Bioinformatics 3

V 5 – Robustness and Modularity

Fri, Nov 4, 2011

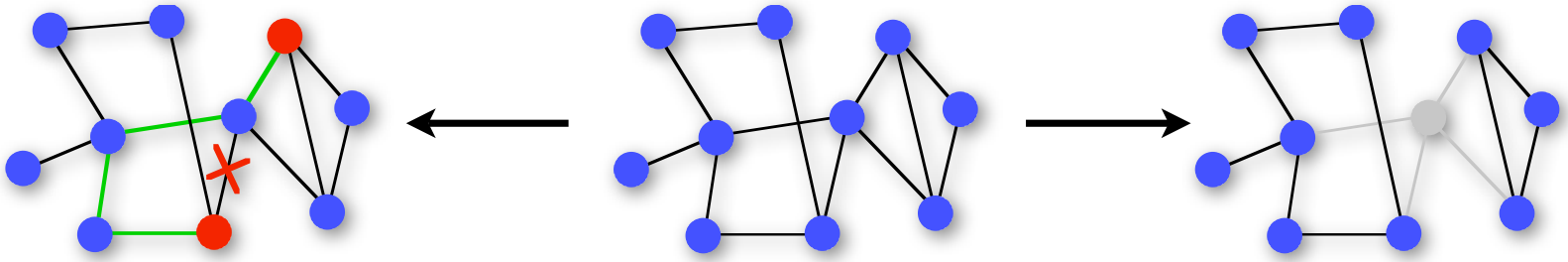
Network Robustness

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

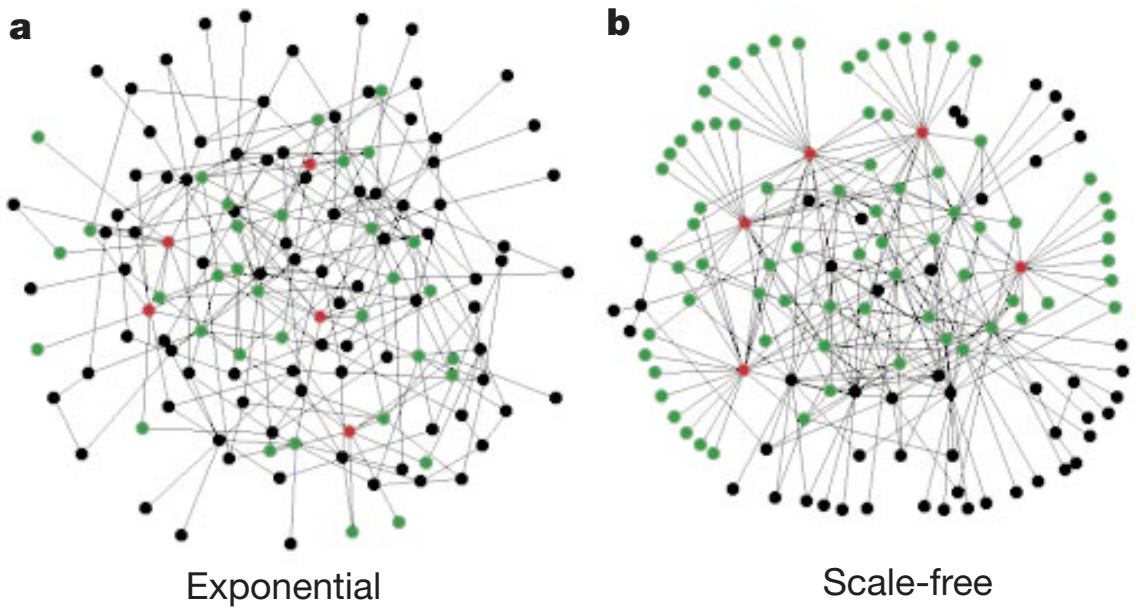
Réka Albert, Hawoong Jeong & Albert-László Barabási

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

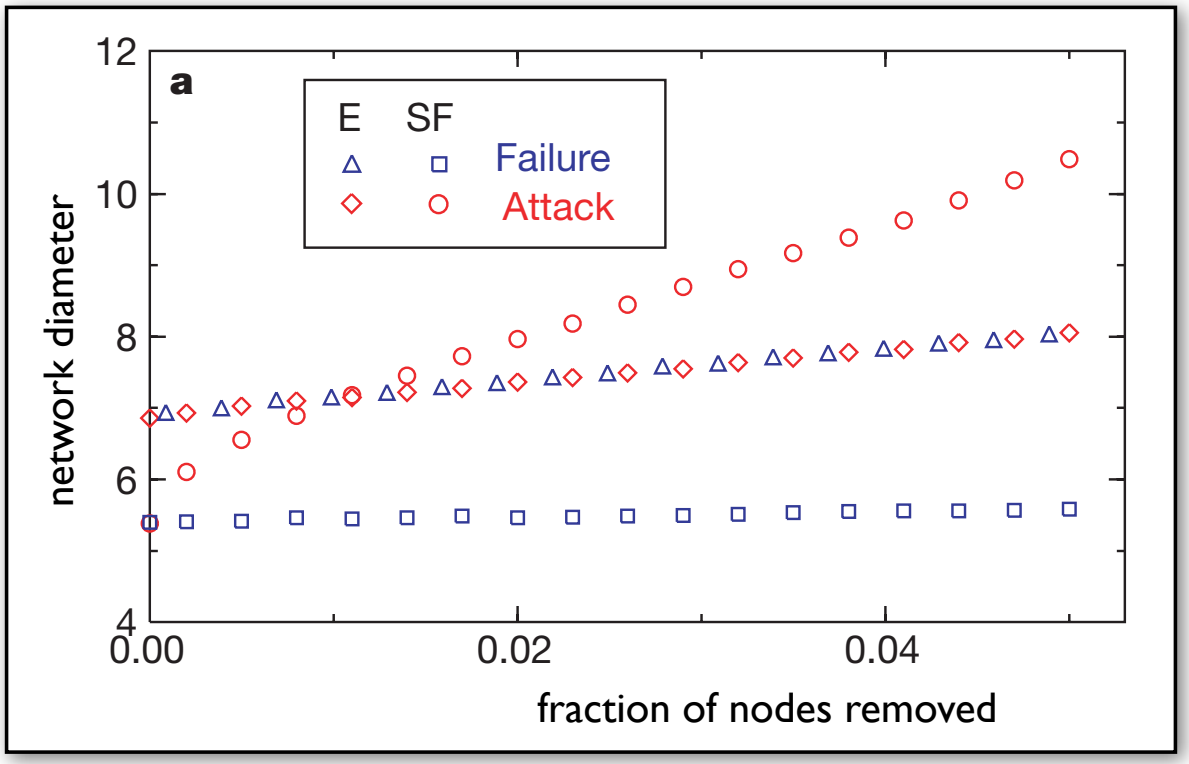
… 60% of the network

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

Failure vs. Attack

Failure: remove **randomly** chosen nodes

Attack: remove nodes with highest **degrees**

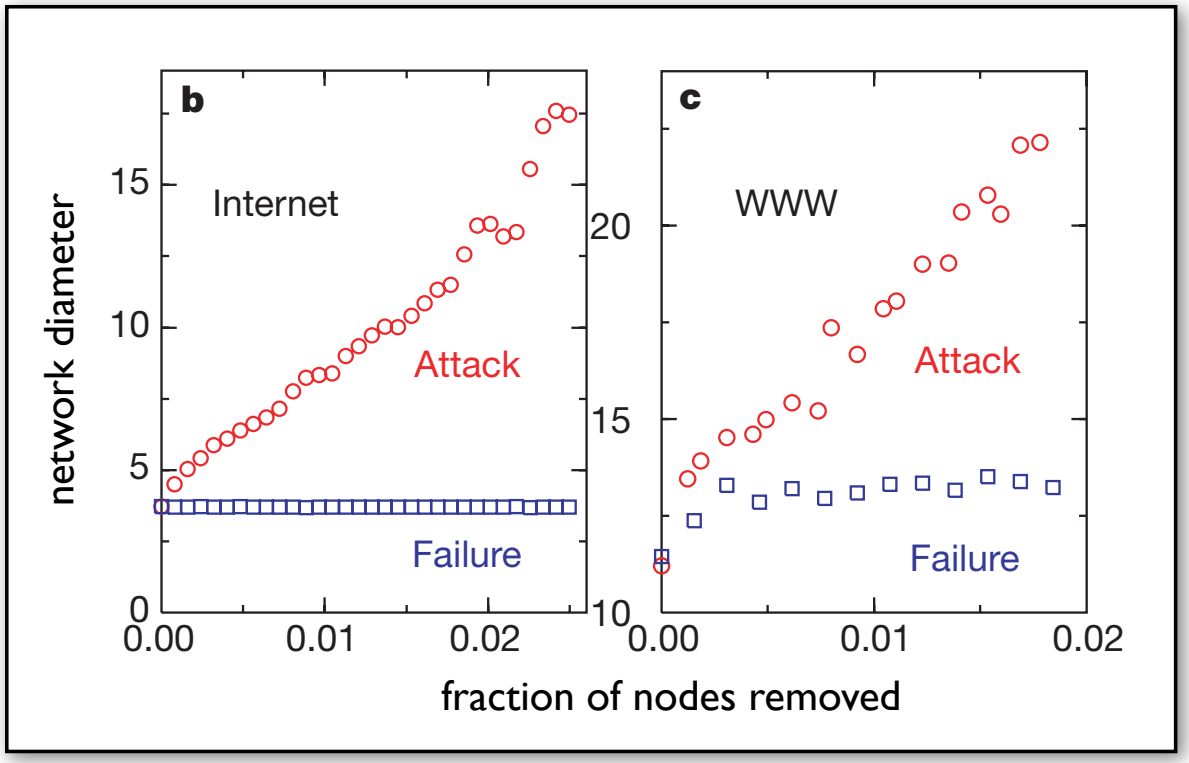


N = 10000, L = 20000, but effect is size-independent

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

Two VINs

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

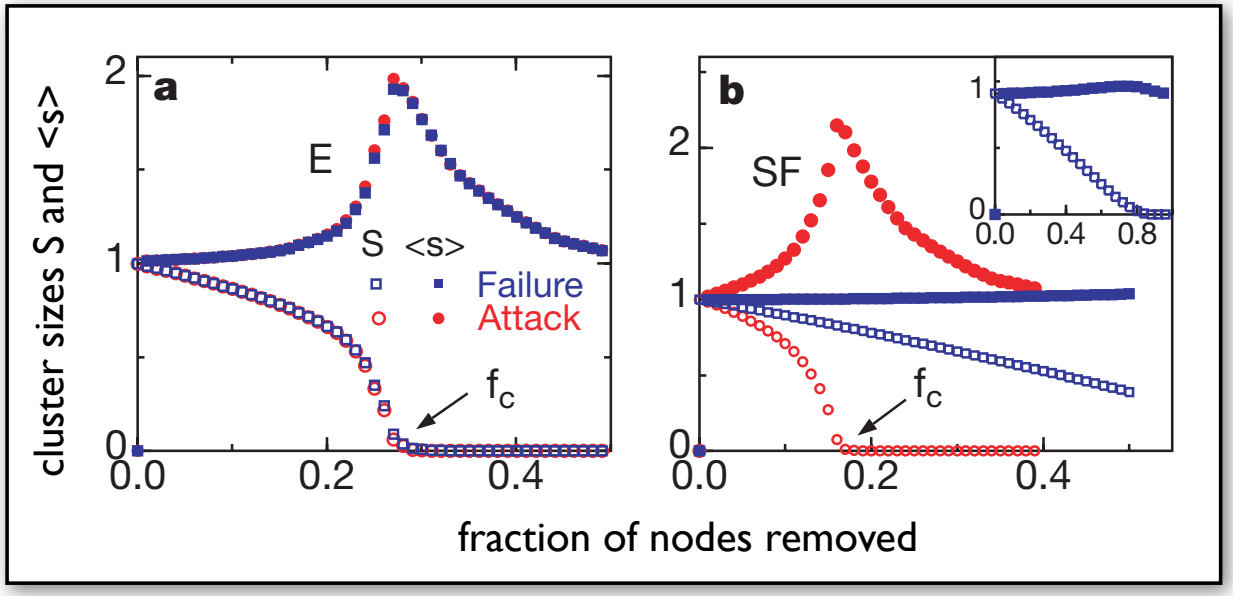


http://moat.nlanr.net/Routing/rawdata/ : 6209 nodes and 12200 links (2000)

WWW-sample containing 325729 nodes and 1498353 links

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

Network Fragmentation



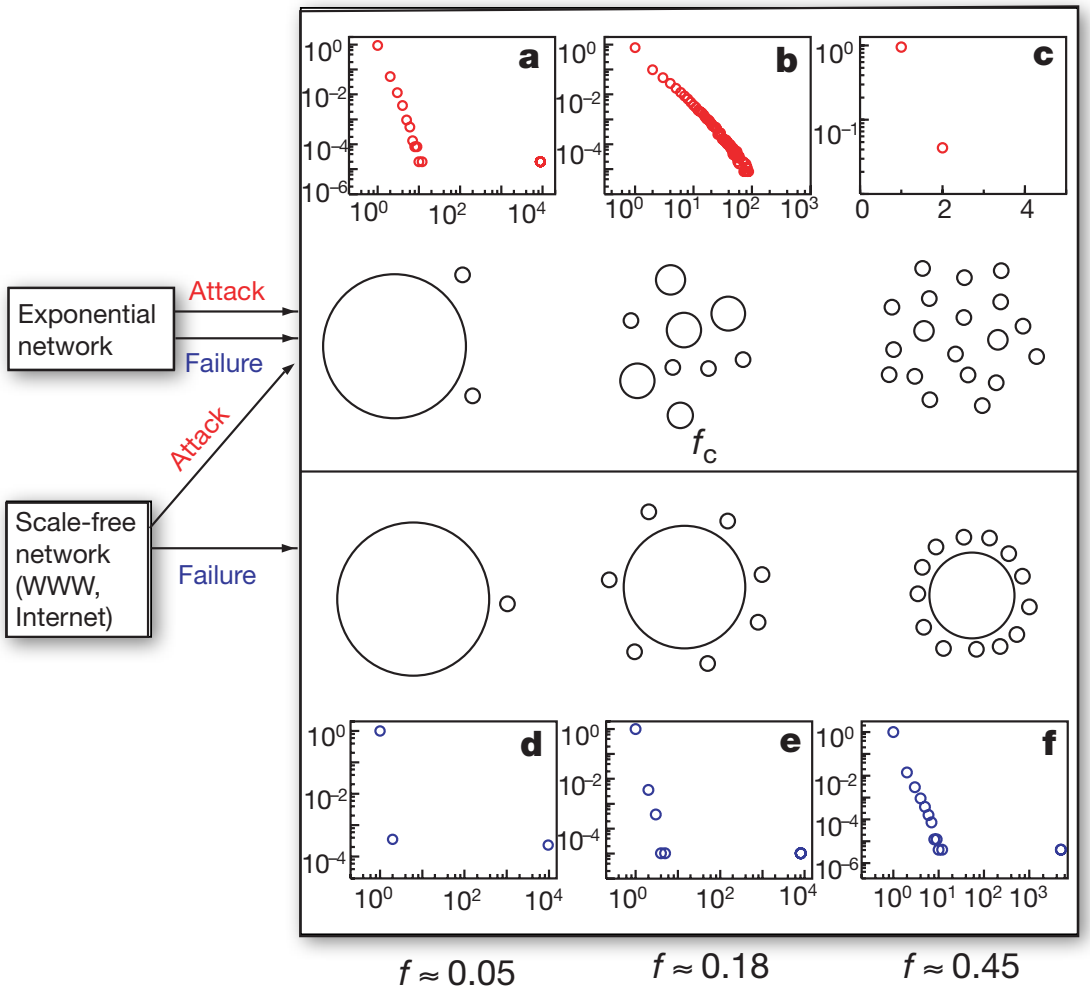
Relative size of the largest clusters S

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 $f_c \approx 0.18$

Cluster Size Distributions



Many biological networks are robust against failures and undirected attacks (environmental influences)

⇕

A drug against one enzyme can knock out a complete pathway.

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

Disease Spreading

Susceptible–infected–susceptible (**SIS**) model for **disease propagation**:

- node = individual (susceptible or infected)
- edge = physical interaction that allows for disease propagation

Propagation:

- susceptible nodes get infected with probability "λ x number of infected neigbors"
- infected nodes recover with probability 1

=> steady state fraction ρ of infected nodes?

For a random network with $k_i \approx \langle k \rangle = 2L/N$:

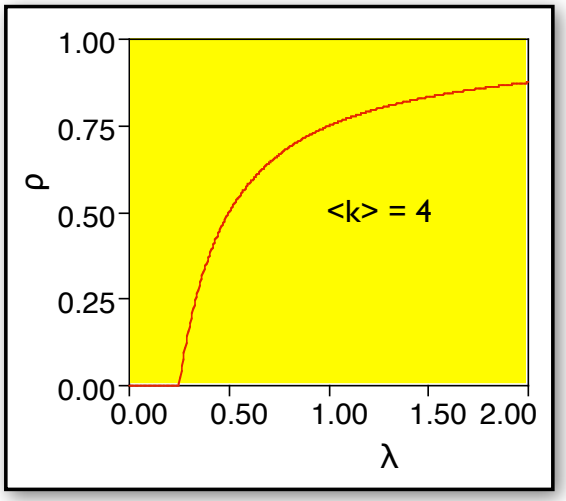
$$\frac{d\rho}{dt} = -\rho(t) + \lambda(1-\rho(t))\langle k \rangle \rho(t) \stackrel{!}{=} 0$$

Epidemic Threshold

For a random network – **stationary prevalence** ρ(t):

$$\frac{d\rho}{dt} = -\rho(t) + \lambda(1-\rho(t))\langle k\rangle\rho(t) \stackrel{!}{=} 0$$

dynamic balance at steady state:
# of recovered = # of newly infected



Solution:

$$\rho(\lambda) = \begin{cases} 0 & \text{for } \lambda \le \lambda_c = 1/\langle k\rangle \\ 1 - \frac{1}{\lambda\langle k\rangle} & \text{for } \lambda > \lambda_c \end{cases}$$

For $\lambda \le \lambda_c$: disease vanishes, all members are healthy
For $\lambda > \lambda_c$: finite prevalence (of infected nodes)

=> λ_c = epidemic threshold

Disease in a Scale-free Network

Now: approximation of k ≈ <k> breaks down,
look at **prevalence $\rho_k(t)$** for degree k

$$\frac{d\rho_k(t)}{dt} = -\rho_k(t) + \lambda k[1-\rho_k(t)] \sum_{k'} P(k'|k)\rho_{k'}(t)$$

Stationary solution => epidemic threshold $\lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle}$

In SF network: max. degree k_c grows with networks size N as

$$k_c \sim N^{1/(\gamma-1)} \quad \text{when} \quad P(k) \sim Ck^{-\gamma}$$

Thus: $\langle k \rangle = \frac{2L}{N}$ $\langle k^2 \rangle = \frac{1}{N}\sum (k_i - \langle k \rangle)^2 \rightarrow \infty$

λ_c => 0 <=> there are always some infected nodes in an infinite SF network
=> diseases never completely die out

Vaccination Strategies

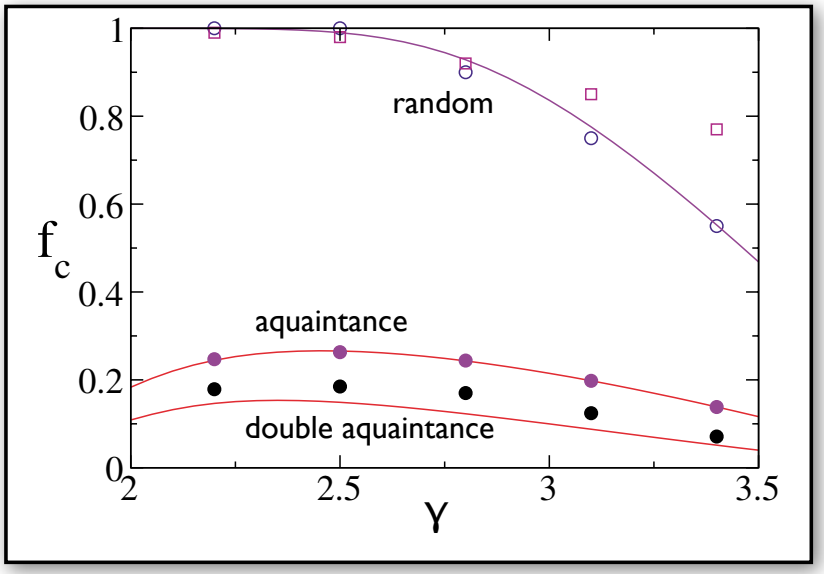
Outbreak of a disease (**pandemia**) is stopped, if the network is **fragmented**

- no infection outside connected component(s)
- finite λ_c in finite SF network

=> most efficient **immunization strategy** to arrest virus?
(Susceptible – Infected – Removed - model)

Fraction f_c of the nodes that has to be immunized to fragment a SF network with $P(k) \sim k^{-\gamma}$ for:

- random immunization
- aquaintance immunization



points: simulations with $N = 10^6$

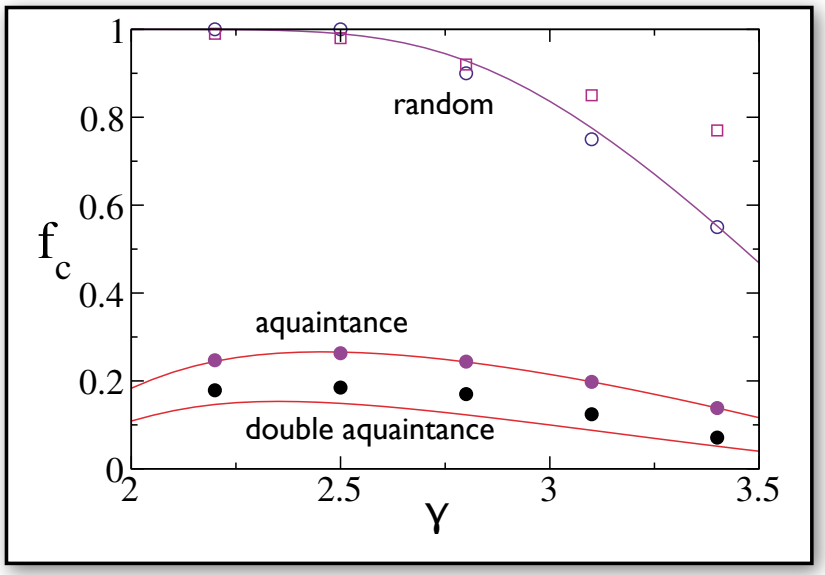
Cohen et al, *PRL* **91** (2003) 247901

Network Breakdown

Random immunization: choose nodes randomly for immunization (removal)
=> "**failure**"

Aquaintance immunization:
choose nodes randomly, immunize one of their neighbors
=> **hubs** are chosen preferentially
=> "**attack**" with local algorithm
(no global knowledge required)

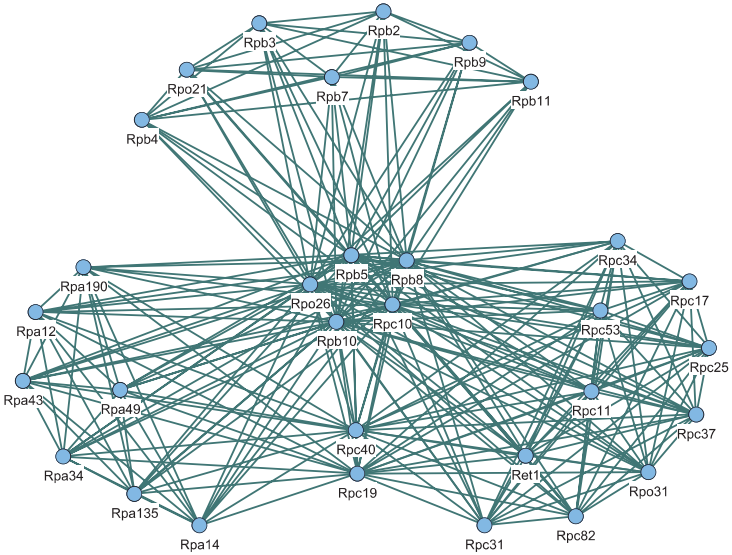
Multiple aquaintance
immunization:
choose set of nodes randomly,
immunize common aquaintances
=> **hubs** are even better identified

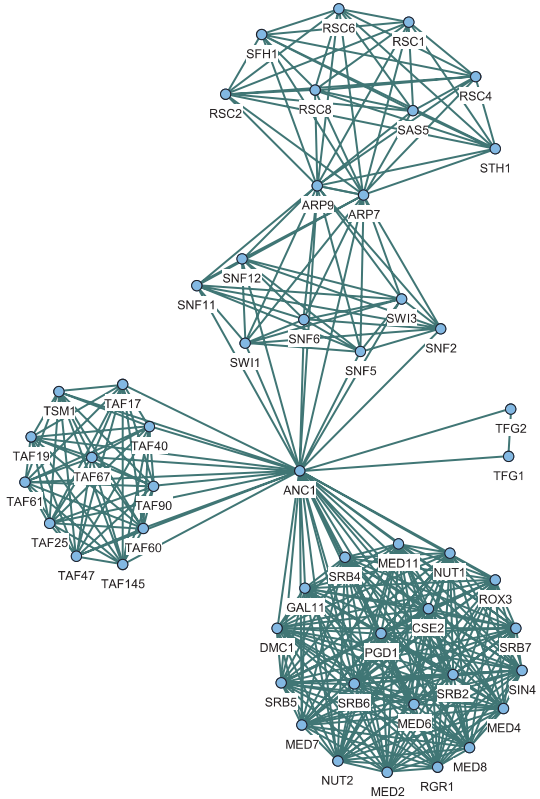


points: simulations with N = 10^6

Cohen et al, *PRL* **91** (2003) 247901

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.

Genome Biology **5** (2004) R57

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.

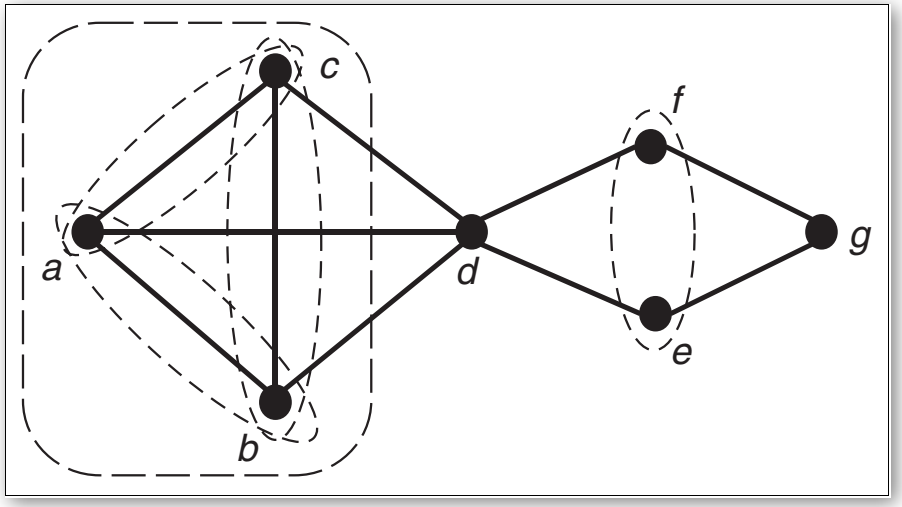
Aim: **identify** and properly **represent** the modularity of protein-protein interaction networks by identifying the **shared components** and the way they are arranged to generate **complexes**.

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}

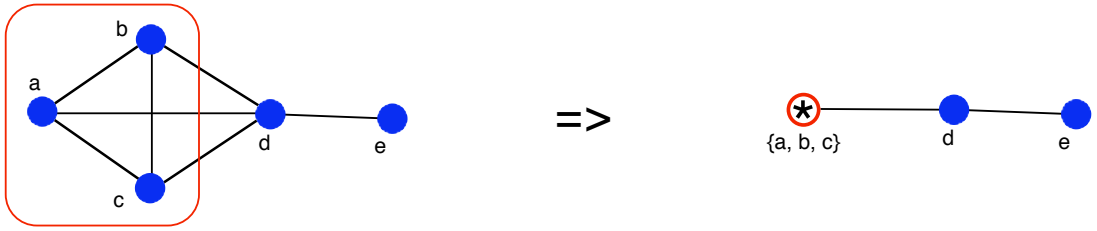
Quotient: representative node for a module

Iterated quotients => labelled tree representing the original network
=> "**modular decomposition**"

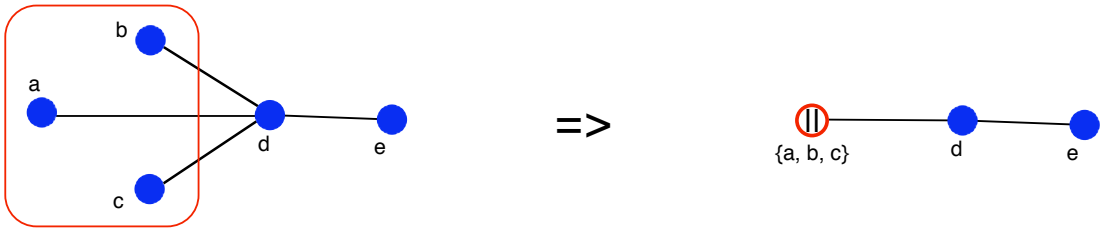
Gagneur et al, *Genome Biology* **5** (2004) R57

Quotients

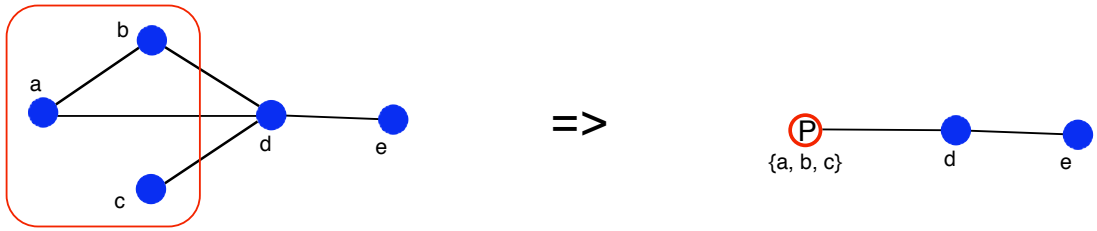
Series: all included nodes are direct neighbors (= **clique**)



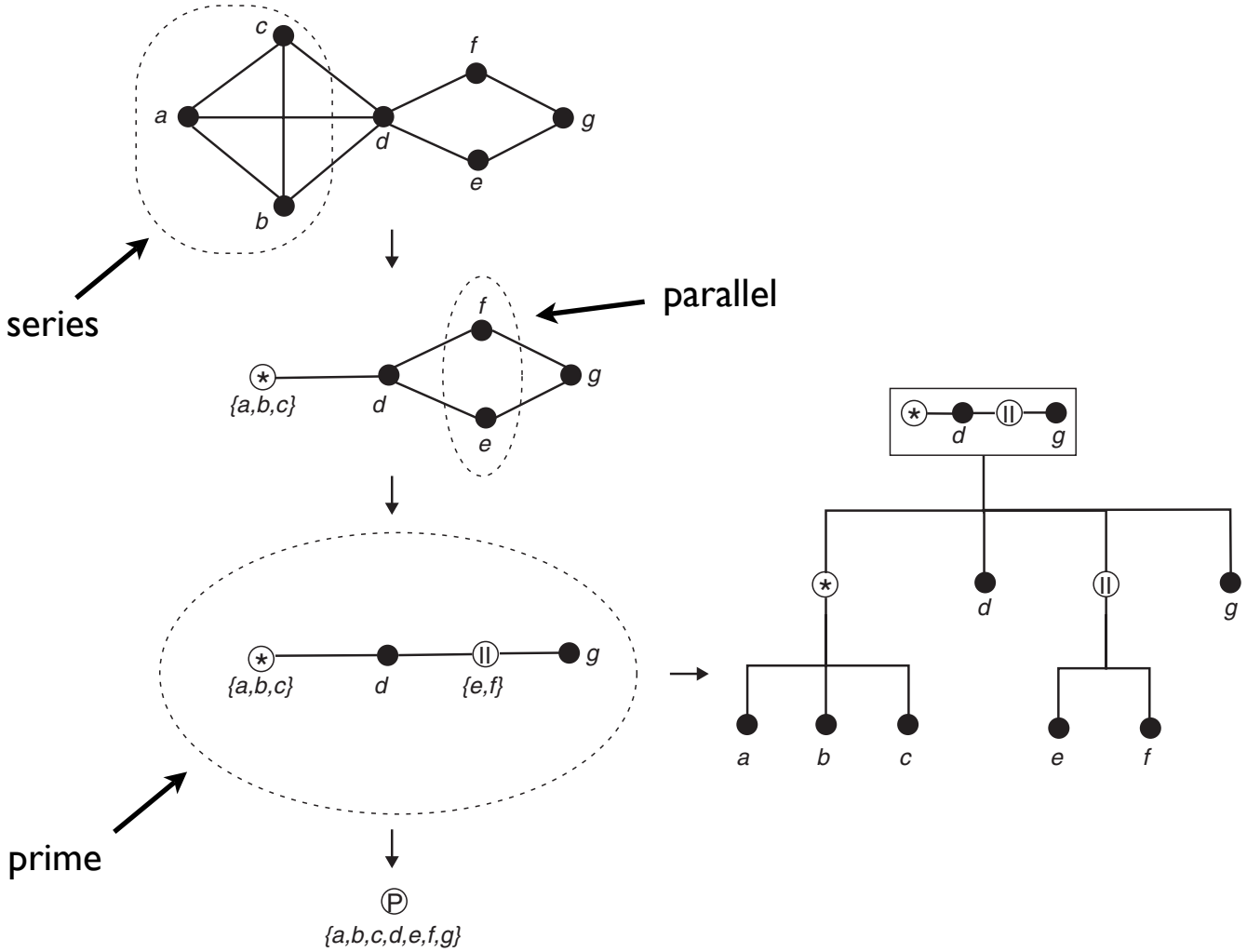
Parallel: all included nodes are non-neighbors



Prime: "anything else" (best labelled 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

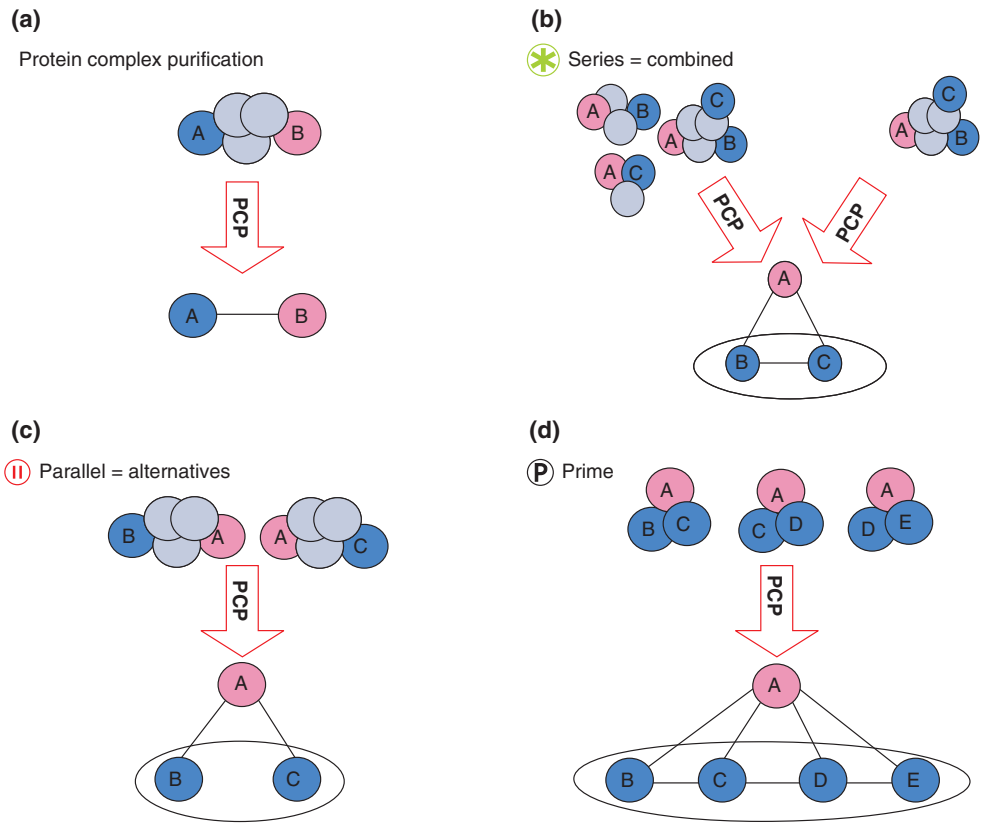
Graphs and module labels from systematic PCP experiments:

(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.



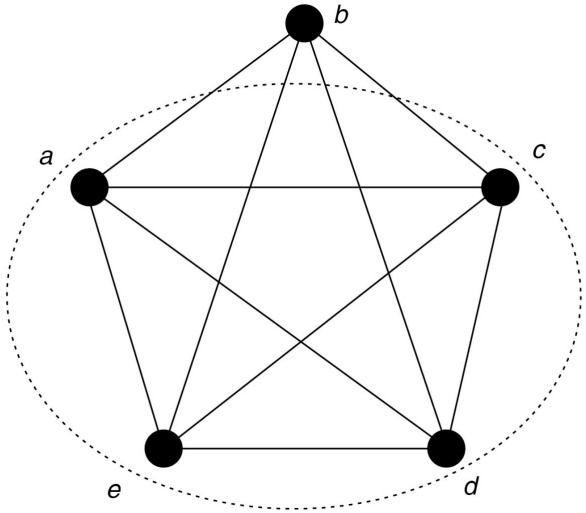
Gagneur et al. Genome Biology 5, R57 (2004)

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.



Assuming complete datasets and ideal results, a permanent complex will appear as a clique (in PCP – position of bait is irrelevant).

The opposite is not true: not every clique in the network necessarily derives from an existing complex. E.g. 3 connected proteins can be the outcome of a single trimer, 3 heterodimers or combinations thereof.

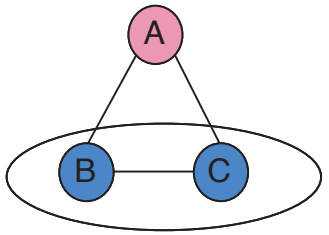
Gagneur et al. Genome Biology 5, R57 (2004)

Obtaining Maximal Cliques

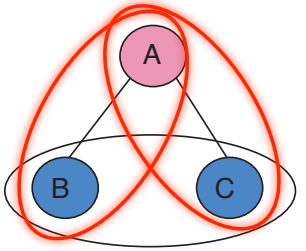
Modular decomposition provides an instruction set to deliver all maximal cliques of a graph.

In particular, when the decomposition has only **series and parallels**, the maximal cliques are straightforwardly retrieved by traversing the tree recursively from top to bottom.

A **series** module acts as a **`product`**:
the maximal cliques are all the combinations made up of one maximal clique from each „child“ node.



A **parallel** module acts as a **`sum`**:
the set of maximal cliques is the **union** of all maximal cliques from the „child“ nodes.



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

Back to the Real World …

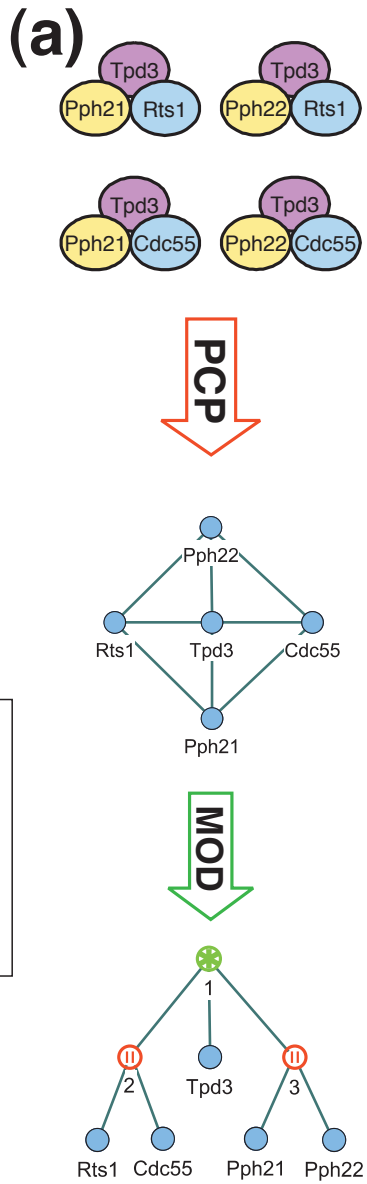
Two examples of modular decompositions of protein-protein interaction networks.

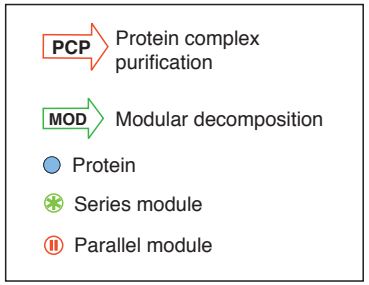
In each case from top to bottom: schemata of the complexes, the corresponding protein-protein interaction network as determined from PCP experiments, and its modular decomposition (MOD).

(a) Protein phosphatase 2A.

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

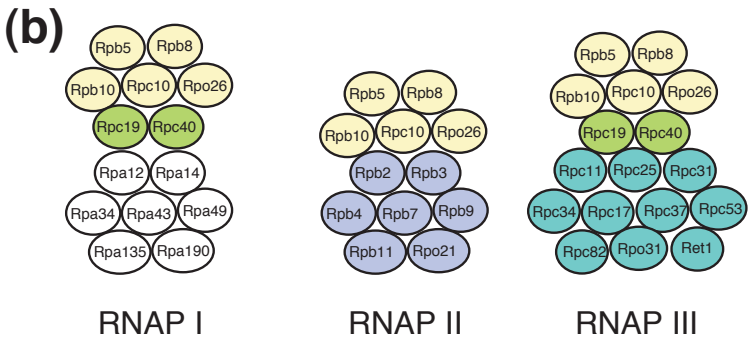
Here these are the catalytic Pph21 and Pph22 (module 2) and the regulatory Cdc55 and Rts1 (module 3), connected by the Tpd3 „backbone“.

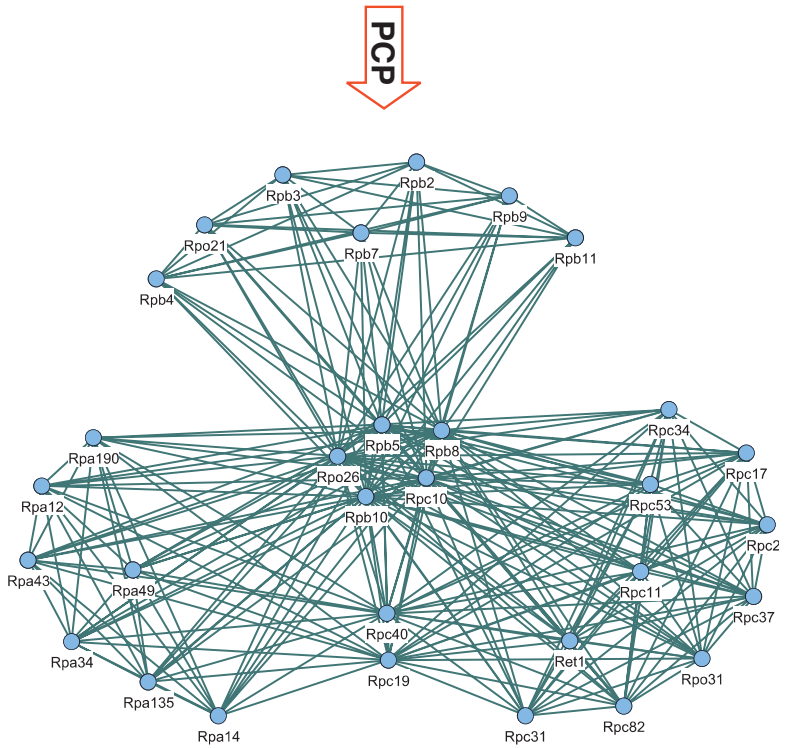


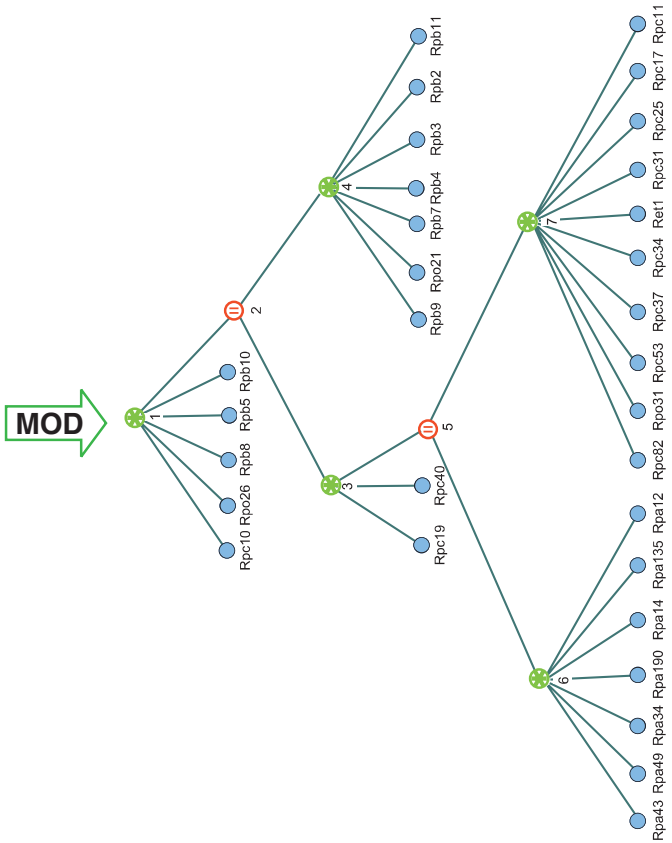


Notes:
- Graph does not show functional alterantives!!!
- other decompositions also possible

RNA polymerases I, II and III







Again: modular decompositon easier to comprehend than graph

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

Conclusions

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

- It can be thoroughly proven 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 allways fulfil 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

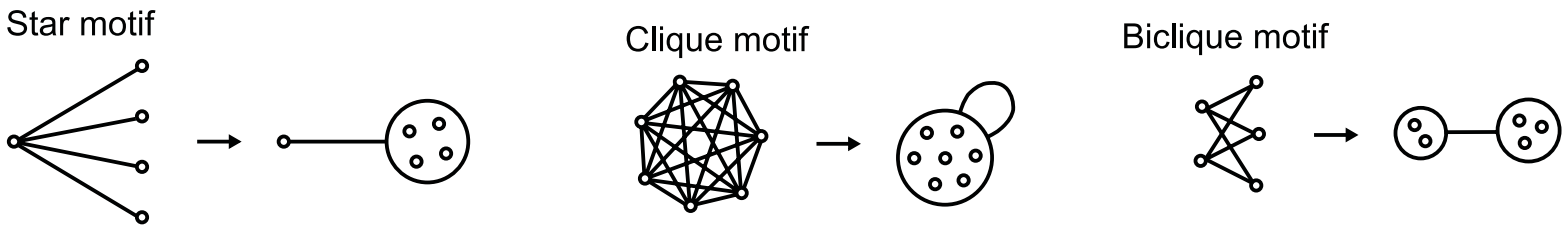
Exploit observation that **cliques** and bi-cliques are **abundant** in real networks => **explicitly** represented in power graphs

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

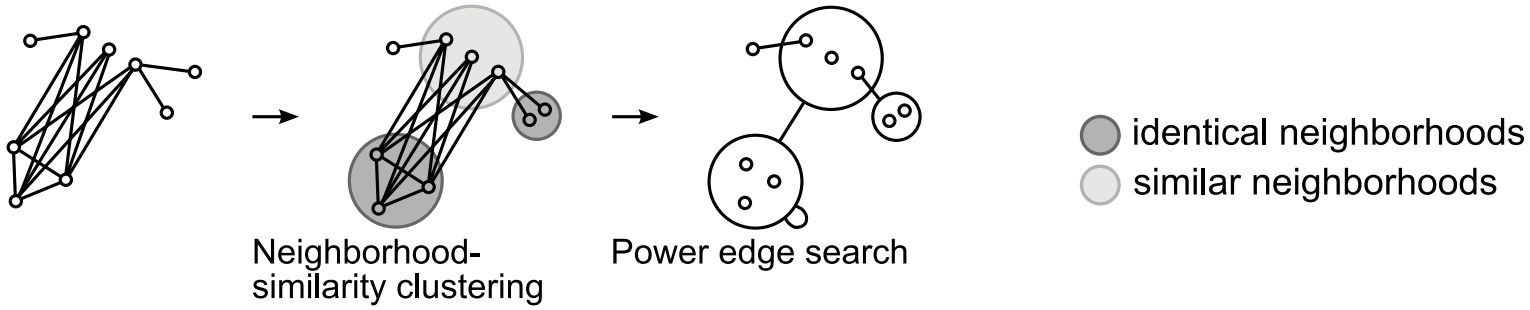
Power Graph Analysis Algorithm

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?

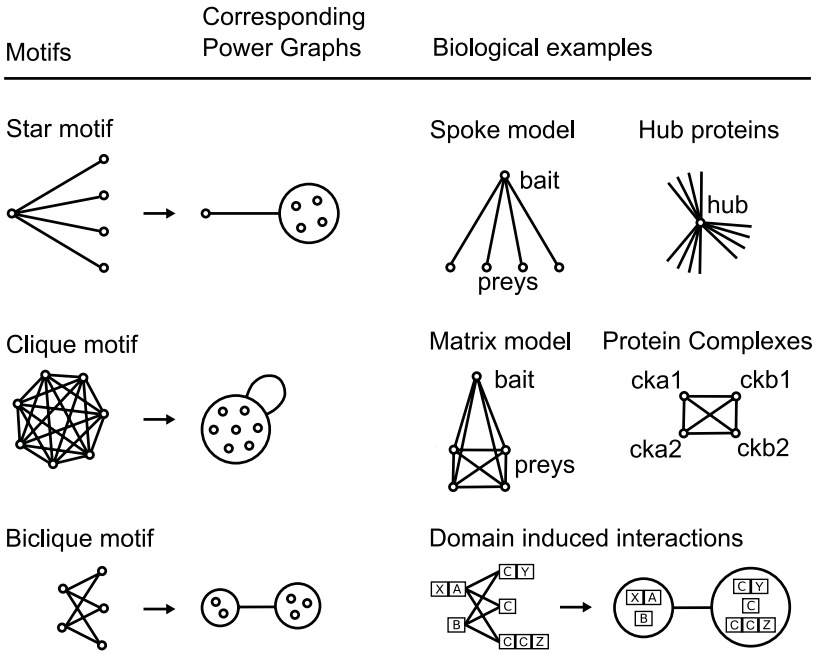


Figure 1. The Three Basic Motifs: Star, Biclique, and Clique. Stars often occur because of hub proteins or when affinity purification complexes are interpreted using the spoke model. Bicliques often arise because of domain-domain or domain-motif interactions inducing protein interactions [25]. Power 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

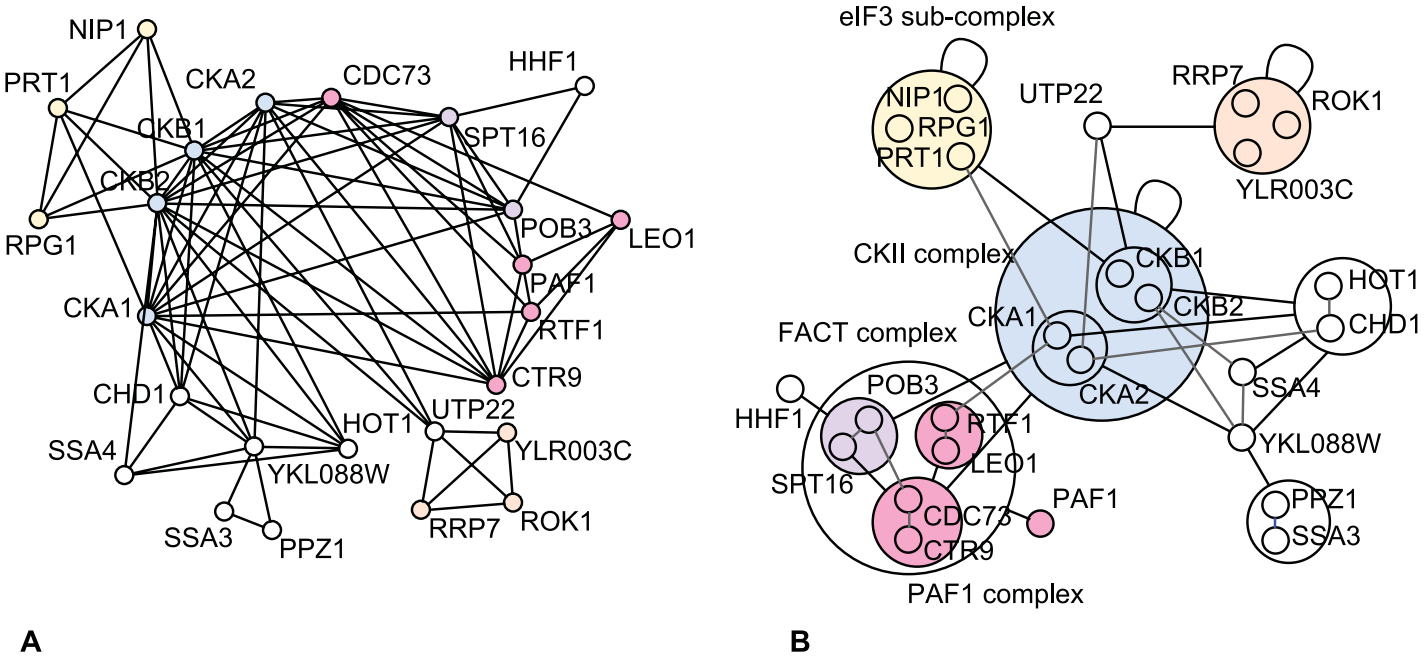


Figure 2. Casein Kinase II Complex. Two catalytic alpha subunits (CKA1, CKA2) and two regulatory beta subunits (CKB1, CKB2) interacting with the FACT complex, with sub-complex NIP1-RPG-PRT1, and with the PAF1 complex. The graph representation (A) consists of 80 edges whereas the power graph representation (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

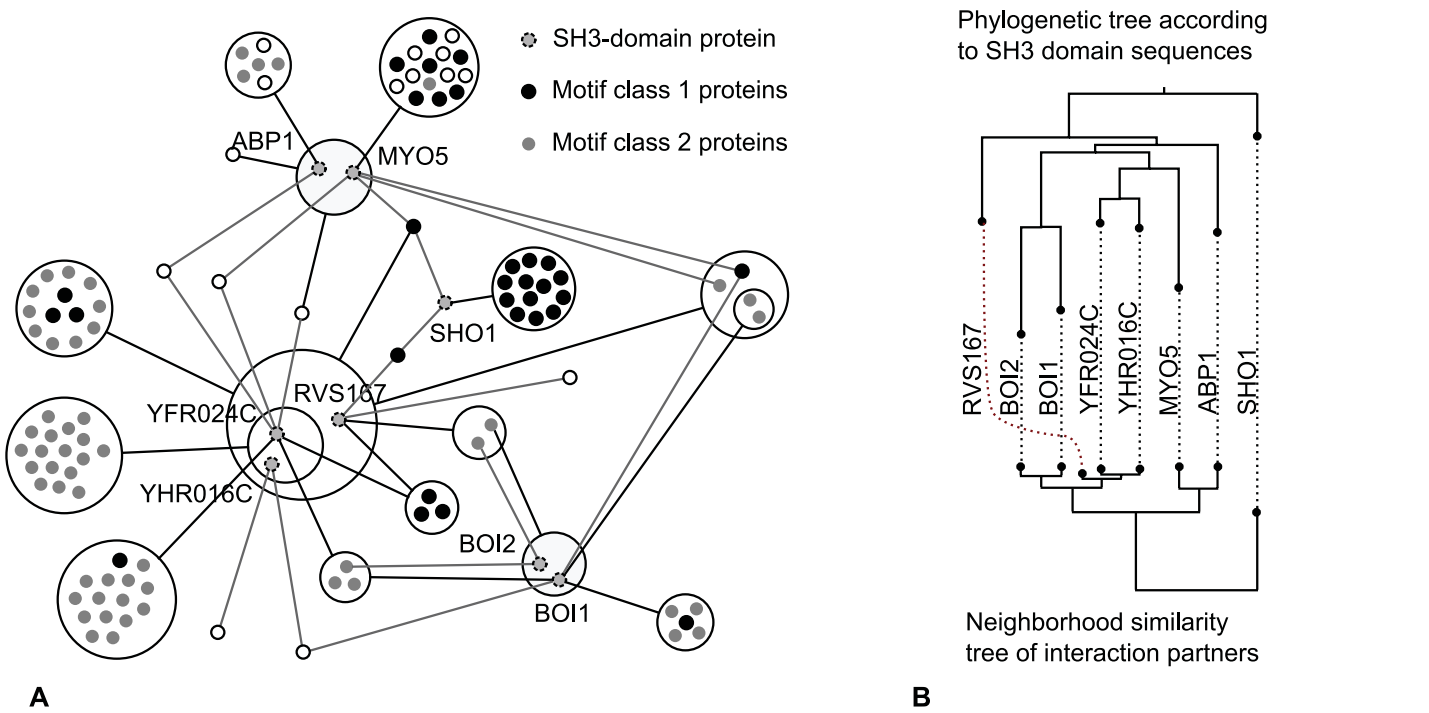


Figure 4. Interactions of SH3 Carrying Proteins. (A) Protein interaction network showing the 105 interaction partners of the SH3 domain carrying proteins: SHO1, ABP1, MYO5, BOI1, BOI2, RVS167, YHR016C and YFR024. The underlying network consists of 182 interactions represented here as 36 power edges–a 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

Many Cliques and Bi-Cliques

Compare original PPI networks to randomly **re-wired** networks:

=> replace edges {u, v} and {s, t} by {u, t} and {s, v}

=> keeps degree of each node, destroys modules

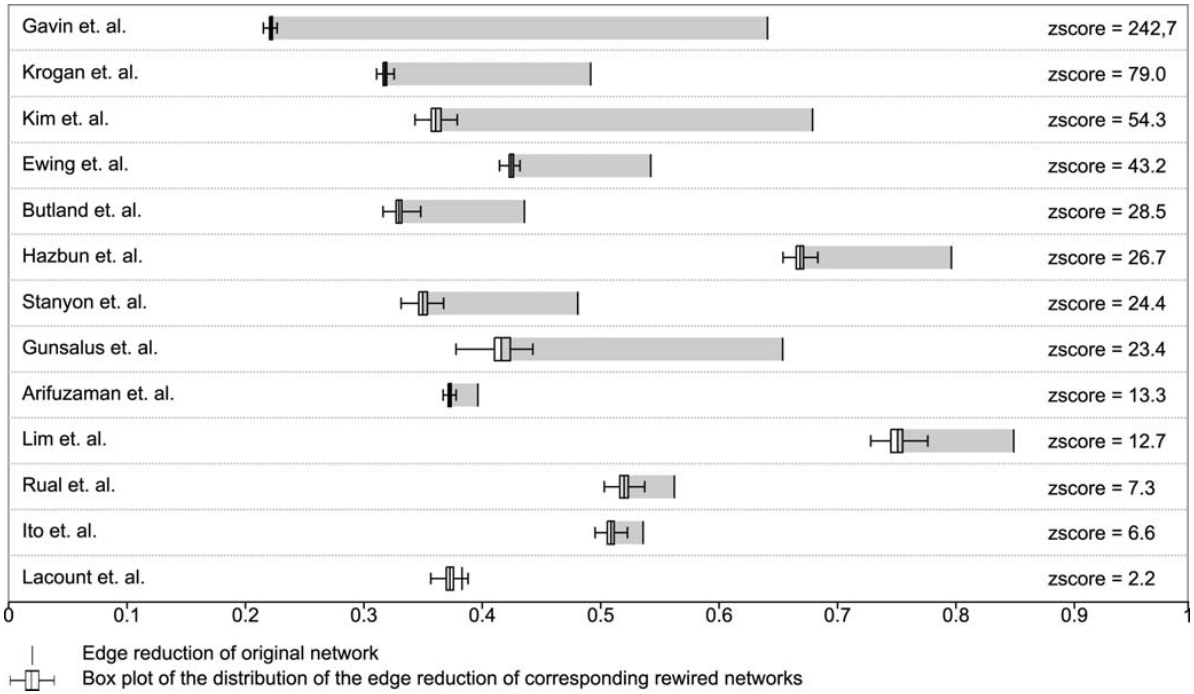
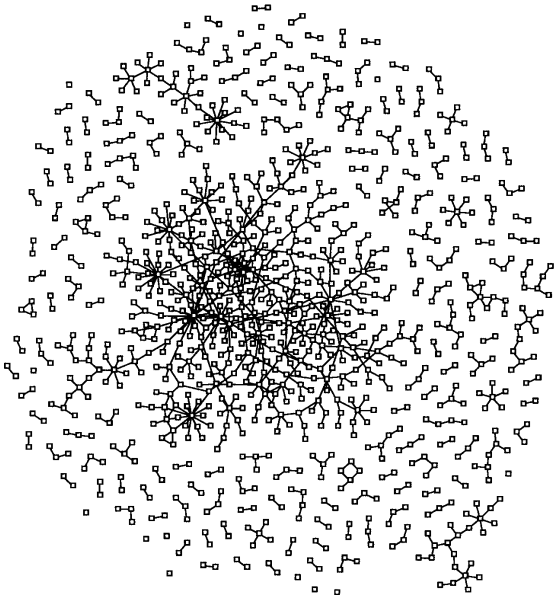


Figure 5. Comparison of 13 Protein Interaction Networks to Corresponding Randomly Rewired Networks. The edge reduction of the rewired networks is represented using a a box-plot. 50% of edge reduction values are inside the box. Most networks exhibit a significant deviation from the null model as indicated by high z-scores between 2.2 and 242.
doi:10.1371/journal.pcbi.1000108.g005

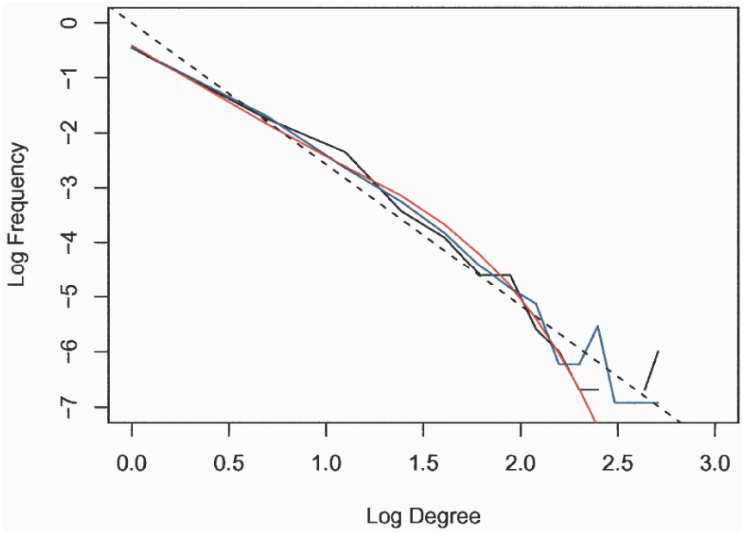
Royer et al, *PLoS Comp Biol* **4** (2008) e1000108

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: Tue, Nov. 8**
- Are biological networks scale-free? (other models?)
- Network growth mechanisms