason S. P., Barabasi A. L. and Oltvai Z. N., "Lethality and centrality in protein networks," Nature, 2001.

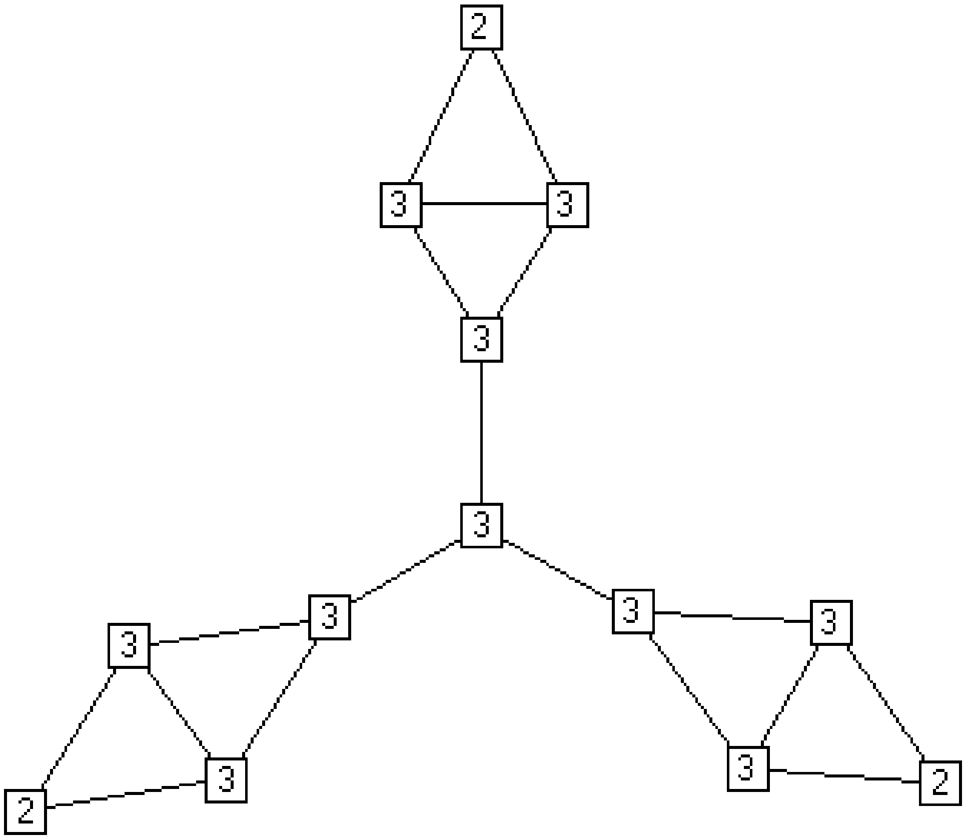
Two Types of Vertices Found by Previous Methods

- Module organizers (for example, o_1 and o_2)
- Module connectors (for example, c_1 and c_2)



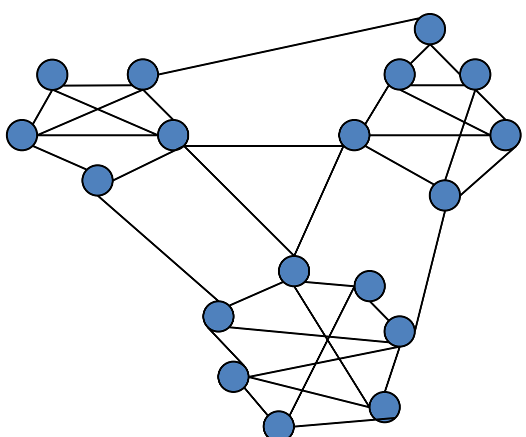
Degree method



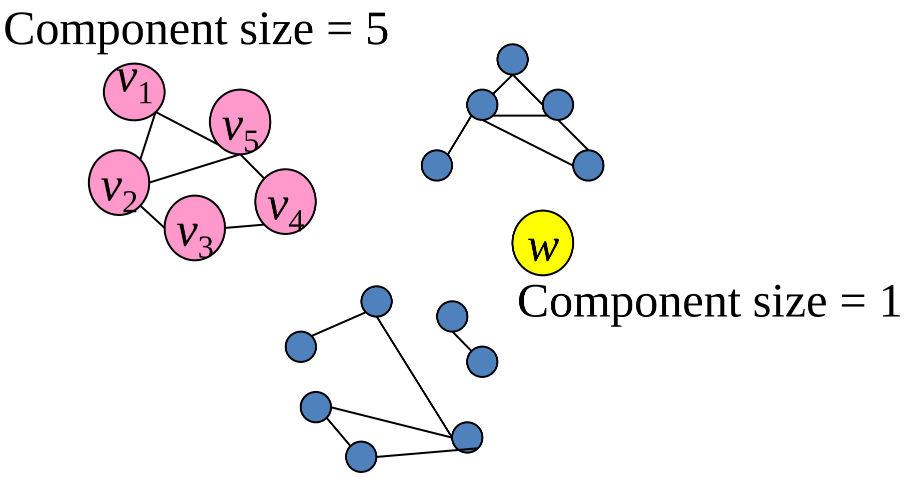


Edge Percolation Component (EPC)

- Threshold p: a given percolation probability
- Generate a subgraph G'
  - For each edge $e = (i, j)$
    - assign a random number p_{ij}.
    - Remove the edge $e=(i, j)$ if $p_{ij} > p$.



A graph G



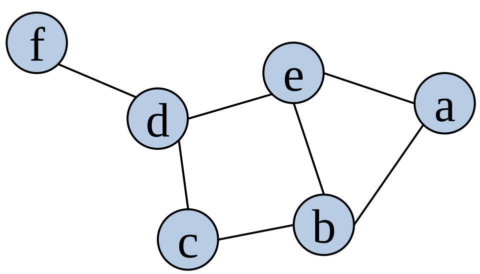
A subgraph G'

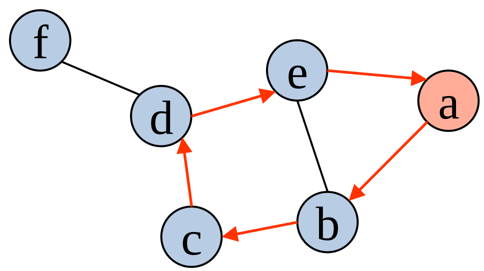
Subgraph Centrality (SC)

- For each node *v*
  - $u_k(v)$:the number of close walks of *v* of length *k*.
  - The subgraph centrality of *v*

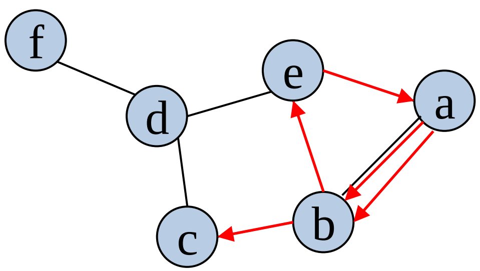
$$SC(v) = \sum_{k=1}^{\infty} \frac{\mu_k(v)}{k!}$$

A example graph





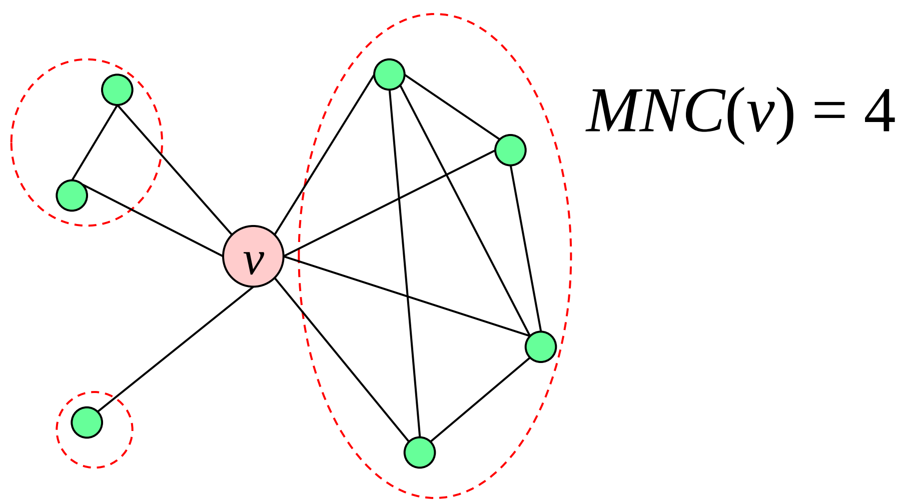
(a, b, c, d, e, a) is a closed walk of length 5.



(a, b, e, a, b, c) is an open walk with length 5

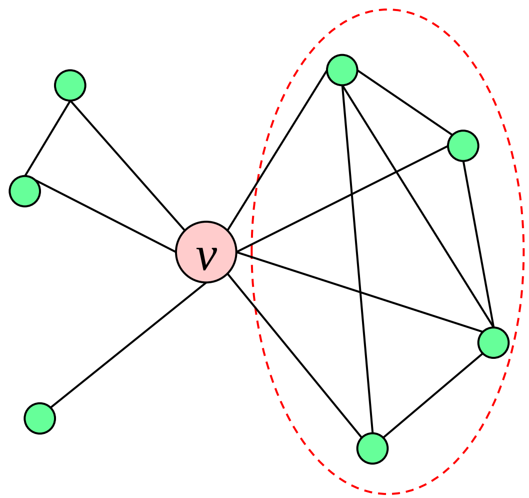
Maximum Neighborhood Component (MNC)

- The neighborhood $N(v)$ is the set of nodes adjacent to v that does not contain node v. The score of node v, $MNC(v)$, is defined to be the size of the maximum connected component of the subgraph induced by $N(v)$.



Density of Maximum Neighborhood Component (DMNC)

- For a node v, let N be the number of nodes and E be the number of edges of $MNC(v)$. The score of node v, $DMNC(v)$, is defined to be E/N^{ε} for some $1 \le \varepsilon \le 2$.

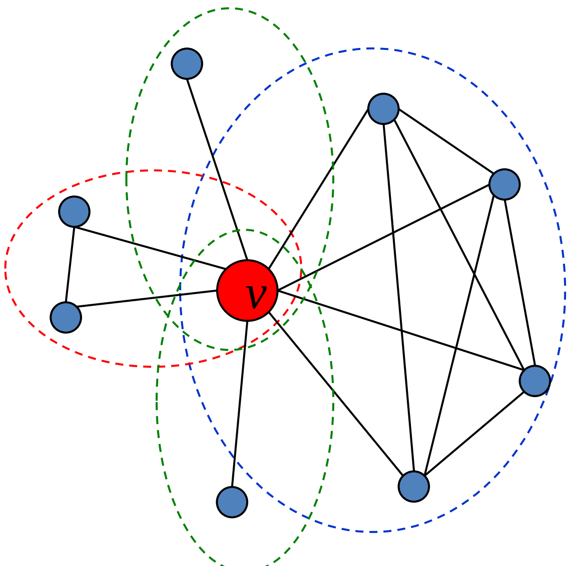


$$DMNC(v) = \frac{5}{4^{\varepsilon}}$$

Maximal Clique Centrality (MCC)

- Given a vertex v
  - $S(v) = \{C \mid C$ is a maximal clique and $v \in C\}$

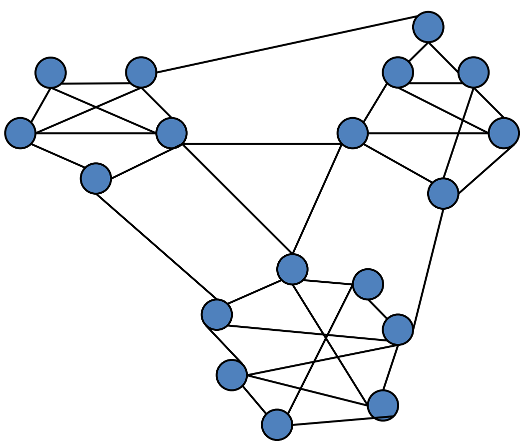
An example



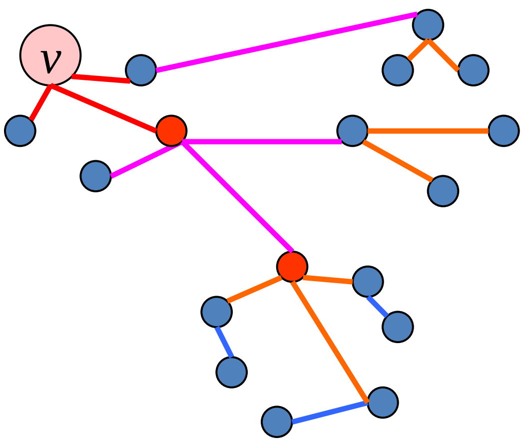
$$MCC(v) = 2*(2-1)! + (3-1)! + (5-1)!$$
$$= 2*(1) + (1*2) + (1*2*3*4)$$

BottleNeck method(BN)

- For each node *v* in the undirected PPI graph
  - T_v : a shortest pathtree rooted at *v*.
    - n_v: the size of T_v.
    - Bottleneck node *w*: at least $n_v/4$ paths of T_v "meet" at *w*.



A graph *G*



T_v: a shortest path tree rooted at *v*.

- $|V|/4 = 18/4 = 4.5$

Centralities Related with Shortest Path

Eccentricity $$\mathcal{C}_{ecc}(v) := \frac{1}{\max\{\mathrm{dist}(v,w) : w \in V\}}$$

Closeness $$\mathcal{C}_{clo}(v) := \frac{1}{\sum_{w \in V} \mathrm{dist}(v,w)}$$

Radiality $$\mathcal{C}_{rad}(v) := \frac{\sum_{w \in V}(\Delta_G + 1 - \mathrm{dist}(v,w))}{n-1}$$

Stress $$\mathcal{C}_{str}(v) := \sum_{s \neq v \in V} \sum_{t \neq v \in V} \sigma_{st}(v)$$

Betweenness $$\mathcal{C}_{spb}(v) := \sum_{s \neq v \in V} \sum_{t \neq v \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

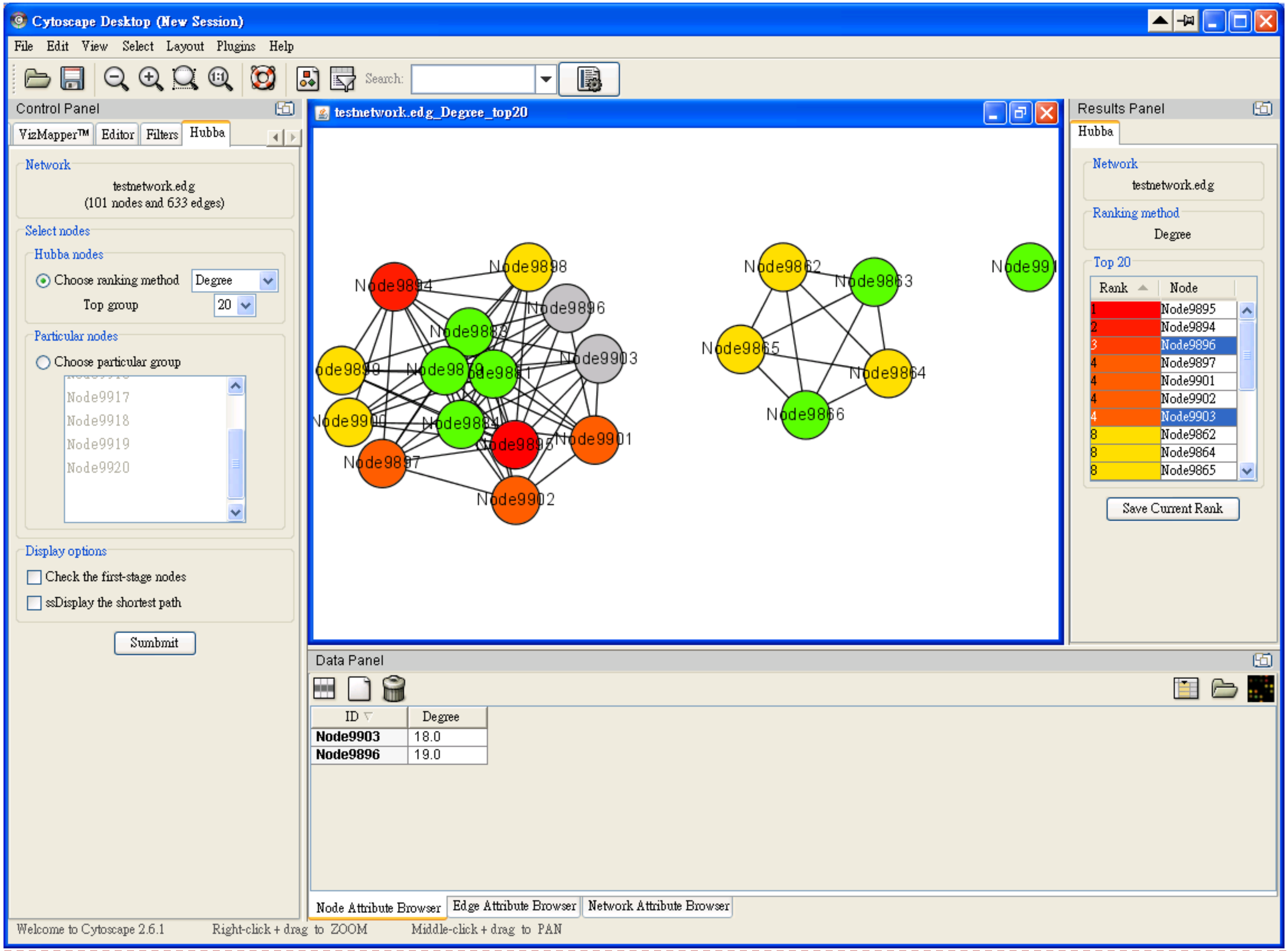
$\mathrm{dist}(v,w)$ is the length of a shortest path between the vertices s and t.

σ_{st} denote the number of shortest paths from s to t

$\sigma_{st}(v)$ the number of shortest path from s to t that use the vertex v.

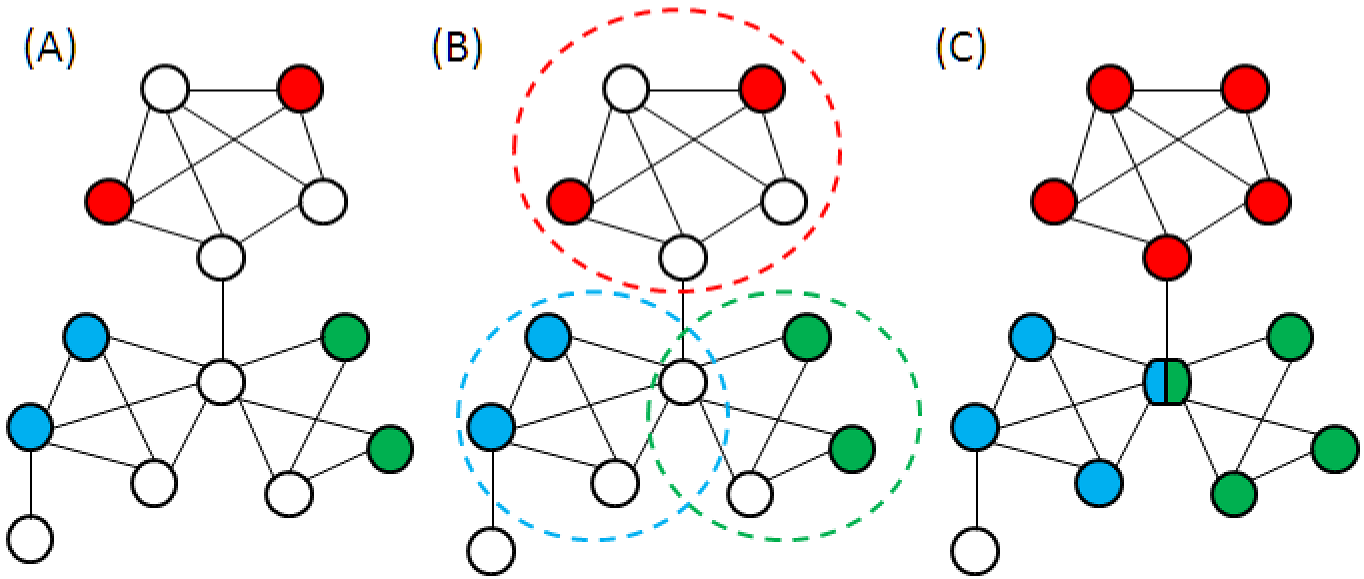
Hubba (http://hub.iis.sinica.edu.tw/Hubba/)

cytoHubba (http://hub.iis.sinica.edu.tw/cytoHubba/)



Prediction of Functional Modules

- A functional module is a discrete entity whose function is separable from those of other modules.

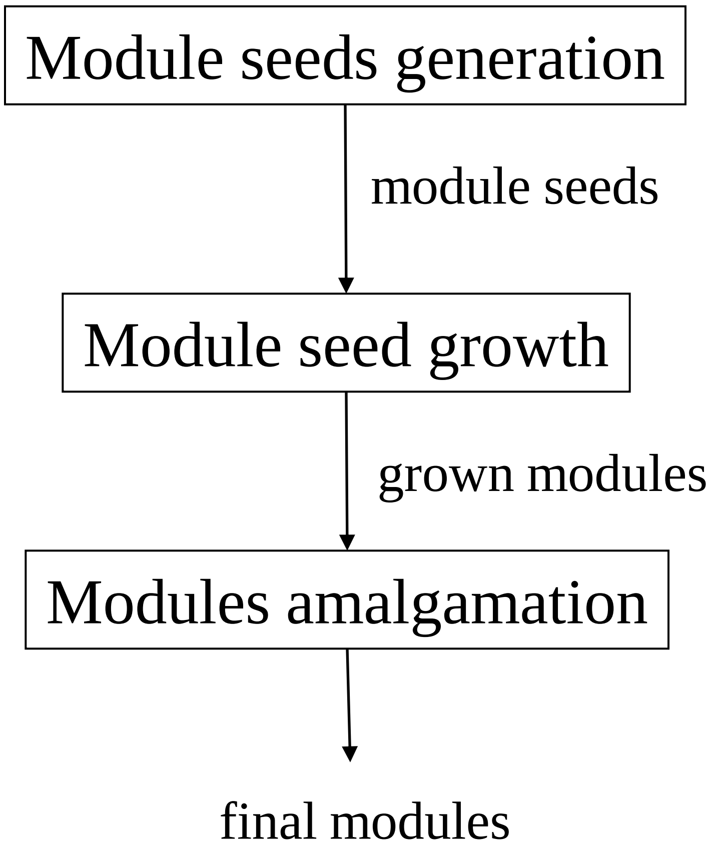


- L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray, "From molecular to modular cell biology," Nature, vol. 402, pp. C47-C52, Dec 2 1999.

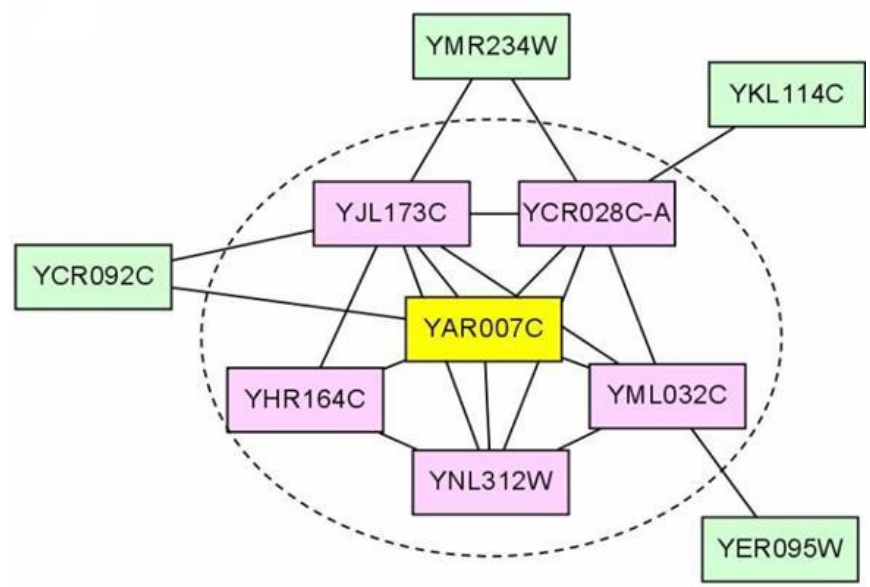
Existing Graph Clusterings Methods

- **CPM**
  - I. Derenyi, G. Palla, and T. Vicsek, "Clique percolation in random networks," Physical Review Letters, vol. 94, Apr 29 2005.
- **FastGreedy**
  - A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," Physical Review E, vol. 70, Dec 2004.
- **Leading Eigenvector**
  - M. E. J. Newman, “Finding community structure in networks using the eigenvectors of matrices,” Phys. Rev. E 74, 036104 (2006).
- **MCL**
  - Stijn van Dongen, "Graph clustering by flow simulation," Ph.D. Thesis, University of Utrecht, The Netherlands, 2000.
- **SpinGlass**
  - P. Pons and M. Latapy, "Computing communities in large networks using random walks," Computer and Information Sciences - ISCIS 2005, Proceedings, vol. 3733, pp. 284-293, 2005.
- **WalkTrap**
  - J. Reichardt and S. Bornholdt, "Statistical mechanics of community detection," Physical Review E, vol. 74, Jul 2006.

The Overview of HUNTER



An Example



An Integration Method

Input:

G=(V, E, w) is a undirected PPI network;

C={ S | $S \subseteq V$ is a cluster generated by a clustering method};

Output:

P is an integrated clustering result;

Description:

1. $P = \varnothing$
2. Sort C by community score in descending order;
3. **for** all $S \in C$ **do**
4. **if** $|S| < 150$ **and** $CS(G, S) > 0$ **then**
5. **if** $\forall\ T \in P$ *s.t.* $2\times|S \cap T| \le \min(|S|, |T|)$ **then**
6. $P = P \cup \{S\}$
7. Output P

$$CS(G, S) = \frac{\text{The number of triangle in } S}{\text{Average shortest path of } S}$$

Spotlight(http://hub.iis.sinica.edu.tw/spotlight/)