Bioinformatics 3

V 5 – Robustness and Modularity

Mon, Oct 29, 2012

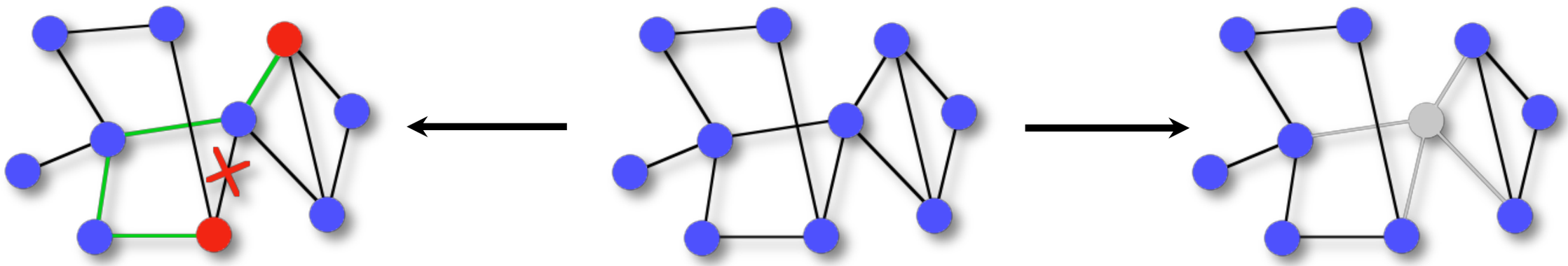
Network Robustness

Network = set of connections

Failure events:
- loss of edges
- loss of nodes (together with their edges)

→ loss of connectivity
- paths become longer (detours required)
- connected components break apart

→ network characteristics change



→ **Robustness** = how much does the network (not) change when edges/nodes are removed

Error and attack tolerance of complex networks

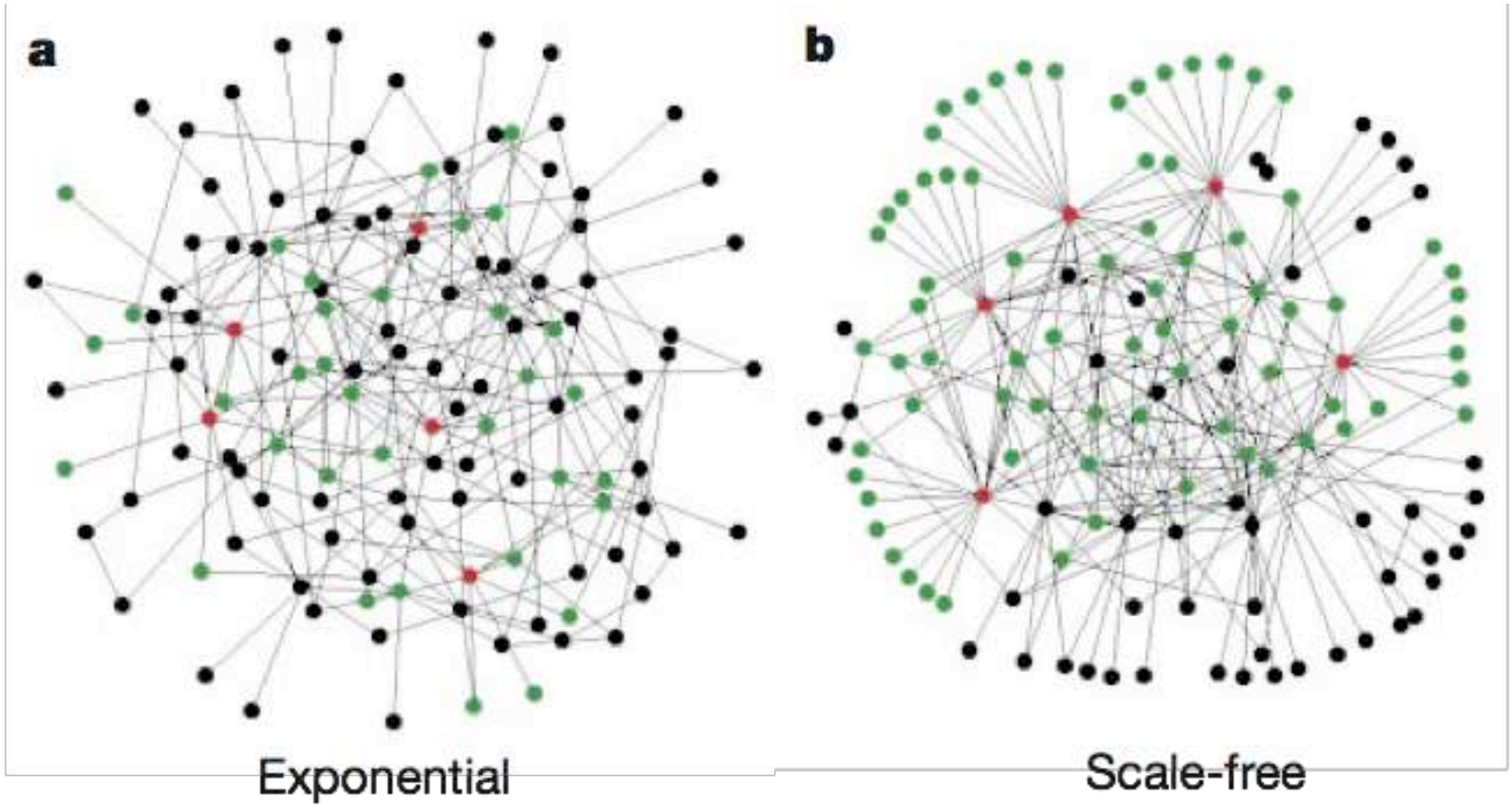
Réka Albert, Hawoong Jeong & Albert-László Barabási

Department of Physics, 225 Nieuwland Science Hall, University of Notre Dame, Notre Dame, Indiana 46556, USA

Many complex systems display a surprising degree of tolerance against errors. For example, relatively simple organisms grow, persist and reproduce despite drastic pharmaceutical or environmental interventions, an error tolerance attributed to the robustness of the underlying metabolic network[1]. Complex communication networks[2] display a surprising degree of robustness: although key components regularly malfunction, local failures rarely lead to the loss of the global information-carrying ability of the network. The stability of these and other complex systems is often attributed to the redundant wiring of the functional web defined by the systems' components. Here we demonstrate that error tolerance is not shared by all redundant systems: it is displayed only by a class of inhomogeneously wired networks,

millan Magazines Ltd NATURE | VOL 406 | 27 JULY 2000 | www.nature.com

Random vs. Scale-Free



130 nodes, 215 edges

The **top 5** nodes with the highest *k* **connect** to…

… 27% of the network

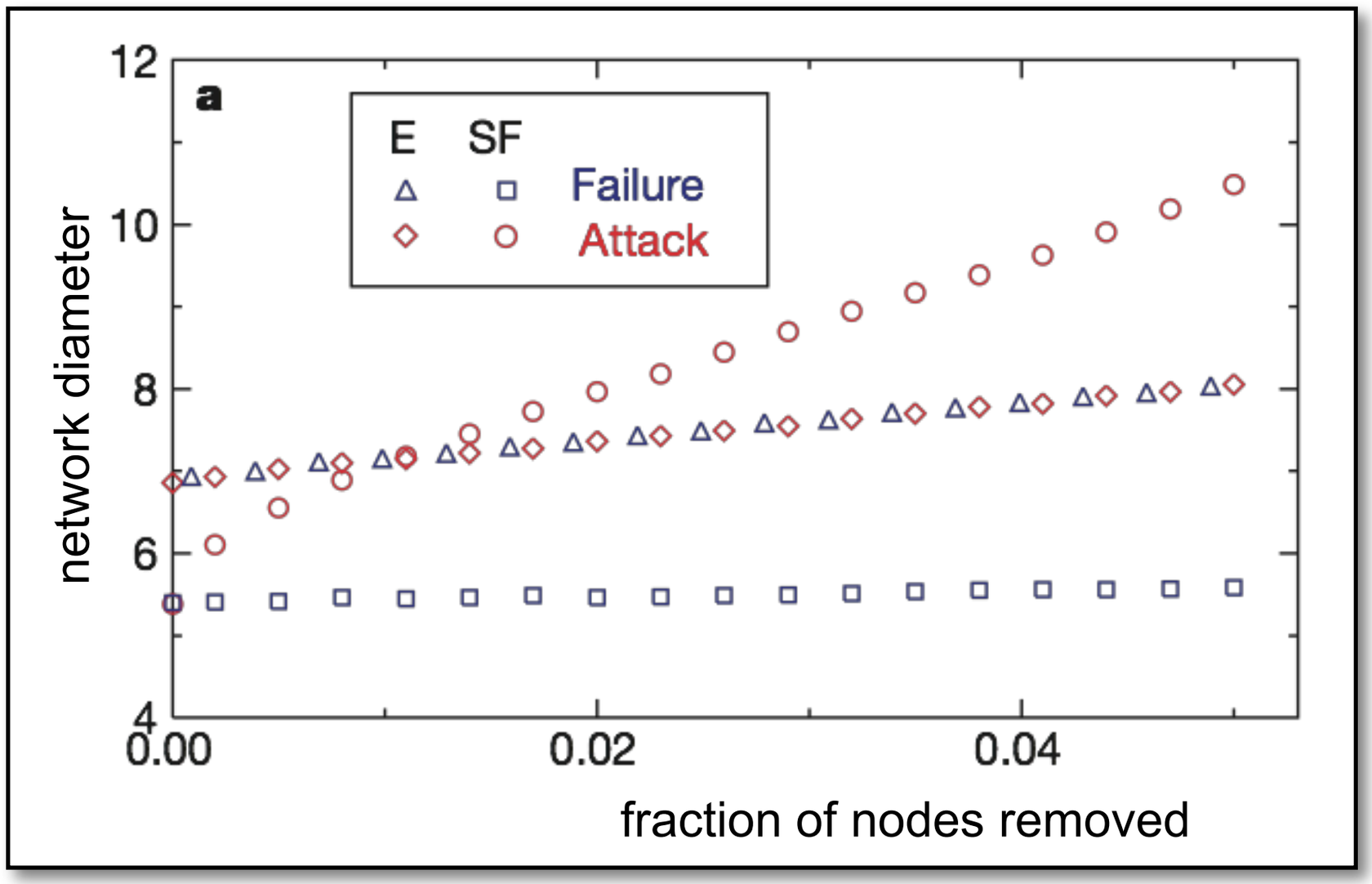
… 60% of the network

Albert, Jeong, Barabási, *Nature* **406** (2000) 378

Failure vs. Attack

Failure: remove **randomly** selected nodes

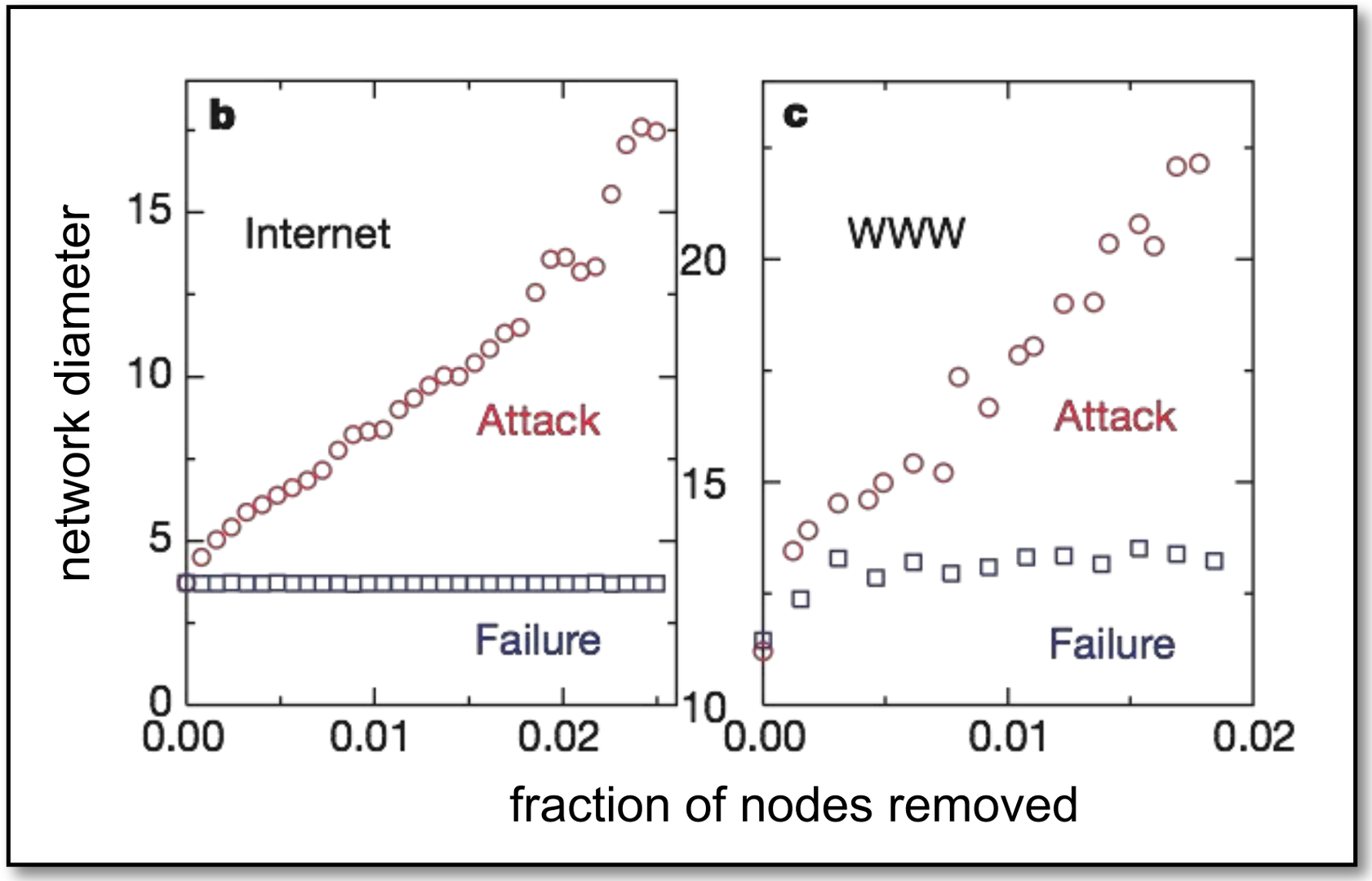
Attack: remove nodes with highest **degrees**



$N = 10000$, $L = 20000$, but effect is size-independent

Two VINs

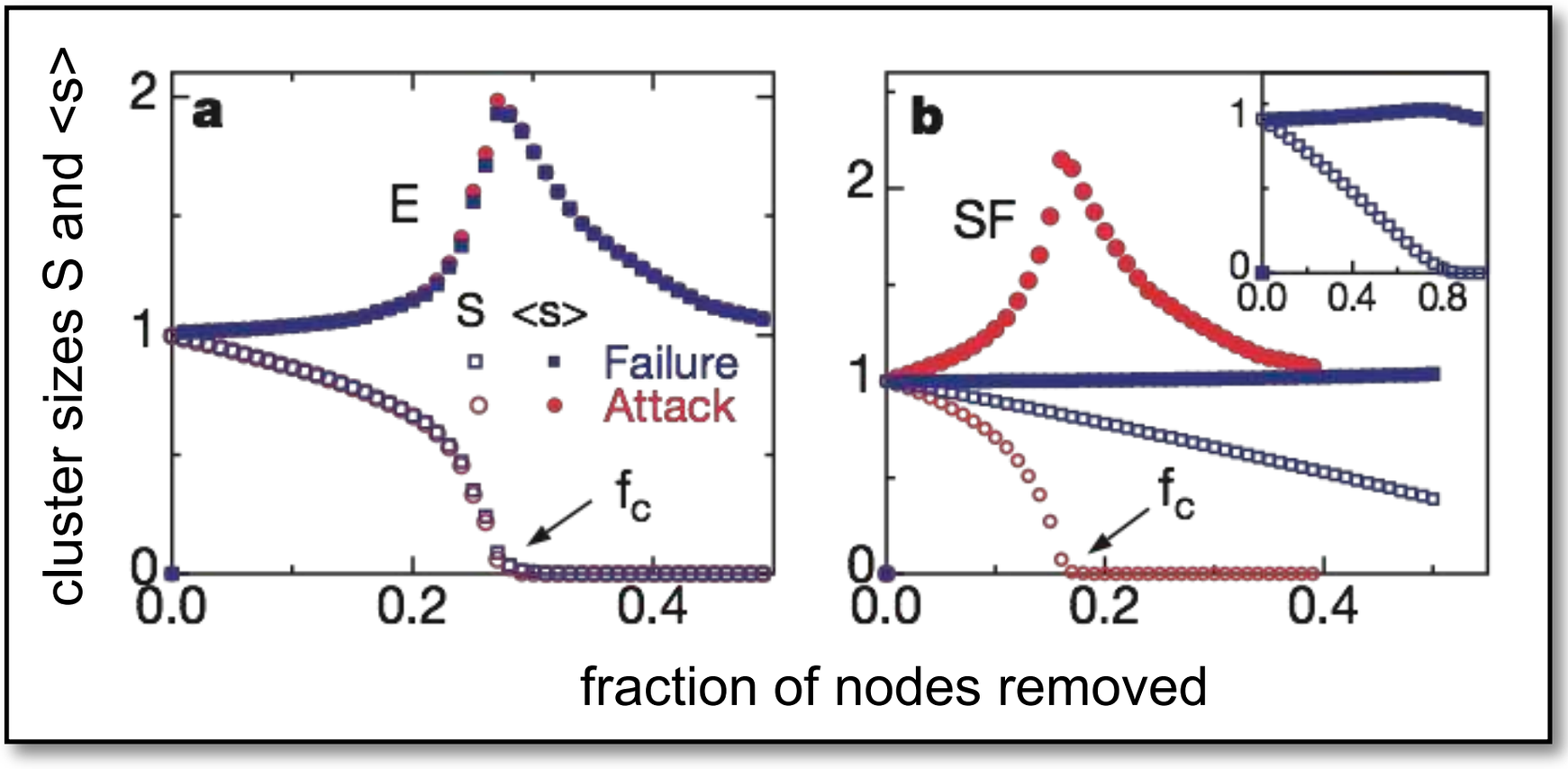
Scale-free:
- very **stable** against random **failure** ("packet re-rooting")
- very **vulnerable** against dedicated **attacks** ("9/11")



http://moat.nlanr.net/Routing/rawdata/ :
6209 nodes and 12200 links (2000)

WWW-sample containing 325729 nodes and 1498353 links

Network Fragmentation



Relative size of the largest clusters S

Average size of the isolated clusters <s> (except the largest one)

Random network:
- **no difference** between attack and failure (homogeneity)
- fragmentation threshold at $f_c \gtrsim 0.28$ ($S \approx 0$)

Scale-free network:
- **delayed fragmentation** and isolated nodes for failure
- critical breakdown under attack at $f_c \approx 0.18$

Mesoscale properties of networks
- identify cliques and highly connected clusters

Most relevant processes in biological networks correspond to the mesoscale (5-25 genes or proteins) not to the entire network.

However, it is computationally enormously expensive to study mesoscale properties of biological networks.
e.g. a network of 1000 nodes contains 1×10^{23} possible 10-node sets.

Spirin & Mirny analyzed combined network of protein interactions with data from CELLZOME, MIPS, BIND: 6500 interactions.

Identify connected subgraphs

The network of protein interactions is typically presented as an undirected graph with proteins as nodes and protein interactions as undirected edges.

Aim: identify **highly connected subgraphs** (clusters) that have more interactions within themselves and fewer with the rest of the graph.

A fully connected subgraph, or **clique**, that is not a part of any other clique is an example of such a cluster. The „maximum clique problem“ – finding the largest clique in a given graph is known be NP-hard.

In general, clusters need not to be fully connected.

Measure density of connections by $Q = \frac{2m}{n(n-1)}$

where n is the number of proteins in the cluster and m is the number of interactions between them.

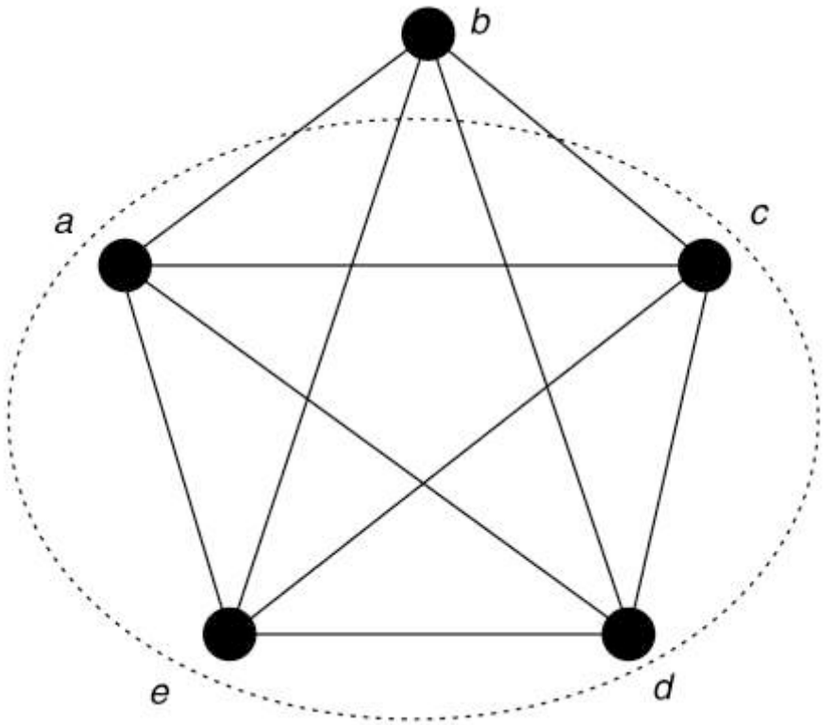
Spirin, Mirny,
PNAS 100, 12123 (2003)

Clique and Maximal Clique

A **clique** is a **fully** connected sub-graph, that is, a set of nodes that are all neighbors of each other.

In this example, the whole graph is a clique and consequently any subset of it is also a clique, for example *{a,c,d,e}* or *{b,e}*.

A **maximal clique** is a clique that is not contained in any larger clique. Here only *{a,b,c,d,e}* is a maximal clique.



Gagneur et al. Genome Biology 5, R57 (2004)

(method I) Identify all fully connected subgraphs (cliques)

The general problem - finding all cliques of a graph - is very hard.
Because the protein interaction graph is sofar very sparse (the number of interactions (edges) is similar to the number of proteins (nodes), this can be done quickly.

To find cliques of size *n* one needs to enumerate only the cliques of size *n-1.*

The search for cliques starts with $n = 4$, pick all (known) pairs of edges
(6500 × 6500 protein interactions) successively.
For every pair *A-B* and *C-D* check whether there are edges between *A* and *C, A* and *D, B* and *C,* and *B* and *D.* If these edges are present, *ABCD* is a clique.

For every clique identified, *ABCD,* pick all known proteins successively.
For every picked protein *E,* if all of the interactions *E-A, E-B, E-C,* and *E-D* exist, then *ABCDE* is a clique with size 5.

Continue for *n = 6, 7, ...*
The largest clique found in the protein-interaction network has size 14.

Spirin, Mirny, PNAS 100, 12123 (2003)

(I) Identify all fully connected subgraphs (cliques)

These results include, however, many redundant cliques.
For example, the clique with size 14 contains 14 cliques with size 13.

To find all nonredundant subgraphs, mark all proteins comprising the clique of size 14, and out of all subgraphs of size 13 pick those that have at least one protein other than marked.

After all redundant cliques of size 13 are removed, proceed to remove redundant twelves etc.

In total, only 41 nonredundant cliques with sizes 4 - 14 were found.

Spirin, Mirny, PNAS 100, 12123 (2003)

(method II) Monte Carlo Simulation

Use MC to find a tight subgraph of a predetermined number of *M* nodes.

At time *t = 0,* a random set of *M* nodes is selected.
For each pair of nodes *i,j* from this set, the shortest path L_{ij} between *i* and *j* on the graph is calculated.
Denote the sum of all shortest paths L_{ij} from this set as L_0.
At every time step one of *M* nodes is picked at random, and one node is picked at random out of all its neighbors.

The new sum of all shortest paths, L_1, is calculated if the original node were to be replaced by this neighbor.
If $L_1 < L_0$, accept replacement with probability 1.
If $L_1 > L_0$, accept replacement with probability $\exp^{-\frac{L_1 - L_0}{T}}$
where *T* is the effective temperature.

Spirin, Mirny, PNAS 100, 12123 (2003)

(method II) Monte Carlo Simulation

Every tenth time step an attempt is made to replace one of the nodes from the current set with a node that has no edges to the current set to avoid getting caught in an isolated disconnected subgraph.

This process is repeated
(i) until the original set converges to a complete subgraph, or
(ii) for a predetermined number of steps,
after which the tightest subgraph (the subgraph corresponding to the smallest L_0) is recorded.

The recorded clusters are merged and redundant clusters are removed.

Spirin, Mirny, PNAS 100, 12123 (2003)

Merging Overlapping Clusters

A simple statistical test shows that nodes which have only one link to a cluster are statistically insignificant. Clean such statistically insignificant members first.

Then merge overlapping clusters:
For every cluster A_i find all clusters A_k that overlap with this cluster by at least one protein.

For every such found cluster calculate Q value of a possible merged cluster $A_i \cup A_k$. Record cluster $A_{best}(i)$ which gives the highest Q value if merged with A_i.

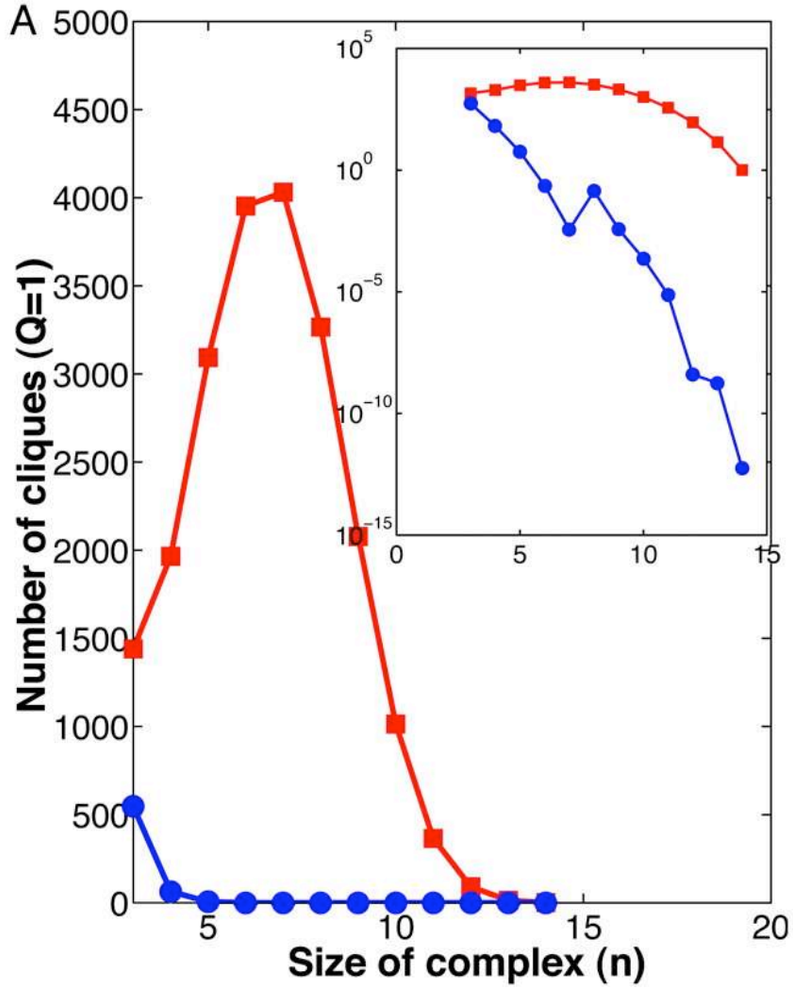
After the best match is found for every cluster, every cluster A_i is replaced by a merged cluster $A_i \cup A_{best}(i)$ unless $A_i \cup A_{best}(i)$ is below a certain threshold value for Q_C.

This process continues until there are no more overlapping clusters or until merging any of the remaining clusters will make a cluster with Q value lower than Q_C.

Spirin, Mirny, PNAS 100, 12123 (2003)

Statistical significance of complexes and modules

Number of complete cliques (Q = 1) as a function of clique size enumerated in the network of protein interactions (red) and in randomly rewired graphs (blue, averaged >1,000 graphs where number of interactions for each protein is preserved).



Inset shows the same plot in log-normal scale. Note the dramatic enrichment in the number of cliques in the protein-interaction graph compared with the random graphs. Most of these cliques are parts of bigger complexes and modules.

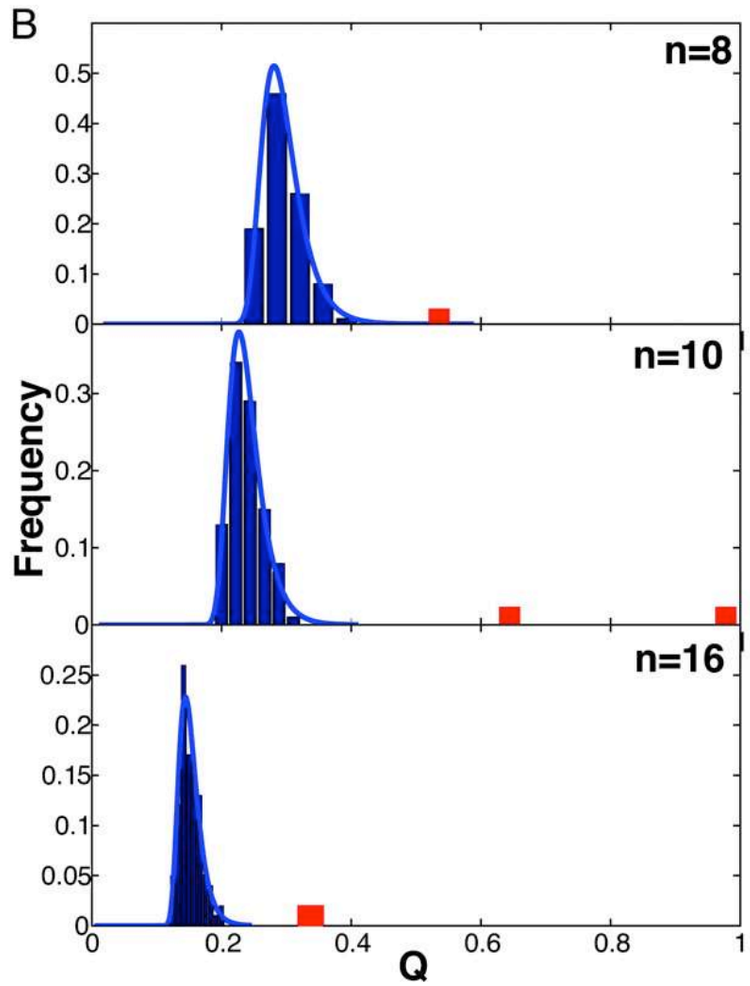
Spirin, Mirny, PNAS 100, 12123 (2003)

Statistical significance of complexes and modules

Distribution of Q of clusters found by the MC search method.

Red bars: original network of protein interactions.

Blue cuves: randomly rewired graphs.



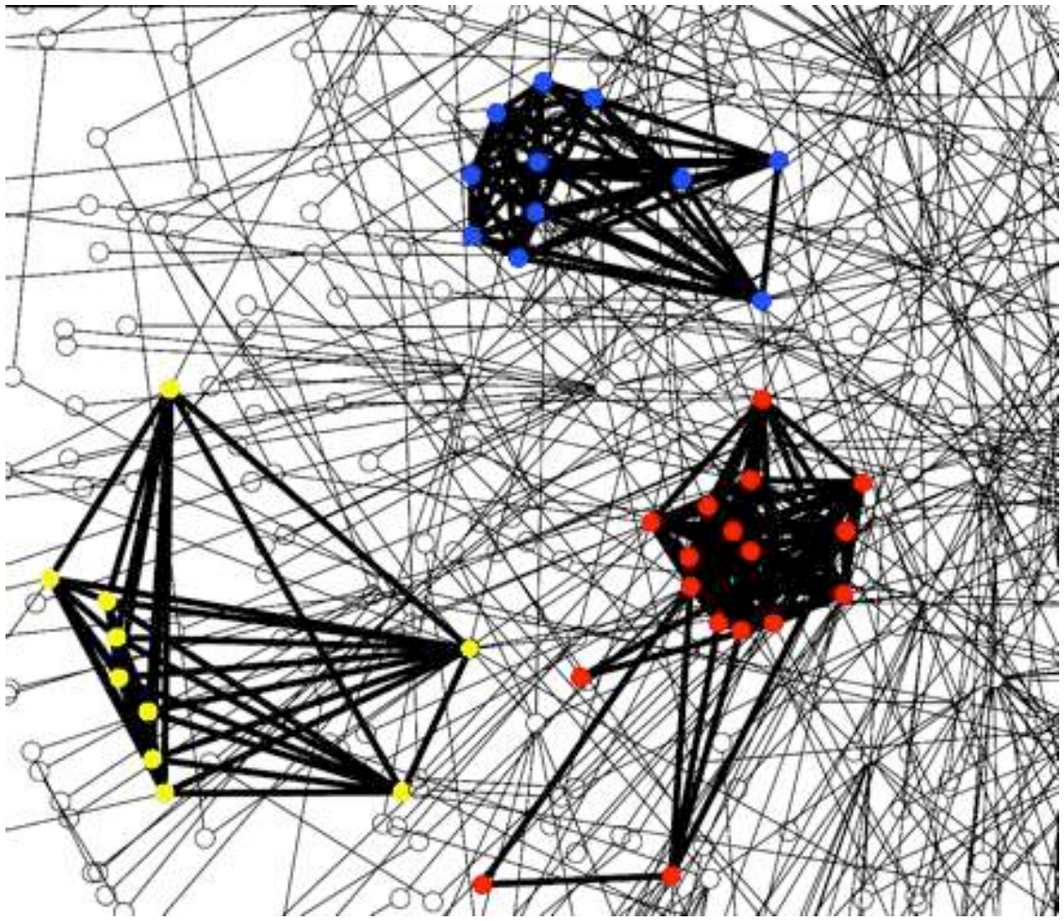
Clusters in the protein network have many more interactions than their counterparts in the random graphs.

Spirin, Mirny, PNAS 100, 12123 (2003)

Architecture of protein network

Fragment of the protein network.
Nodes and interactions in discovered clusters are shown in bold.

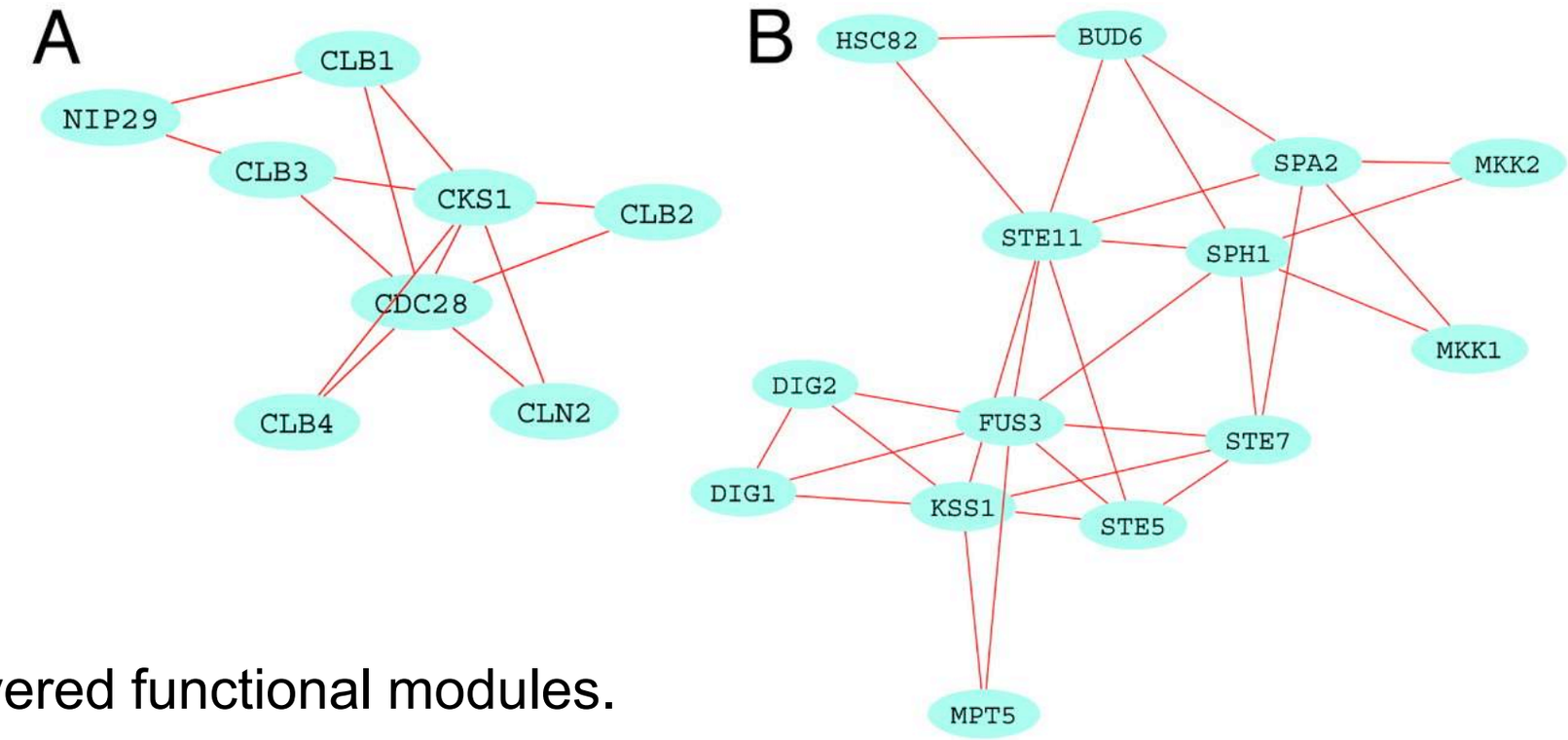
Nodes are colored by functional categories in MIPS:
red, transcription regulation;
blue, cell-cycle/cell-fate control;
green, RNA processing; and
yellow, protein transport.



Complexes shown are the SAGA/TFIID complex (red), the anaphase-promoting complex (blue), and the TRAPP complex (yellow).

Spirin, Mirny, PNAS 100, 12123 (2003)

Discovered functional modules



Examples of discovered functional modules.

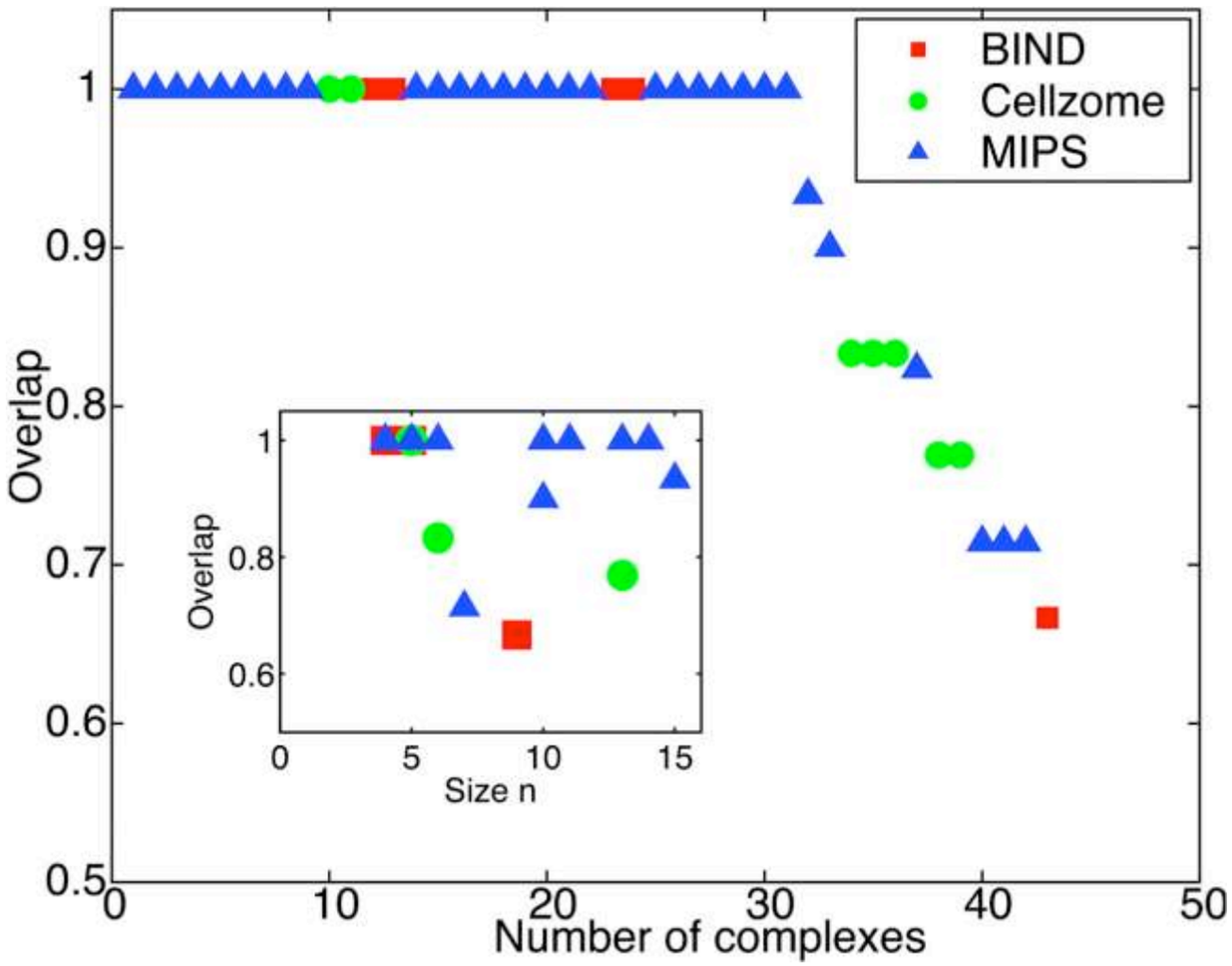
(A) A module involved in cell-cycle regulation. This module consists of cyclins (CLB1-4 and CLN2) and cyclin-dependent kinases (CKS1 and CDC28) and a nuclear import protein (NIP29). Although they have many interactions, these proteins are not present in the cell at the same time.

(*B*) Pheromone signal transduction pathway in the network of protein–protein interactions. This module includes several MAPK (mitogen-activated protein kinase) and MAPKK (mitogen-activated protein kinase kinase) kinases, as well as other proteins involved in signal transduction. These proteins do not form a single complex; rather, they interact in a specific order.

Spirin, Mirny, PNAS 100, 12123 (2003)

Architecture of protein network

Comparison of discovered complexes and modules with complexes derived experimentally (BIND and Cellzome) and complexes catalogued in MIPS.
Discovered complexes are sorted by the overlap with the best-matching experimental complex. The overlap is defined as the number of common proteins divided by the number of proteins in the best-matching experimental complex.
The first 31 complexes match exactly, and another 11 have overlap above 65%.
Inset shows the overlap as a function of the size of the discovered complex. Note that discovered complexes of all sizes match very well with known experimental complexes.
Discovered complexes that do not match with experimental ones constitute our predictions.



Spirin, Mirny, PNAS 100, 12123 (2003)

Robustness of clusters found

Model effect of false positives in experimental data: randomly reconnect, remove or add 10-50% of interactions in network.

Cluster recovery probability as a function of the fraction of altered links .

Black curves : a fraction of links are rewired.

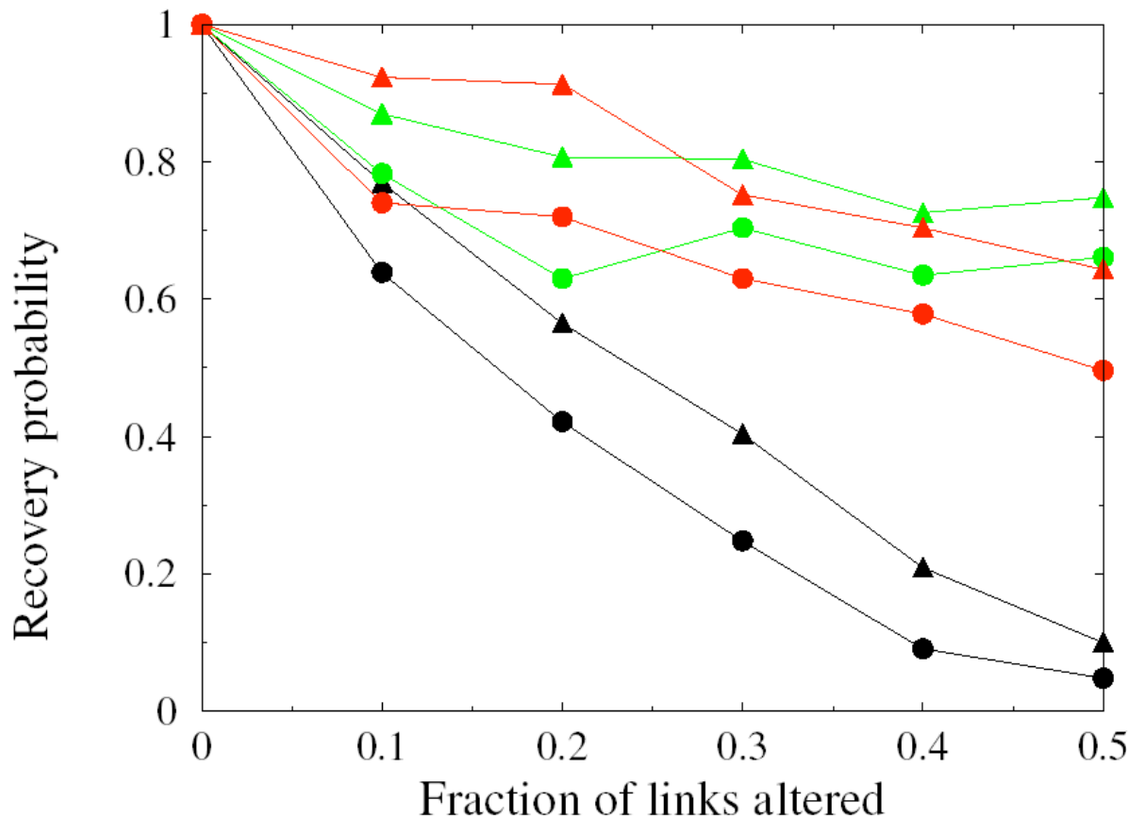
Red, links are removed;

green, links are added.

Circles: probability to recover 75% of the original cluster;

Triangles: probability to recover 50%.

Noise in the form of removal or addions If links has less deteriorating effect than random rewiring. About 75% of clusters can still be found when 10% of links are rewired.



Spirin, Mirny, PNAS 100, 12123 (2003)

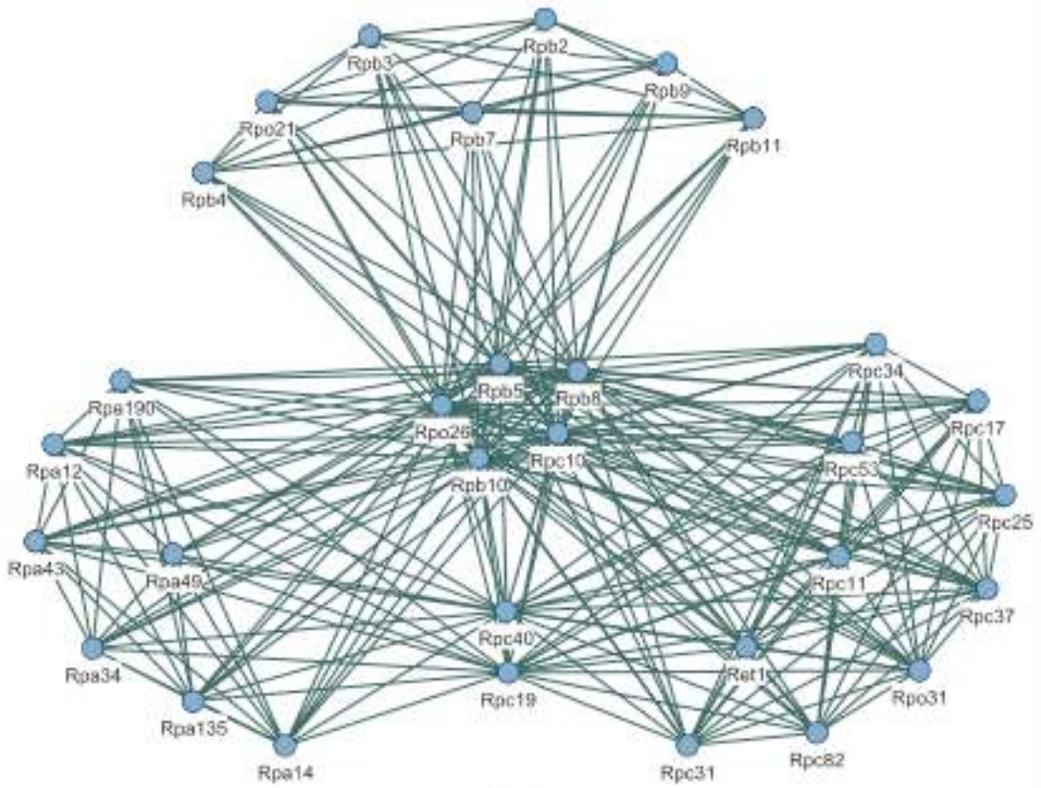
Summary

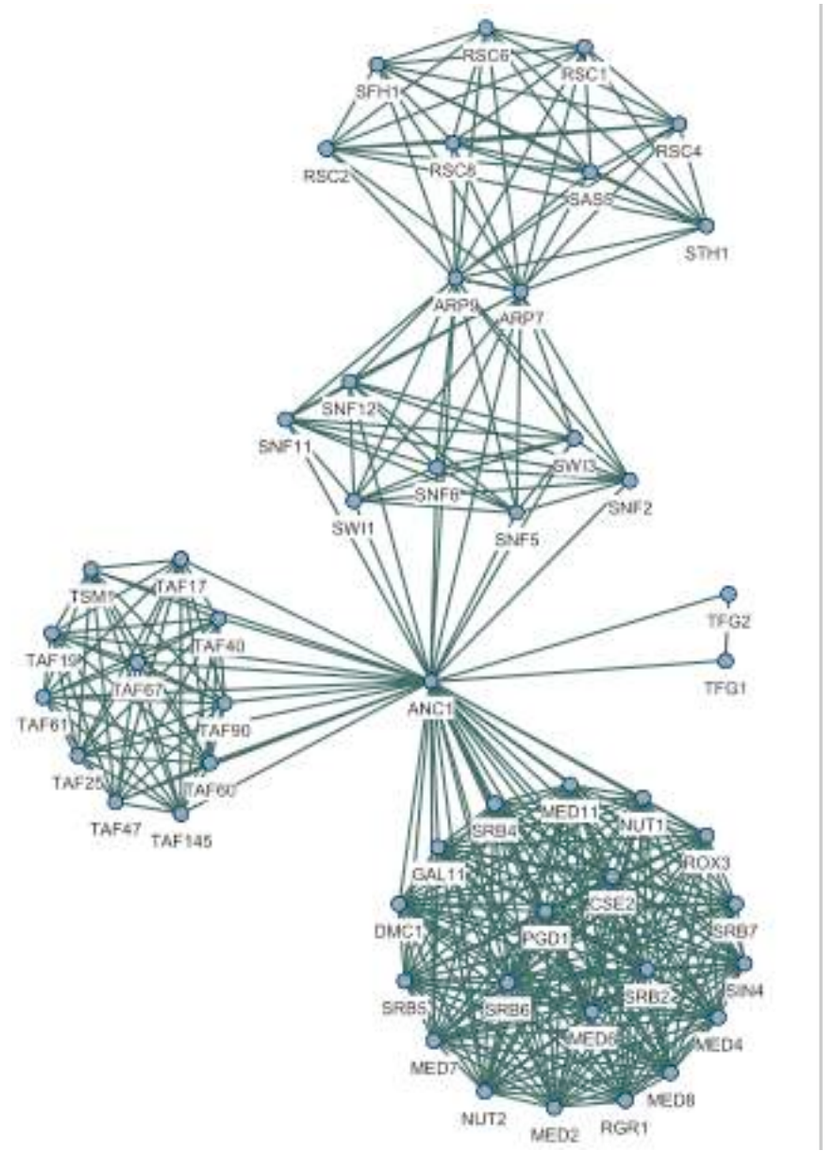
Analysis of meso-scale properties demonstrated the presence of **highly connected clusters** of proteins in a network of protein interactions. Strong support for suggested **modular architecture** of biological networks.

There exist 2 types of clusters: **protein complexes** and **dynamic functional modules**. Both have more interactions among their members than with the rest of the network.

Dynamic modules cannot be purified in experiments because they are not assembled as a complex at any single point in time.
Computational analysis allows detection of such modules by integrating pairwise molecular interactions that occur at different times and places.
However, computational analysis alone does not allow to distinguish between complexes and modules or between transient and simultaneous interactions.

Reducing Network Complexity?





Is there a **representation** that highlights the **structure** of these networks???

- Modular Decomposition (Gagneur, …, Casari, 2004)
- Network Compression (Royer, …, Schröder, 2008)

Method

Open Access

Modular decomposition of protein-protein interaction networks

Julien Gagneur*†, Roland Krause*, Tewis Bouwmeester* and Georg Casari*

Addresses: *Cellzome AG, Meyerhofstrasse 1, 69117 Heidelberg, Germany. †Laboratoire de Mathématiques Appliquées aux Systèmes, Ecole Centrale Paris, Grande Voie des Vignes, 92295 Châtenay-Malabry cedex, France.

Abstract

We introduce an algorithmic method, termed modular decomposition, that defines the organization of protein-interaction networks as a hierarchy of nested modules. Modular decomposition derives the logical rules of how to combine proteins into the actual functional complexes by identifying groups of proteins acting as a single unit (sub-complexes) and those that can be alternatively exchanged in a set of similar complexes. The method is applied to experimental data on the pro-inflammatory tumor necrosis factor-α (TNF-α)/NFκB transcription factor pathway.

Shared Components

Shared components = proteins or groups of proteins occurring in different complexes are fairly common. A shared component may be a small part of many complexes, acting as a **unit** that is constantly **reused** for its function.

Also, it may be the **main part** of the complex e.g. in a family of variant complexes that differ from each other by distinct proteins that provide functional specificity.

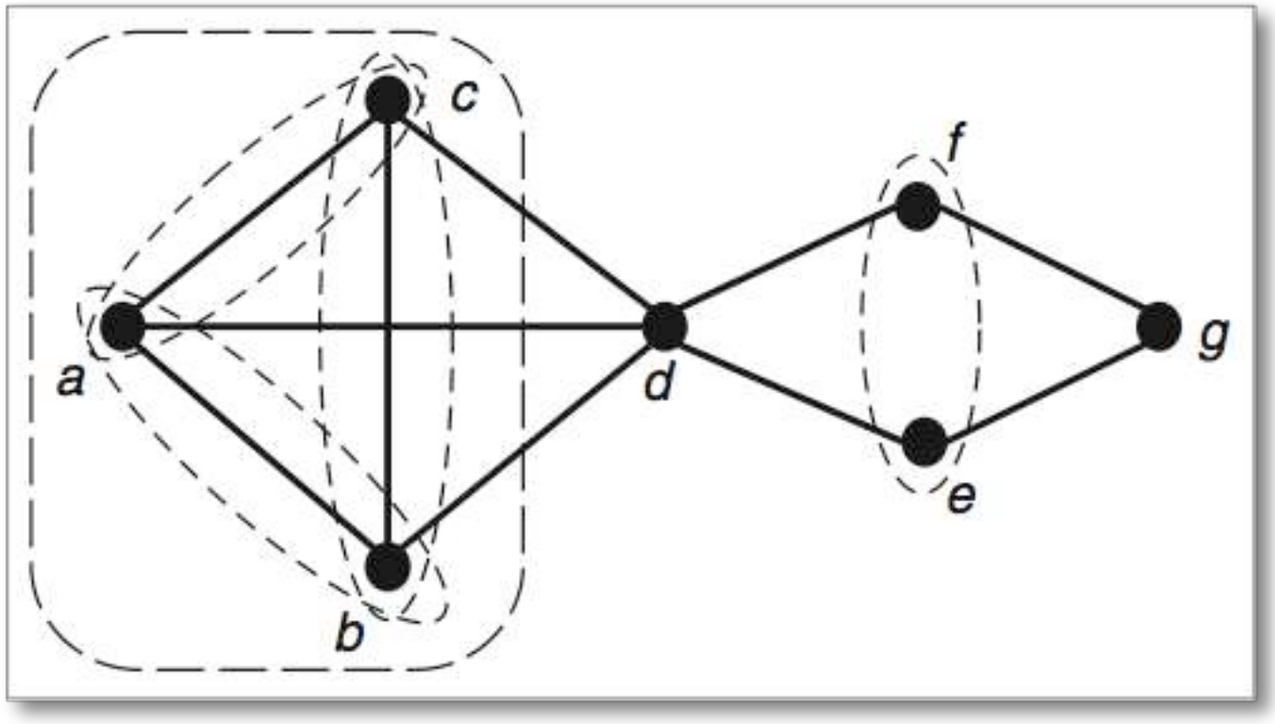
Aim: **identify** and properly **represent** the modularity of protein-protein interaction networks by identifying the **shared components** and the way they are arranged to generate **complexes**.

Georg Casari, Cellzome (Heidelberg)

Gagneur et al. Genome Biology 5, R57 (2004)

Modules in a Graph

Module := set of **nodes** that have the **same neighbors** outside of the module



trivial modules:

{a}, {b}, …, {g}
{a, b, …, g}

non-trivial modules:

{a, b}, {a, c}, {b, c}
{a, b, c}
{e, f}

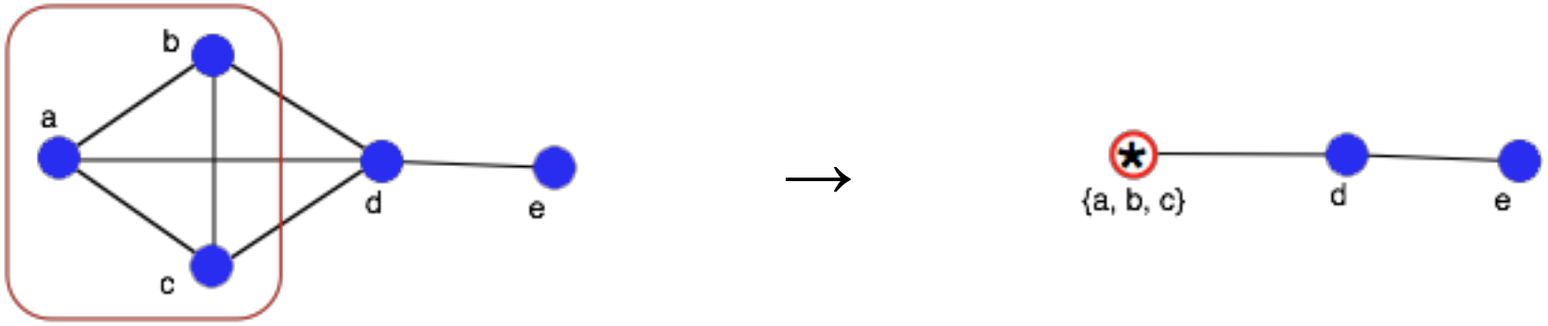
Quotient: representative node for a module

Iterated quotients → labeled tree representing the original network
→ "**modular decomposition**"

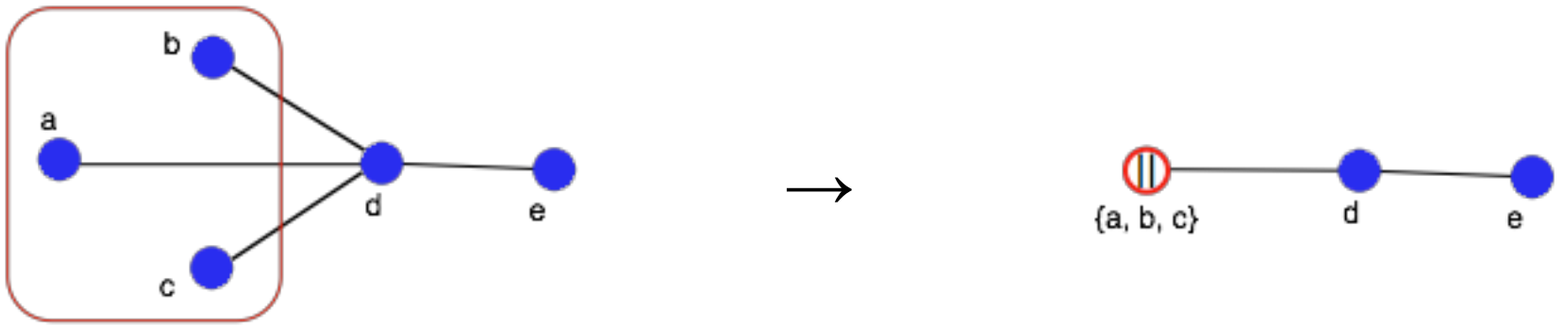
Gagneur et al, *Genome Biology* **5** (2004) R57

Quotients

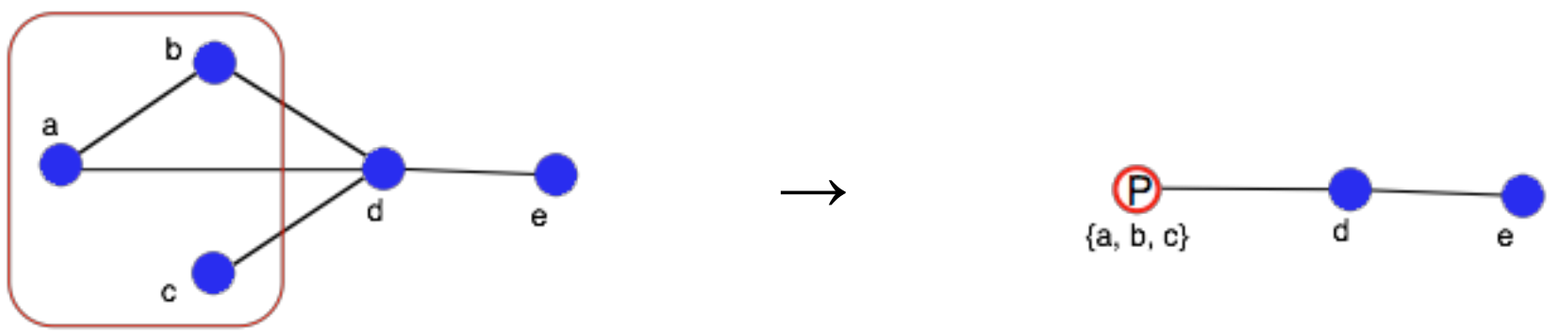
Series: all included nodes are direct **neighbors** (= **clique**)



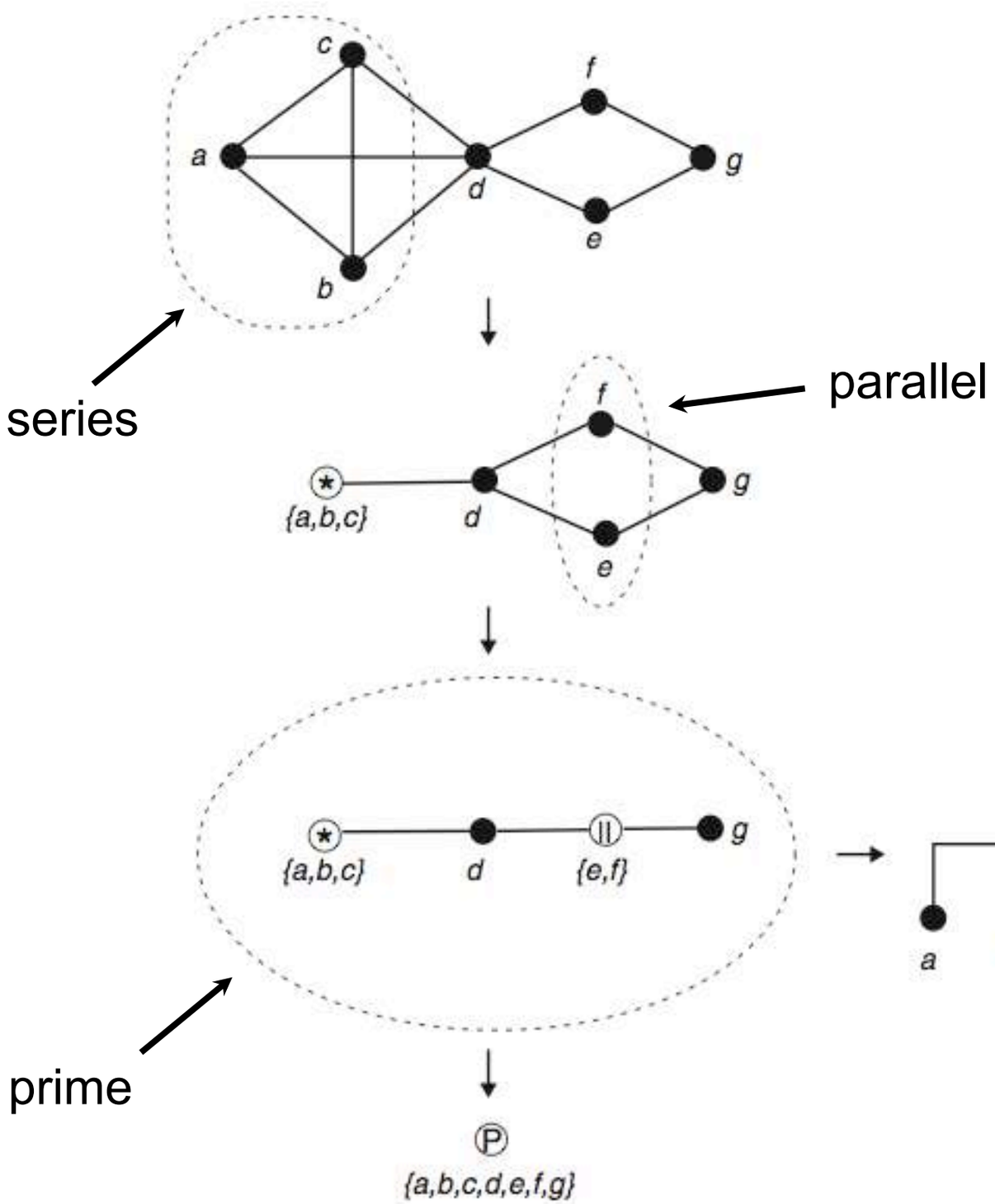
Parallel: all included nodes are **non-neighbors**



Prime: "anything else" (best labeled with the actual structure)



A Simple Recursive Example



Gagneur et al, *Genome Biology* **5** (2004) R57

Results from protein complex purifications (PCP), e.g. TAP

Different types of data:

• Y2H: detects direct physical interactions between proteins

• PCP by tandem affinity purification with mass-spectrometric identification of the protein components identifies multi-protein complexes

→ Molecular decomposition will have a **different meaning** due to different **semantics** of such graphs.

Here, focus analysis on PCP content. PCP experiment: select bait protein where TAP-label is attached

→ Co-purify protein with those proteins that co-occur in at least one complex with the bait protein.

Gagneur et al. Genome Biology 5, R57 (2004)

Data from Protein Complex Purification

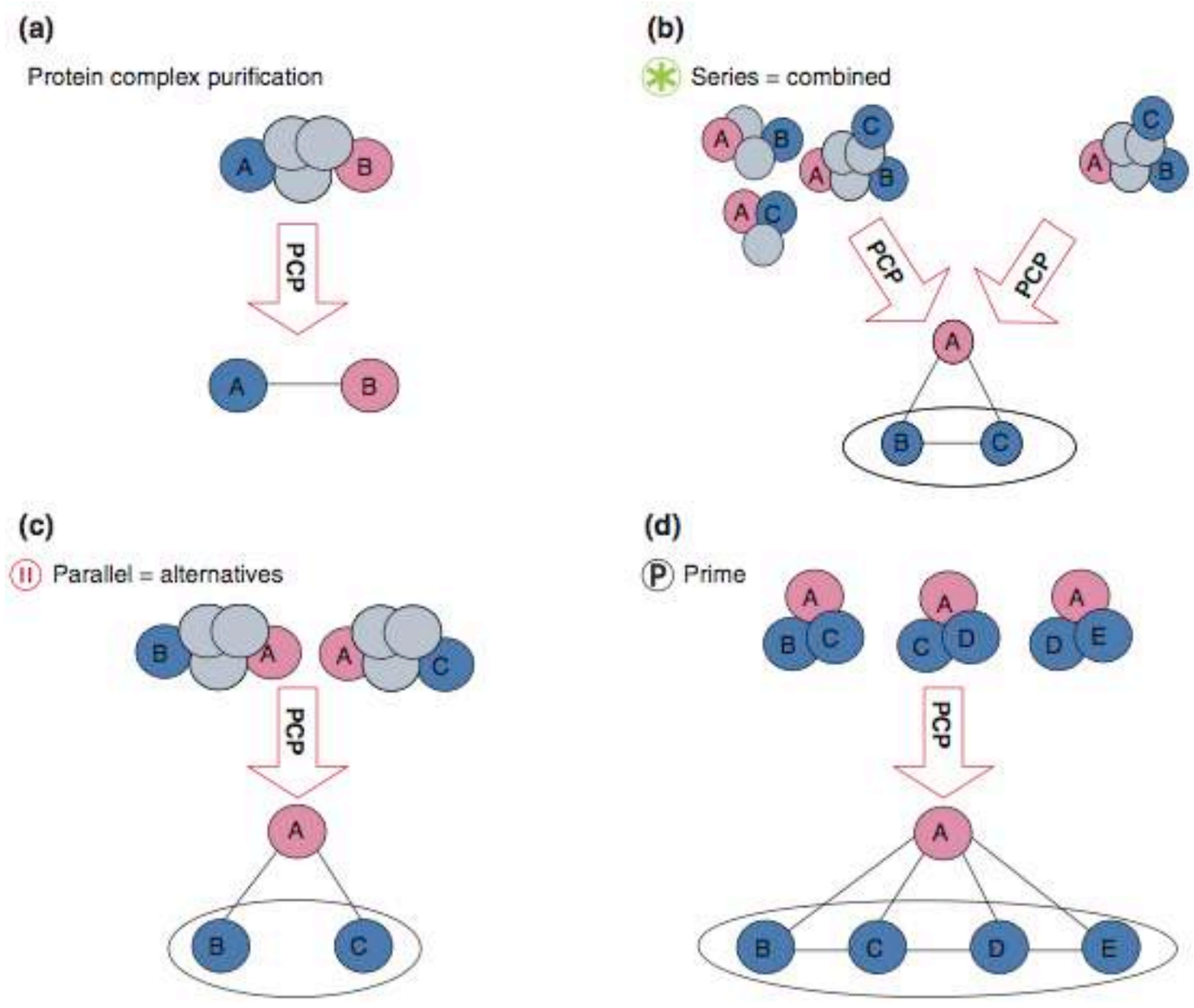
Graphs and module labels from systematic PCP experiments:

(a) Two neighbors in the network are proteins occurring in a same complex.

(b) Several potential sets of complexes can be the origin of the same observed network. Restricting interpretation to the simplest model (top right), the **series** module reads as a logical AND between its members.

(c) A module labeled ´**parallel**´ corresponds to proteins or modules working as strict alternatives with respect to their common neighbors.

(d) The ´**prime**´ case is a structure where none of the two previous cases occurs.



Gagneur et al. Genome Biology 5, R57 (2004)

Real World Examples

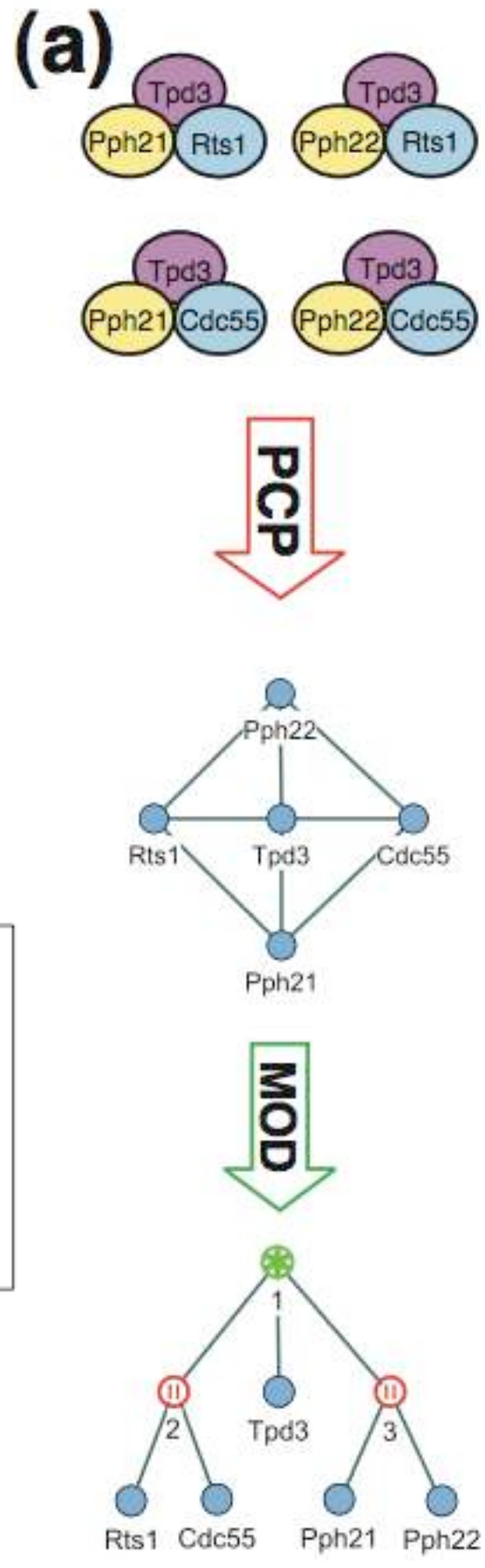
Two examples of modular decompositions of protein-protein interaction networks.

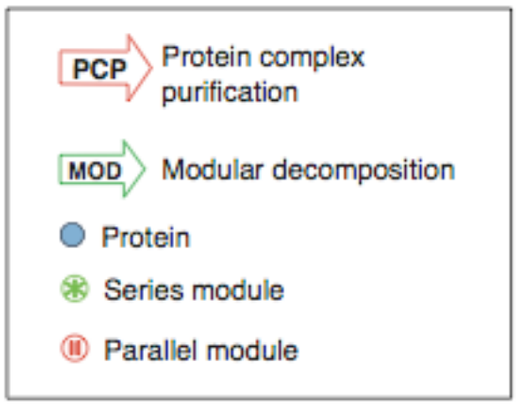
In each case from top to bottom: schemata of the complexes, the corresponding protein-protein interaction network as determined from PCP experiments, and its modular decomposition (MOD).

(a) Protein phosphatase 2A.

Parallel modules group proteins that do not interact but are functionally equivalent.

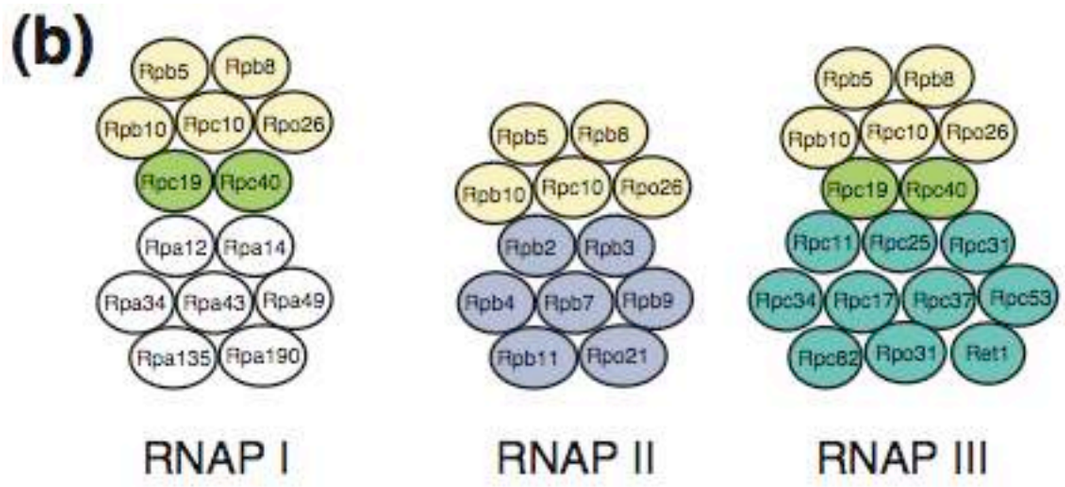
Here these are the catalytic Pph21 and Pph22 (module 2) and the regulatory Cdc55 and Rts1 (module 3), connected by the Tpd3 „backbone“.

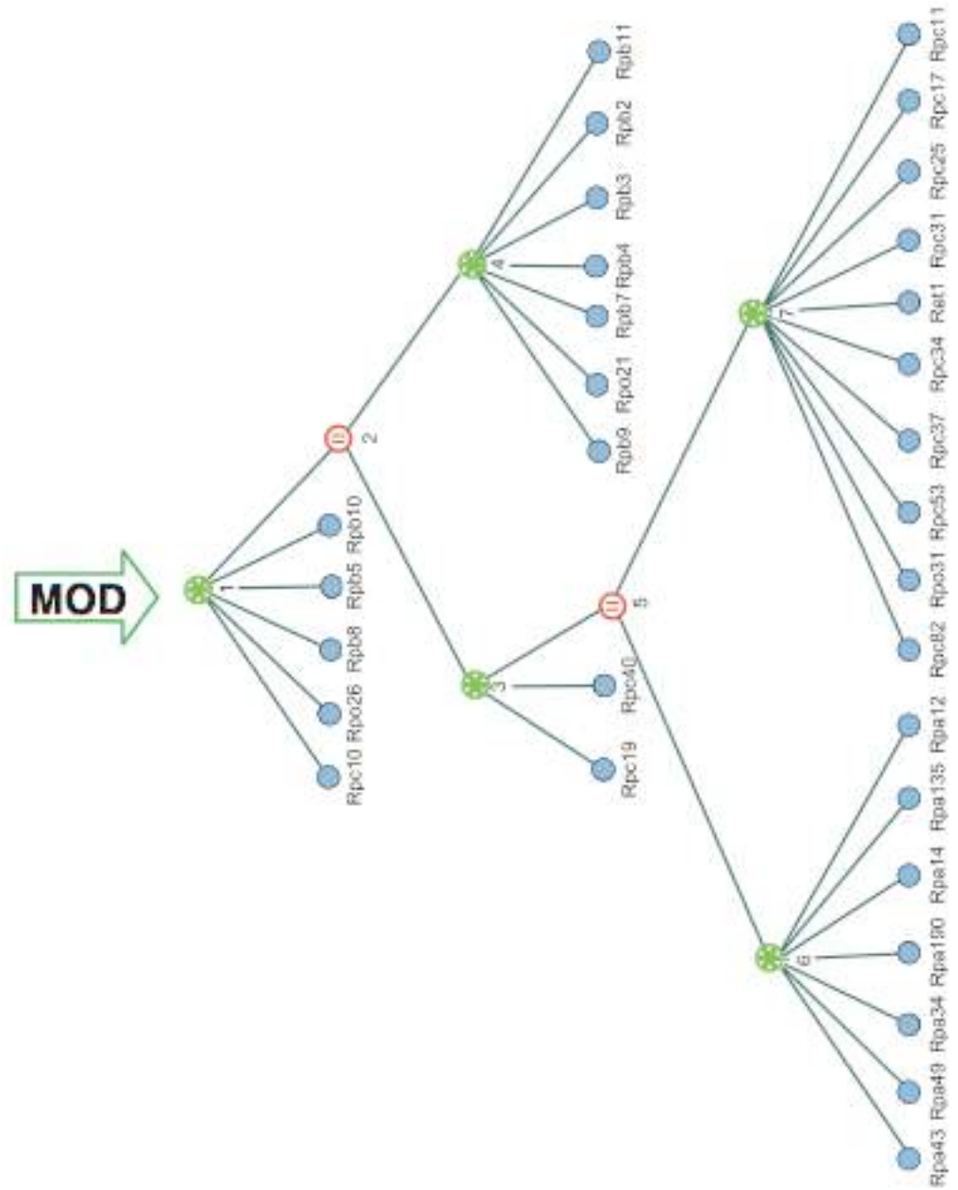


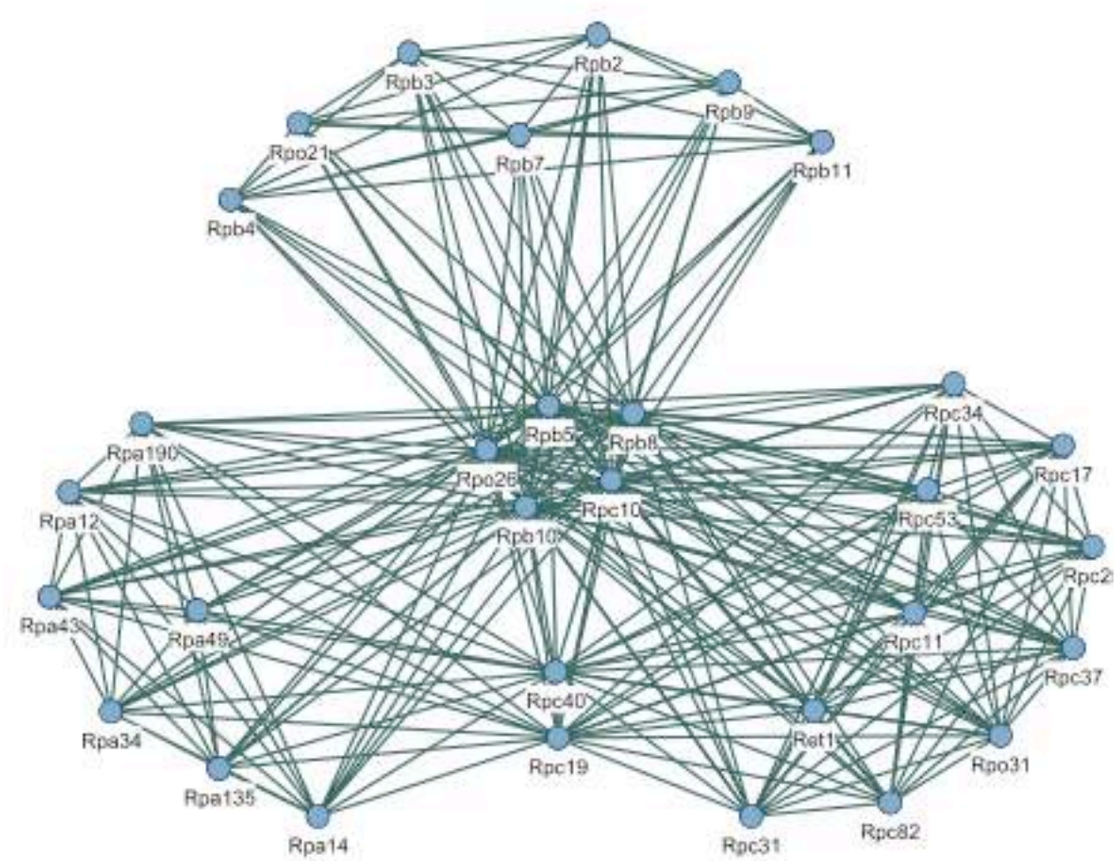


Notes: • Graph does not show functional alterantives!!!
• other decompositions also possible

RNA polymerases I, II and III







Again: modular decompositon easier to comprehend than graph

Gagneur et al. Genome Biology 5, R57 (2004)

Summary

Modular decomposition of graphs is a **well-defined concept**.

• One can proof thoroughly for which graphs a modular decomposition exists.

• Efficient *O(m + n)* algorithms exist to compute the decomposition.

However, experiments have shown that **biological** complexes are **not strictly disjoint**. They often share components

→ separate complexes do not always fulfill the strict requirements of modular graph decomposition.

Also, there exists a „danger“ of false-positive or false-negative interactions.

→ **other methods**, e.g., for detecting communities (Girven & Newman) or clusters (Spirin & Mirny) are **more suitable** for identification of **complexes** because they are more sensitive.

Power Graph Analysis

OPEN ACCESS Freely available online

PLoS COMPUTATIONAL BIOLOGY

Unraveling Protein Networks with Power Graph Analysis

Loïc Royer, Matthias Reimann, Bill Andreopoulos, Michael Schroeder*

Biotechnology Center, Technische Universität Dresden, Germany

PLoS Comp Biol **4** (2008) e1000108

Lossless compact abstract representation of graphs:

- **Power nodes** = set of nodes (criterion for grouping?)
- **Power edges** = edges between power nodes

Exploit observation that **cliques** and bi-cliques are **abundant** in real networks
→ **explicitly** represented in power graphs

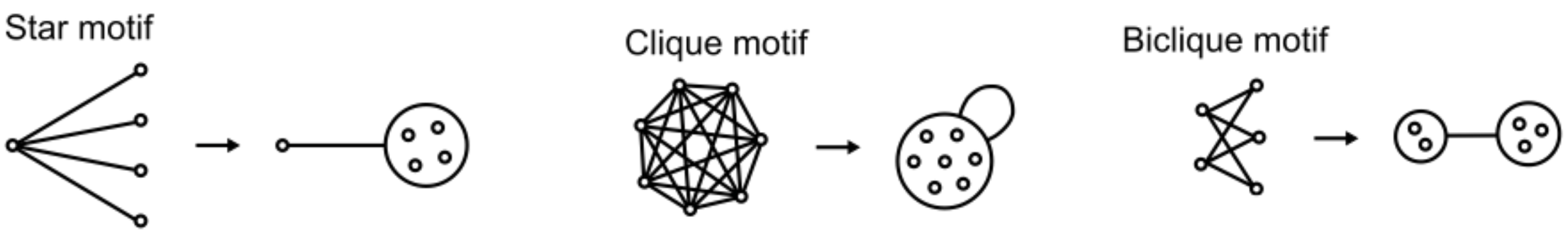
Power Nodes

In words: "… if two **power nodes** are **connected** by a power edge in G', this means in G that **all nodes** of the first power node are **connected to all nodes** of the second power node.

Similarly, if a power node is connected to itself by a power edge in G', this signifies that all nodes in the power node are connected to each other by edges in G"

With: "real-world" graph G = {V, E}

power graph G' = {V', E'}



Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Power Graph Analysis Algorithm

Two **conditions**:

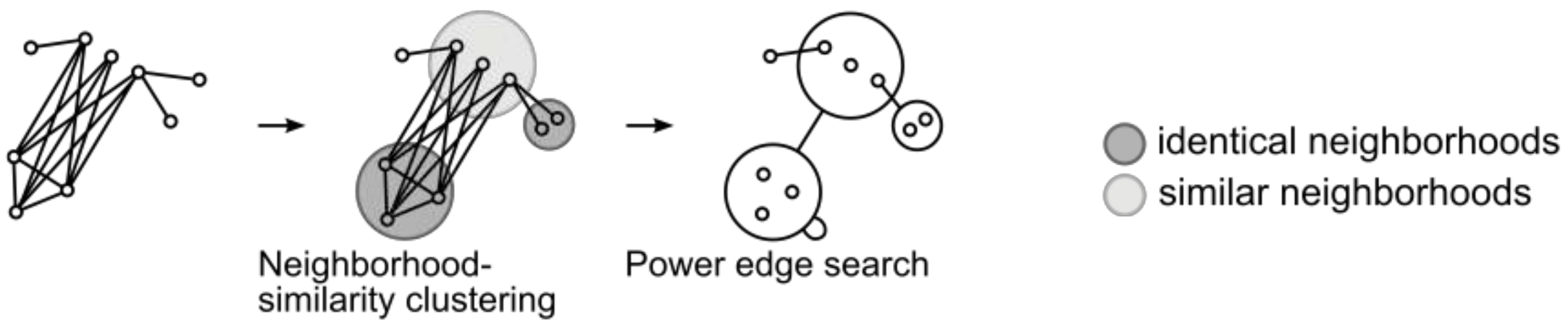
• power **node hierarchy** condition:
two power nodes are either disjoint, or one is included in the other

• power **edge disjointness** condition: each edge of the original graph is represented by one and only one power edge

Algorithm:

1) identify potential power nodes with hierarchical clustering based on neighborhood similarity

2) greedy power edge search



Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Complex = Star or Clique?

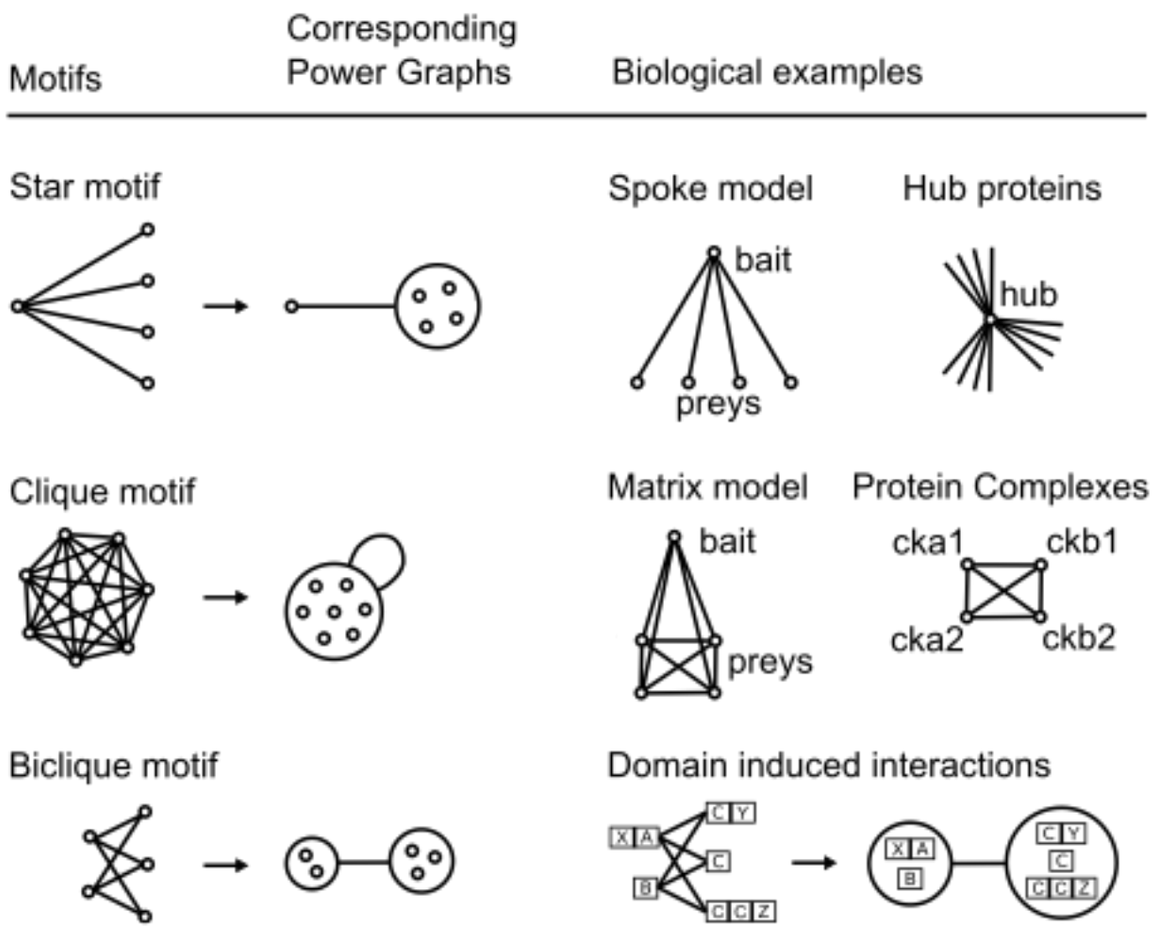


Figure 1. The Three Basic Motifs: Star, Biclique, and Clique. Stars often occur because of hub proteins or when affinity purification complexes are interpreted using the spoke model. Bicliques often arise because of domain-domain or domain-motif interactions inducing protein interactions [25]. Power nodes are sets of nodes and power edges connect power nodes. A power edge between two power nodes signifies that all nodes of the first set are connected to all nodes of the second set. Note that nodes within a power node are not necessarily connected to each other.
doi:10.1371/journal.pcbi.1000108.g001

In **pull-down** experiments:
Bait is used to capture **complexes** of prey proteins
→ do they all just stick to the bait or to each other?

spoke model
→ **underestimates** connectivity

matrix model
→ **overestimates** connectivity

Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Casein Kinase II Complex

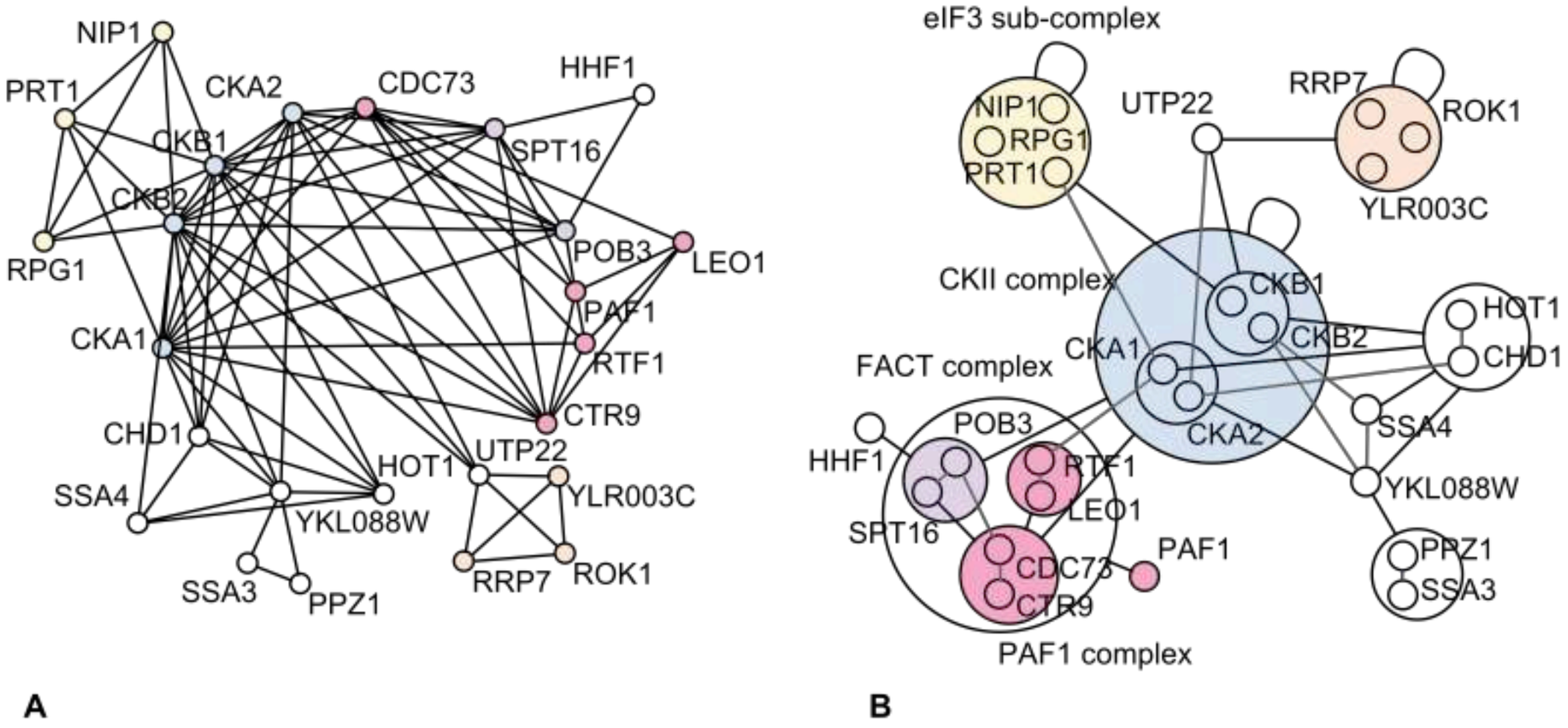


Figure 2. Casein Kinase II Complex. Two catalytic alpha subunits (CKA1, CKA2) and two regulatory beta subunits (CKB1, CKB2) interacting with the FACT complex, with sub-complex NIP1-RPG-PRT1, and with the PAF1 complex. The graph representation (A) consists of 80 edges whereas the power graph representation (B) has 30 power edges, thus an edge reduction of 62%. This simplification of the representation makes the separation of the regulatory subunits from the catalytic subunits immediately apparent without loss of information on individual interactions.
doi:10.1371/journal.pcbi.1000108.g002

→ Power graph: compressed and cleaner representation

Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Various Similarities

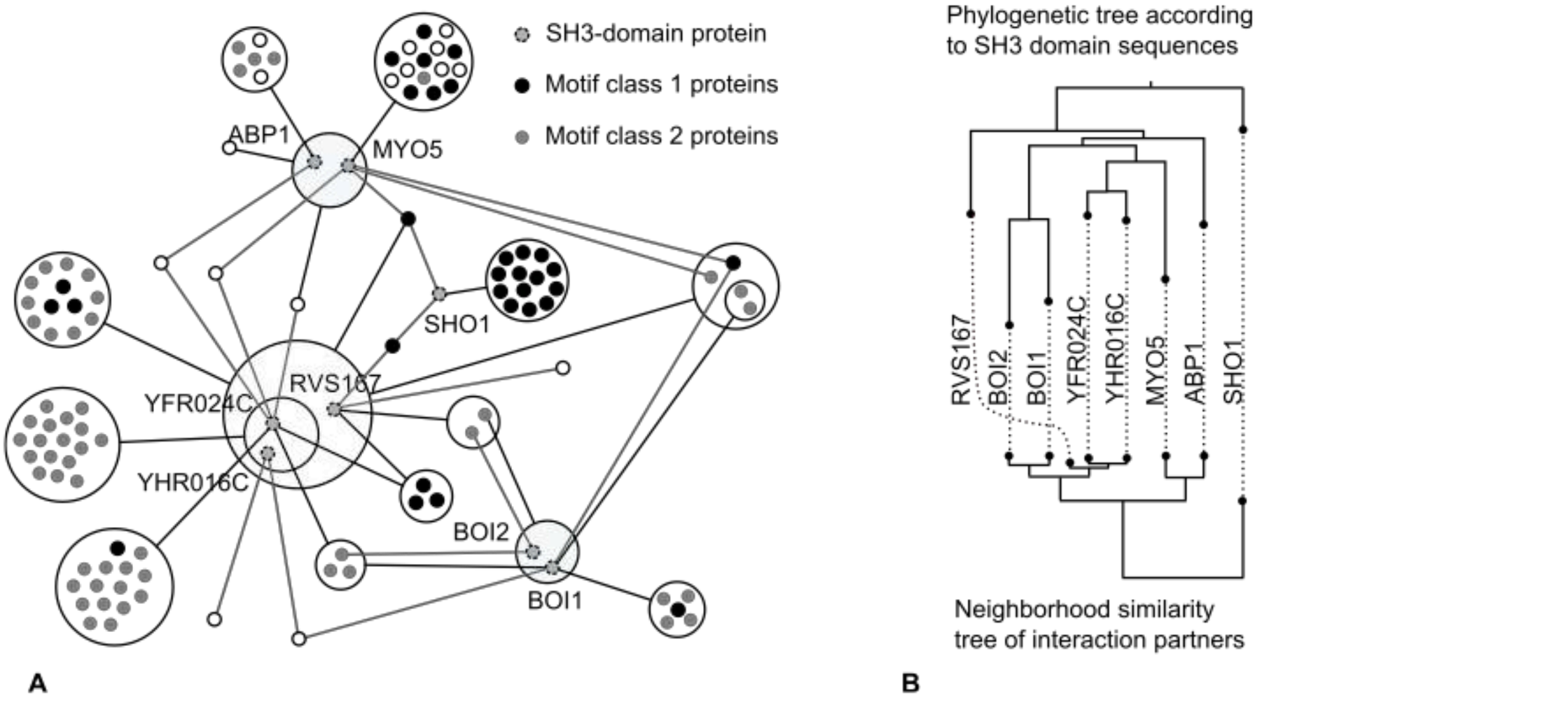


Figure 4. Interactions of SH3 Carrying Proteins. (A) Protein interaction network showing the 105 interaction partners of the SH3 domain carrying proteins: SHO1, ABP1, MYO5, BOI1, BOI2, RVS167, YHR016C and YFR024. The underlying network consists of 182 interactions represented here as 36 power edges–a reduction of 80%–leaving all but only the core information. Class 1 motif (RxxPxxP) proteins are shown in black. Class 2 motif (PxxPxR) proteins are shown in light grey [15]. Note how power graphs group proteins having similar binding motifs together. (B) Phylogeny and interaction profiles. Comparison of the phylogenetic tree of the SH3 domains sequences with the neighbourhood similarity tree of interaction partners. The neighbourhood similarity implied by the power graph reflects the sequence similarity of the SH3 domains.
doi:10.1371/journal.pcbi.1000108.g004

Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Network Compression

Power graph analysis: group nodes with **similar neighborhood**
→ often **functionally** related proteins end up in one power node

Lossless compression
of graphs:
38…**85% edge reduction**
for biological networks

Protein Interaction Network	# Nodes	# Edges	Avg. Degree	e.r.	c.r
Lim et al. (2006) [46]	571	701	2.45	85%	12.1
Hazbun et al. (2003) [47]	2243	3130	2.79	79%	13
Kim et al. (2006) [48]	577	1090	3.78	67%	4.1
Gunsalus et al. (2004) [49]	281	514	3.6	65%	4.6
Gavin et al. (2006) [4]	1462	6942	9.4	64%	7.2
Ewing et al. (2007) [50]	2294	6449	5.62	54%	6.6
Ito et al. (2001) [51]	3243	4367	2.69	53%	5.3
Rual et al. (2005) [12]	1527	2529	3.31	50%	4.5
Krogan et al. (2006) [6]	2708	7123	5.26	49%	4.5
Stanyon et al. (2004) [9]	478	1778	7.43	48%	5.3
Stanyon et al. (2004) [9]	478	1778	7.43	48%	5.3
Butland et al. (2005) [52]	1277	5324	8.33	43%	6.0
Arifuzzaman et al. (2006) [53]	2457	8663	7.05	39%	5.4
Lacount et al. (2005) [13]	1272	2643	4.16	38%	3.8

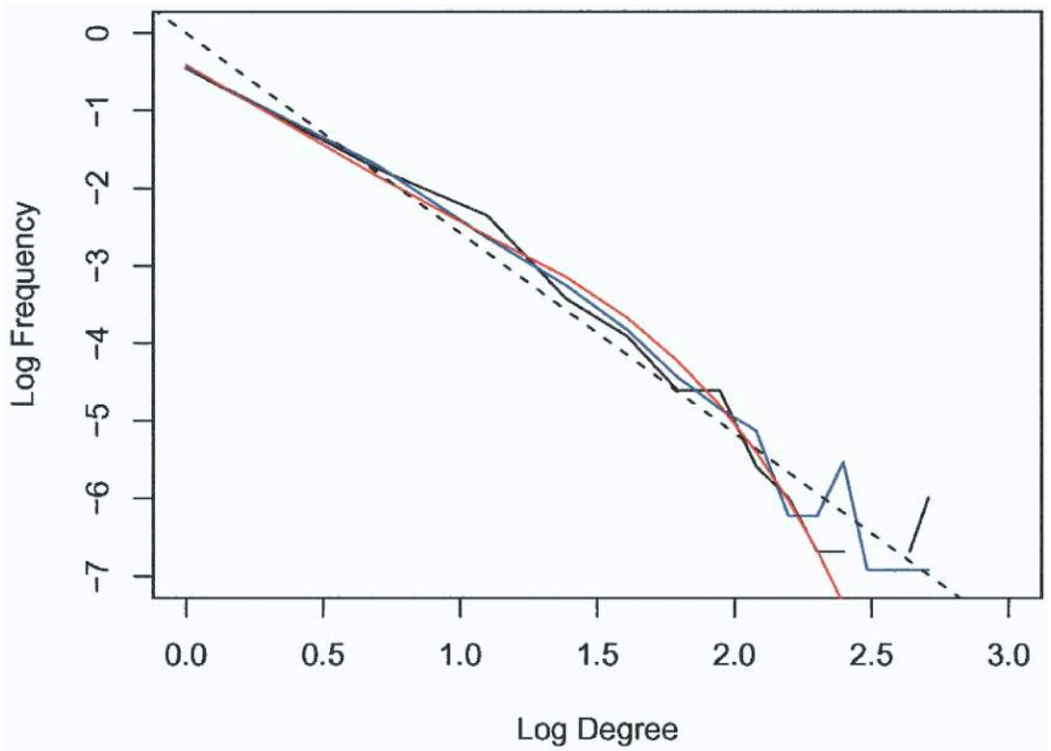
Average degree, edge reduction (e.r.), and edge to power node conversion rate (c.r.).
doi:10.1371/journal.pcbi.1000108.t001

Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

Some PPI Networks

For some time: "**Biological** networks are **scale-free**…"

Y2H PPI network from Uetz etal, *Nature* **403** (2003) 623



P(k) compared to a power law

→ **Tutorial 3**: PPI networks for various species

However, there are some doubts… → next lecture

Summary

What you learned **today**:

- Network **robustness**
  - scale-free networks are failure-tolerant, but fragile to attacks
    <=> the few **hubs** are important
  - => immunize hubs!

- **Modules** in networks
  - => modular decomposition
  - => power graph analysis

Next lecture:

- **Short Test #1: Fri, Nov. 2**
- Are biological networks scale-free? (other models?)
- Network growth mechanisms