

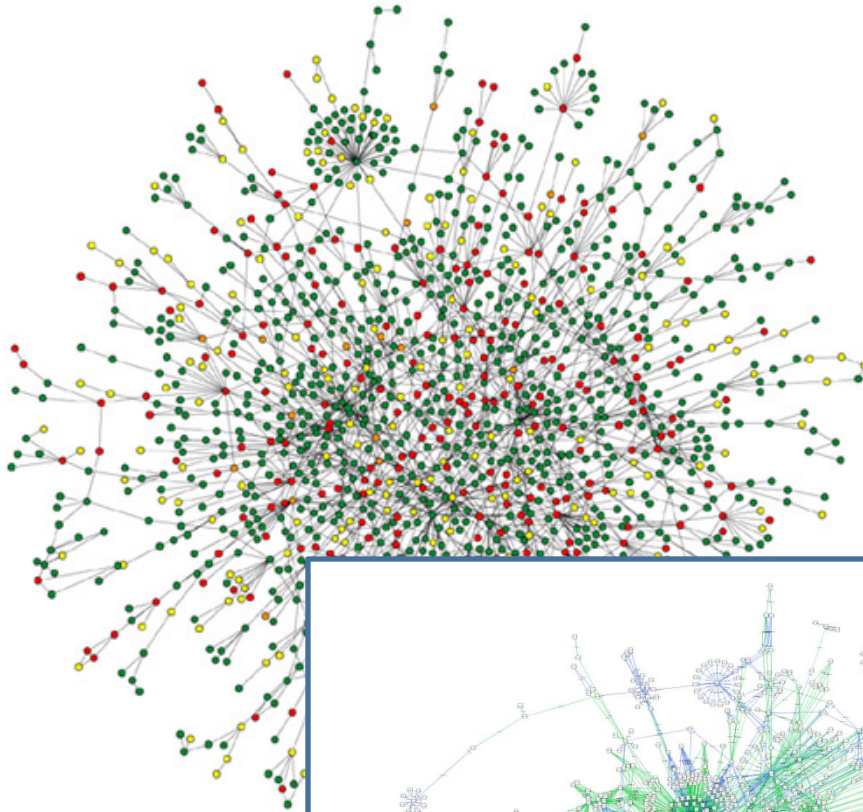
Networks

Can (John) Bruce

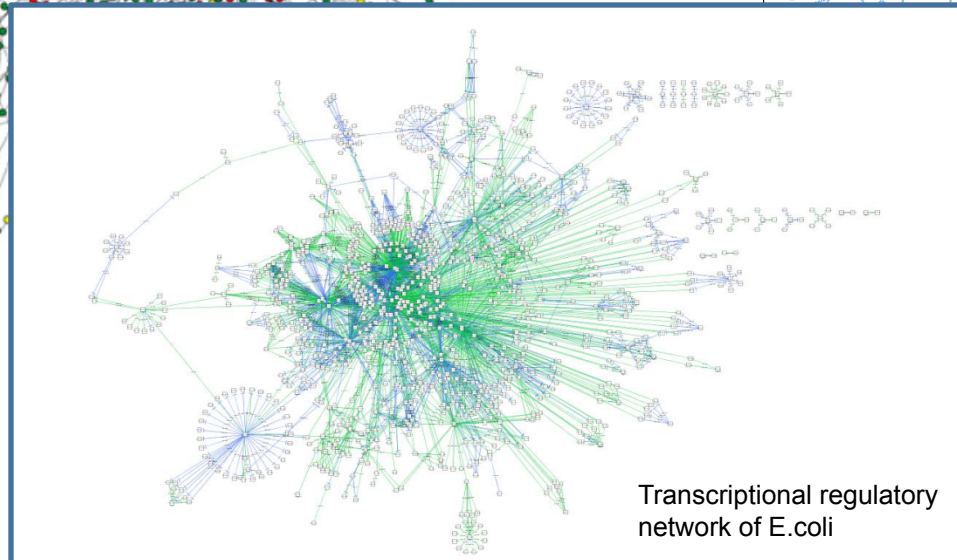
Keck Foundation Biotechnology Lab

Bioinformatics Resource

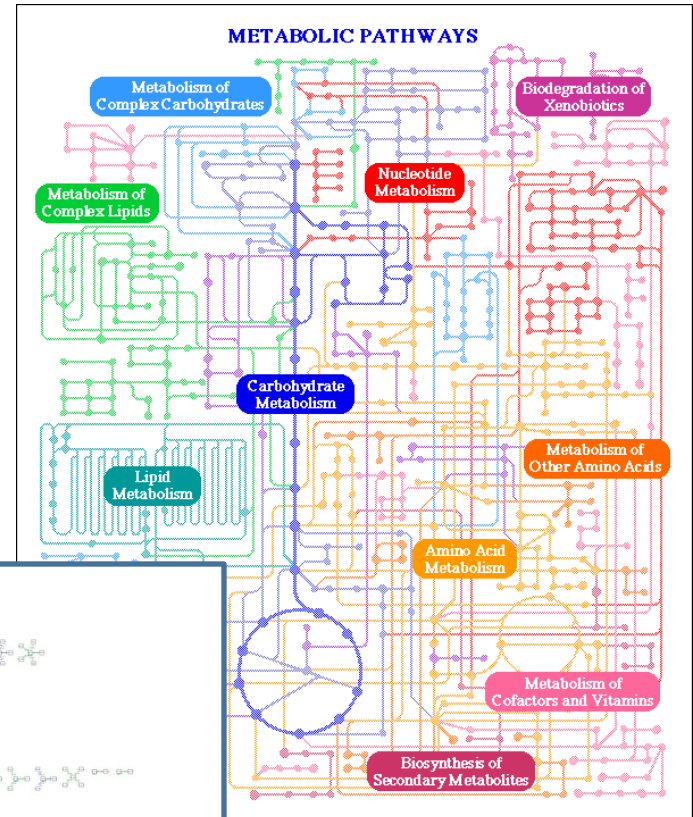
Networks in biology



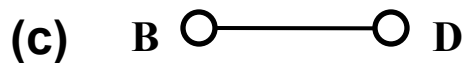
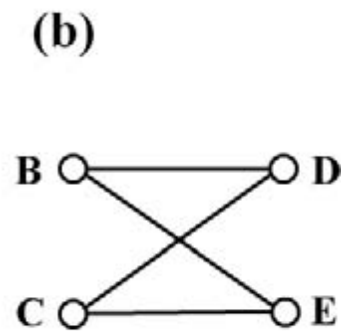
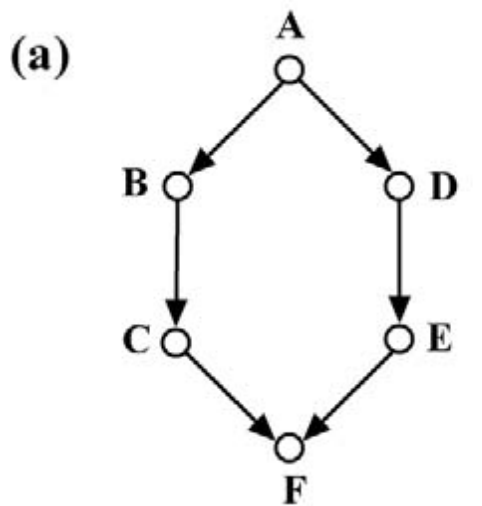
Protein-Protein
Interaction
Network of Yeast



Transcriptional regulatory
network of E.coli



Experimental data shown as networks



(a) Regulatory network

(b) Synthetic lethal interactions network where edges correspond to pairs of nodes whose loss causes the disconnection of nodes A and F.

(c) Expression correlation network, where pairs of genes whose expression are correlated under different conditions define edges.

What do networks tell us?

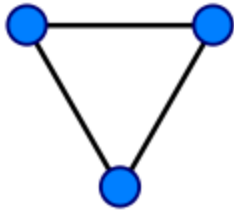
1) The structure of these networks becomes now a new entity that can be studied.

- How these networks came about?
- How does the structure of the network or its components relates to their function?

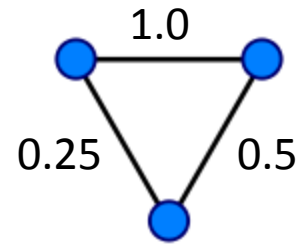
2) We can make use of existing network information to interpret better new experimental data:

- What interactions connect genes that are differentially regulated with a certain treatment?
- What is the likeliest mechanism to explain the observed behavior of a number of proteins?

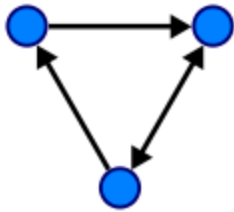
Glossary



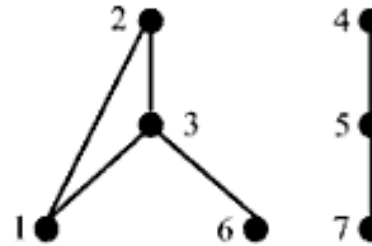
An undirected graph



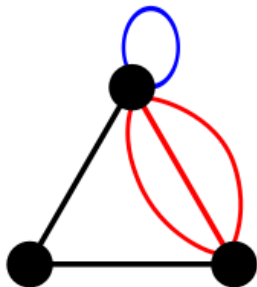
A weighted, undirected graph



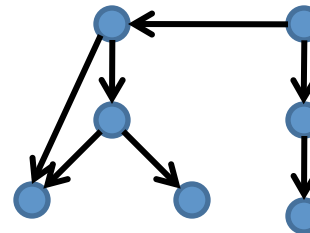
A directed graph



An unconnected, undirected graph

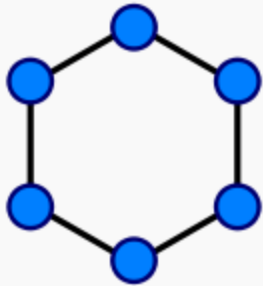


A multigraph graph (multiple edges with the same end nodes allowed)

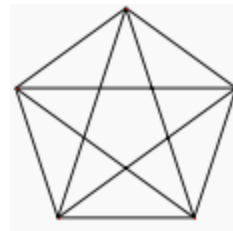


A weakly connected graph

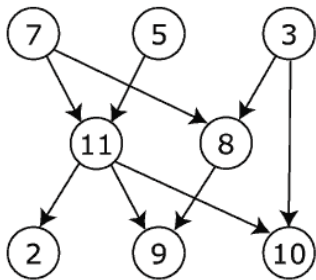
Glossary



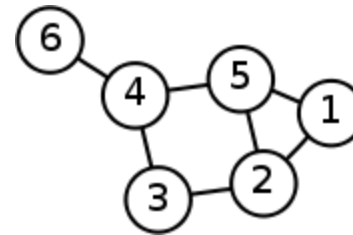
A cyclic graph



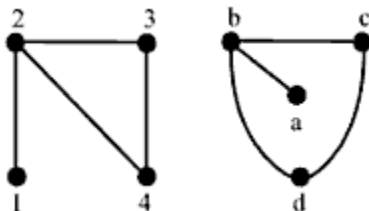
A complete graph



An acyclic graph



Nodes 1,2,5 form a clique of size 3.



Two isomorphous graphs

Some Network Properties

How do you tell whether two networks are "similar"?

- Global Network Properties
 - Degree distribution
 - Network diameter
 - Clustering coefficients
- Local Network Properties

Degree distribution of a network

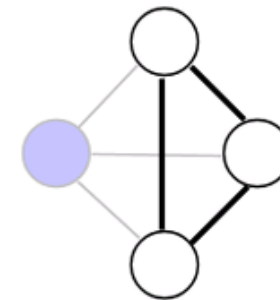
- The degree of a node is the number of edges (connections) linking to the node.
- The degree distribution, $P(k)$, describes the probability that a node has degree k .
- Erdős–Rényi random networks have a Poisson degree distribution
- Scale-free networks have a power-law degree distribution $P(k) \sim k^{-\gamma}$, where γ is a positive number

Network diameter

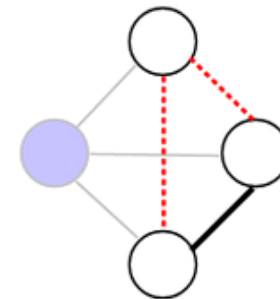
- The smallest number of links that have to be traversed to get from node x to node y in a network is called the distance between nodes x and y .
- A path through the network that achieves this distance is called shortest path between x and y .
- The average of shortest path lengths over all pairs of nodes in a network is called the network diameter.
 - Erdős–Rényi random networks: $\sim \log n$
 - Scale-free random networks with degree exponent $2 < \gamma < 3$ (i.e., most real-world networks): $\sim \log \log n$.

Clustering coefficient

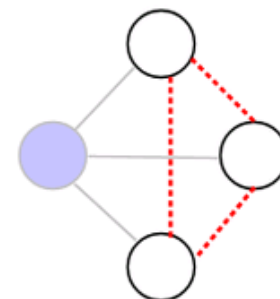
- The clustering coefficient of a vertex in a graph quantifies how close the node and its neighbors are to being a clique (complete graph)
- The clustering coefficient of node v in an undirected network is defined as $C_v = 2e_1 / [n_1(n_1 - 1)]$, where v is linked to n_1 neighboring nodes and e_1 is the number of edges amongst the n_1 neighbors of v .
- For a directed network $C_v = e_1 / [n_1(n_1 - 1)]$
- The average of C_v over all nodes v of a network is the clustering coefficient C of the whole network and it measures the tendency of the network to form highly interconnected regions called clusters.



$$c = 1$$



$$c = 1/3$$



$$c = 0$$

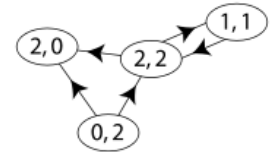
Example clustering coefficient on an undirected graph for the shaded node i . Black line segments are actual edges connecting neighbors of i , and dotted red segments are missing edges.

Local characteristics of networks

- Local characteristics within a network are associated with individual nodes, edges or subgraphs.
- These characteristics provide information about the role of these network components within the network as a whole.

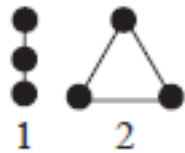
Local Network properties

- Degree: Number of connections to a node (Divided into 'in-degree' and 'out-degree' for directed systems)
- Centrality (importance of a node)
 - Degree centrality (Important nodes have more connections to other nodes) "Hubs"
 - Closeness centrality (Important nodes have low average distance between them and other nodes)
 - Betweenness centrality (Important nodes are those that make paths within the network short)
- Subgraphs
 - Cliques (complete connected subgraph)
 - Motifs (Graphlets) : a statistically overrepresented subgraph in a network



Graphlets (network motifs) of up to 5 nodes

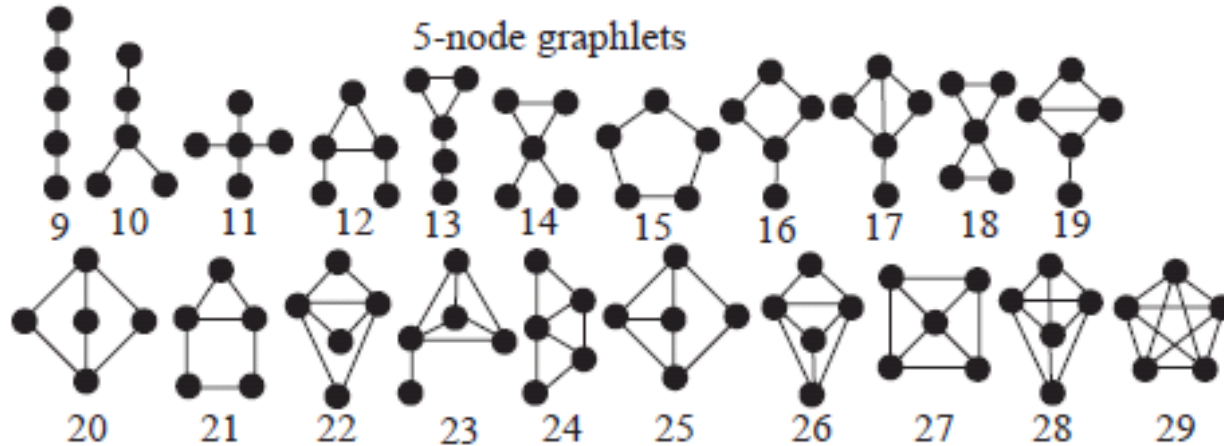
3-node graphlets



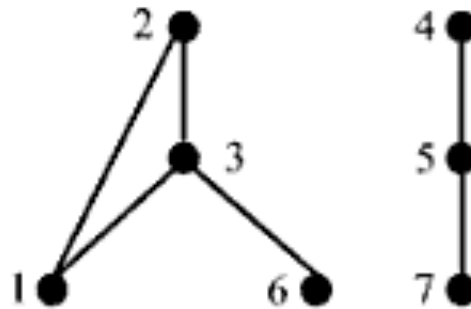
4-node graphlets



5-node graphlets



Graph representations



An $n \times n$ adjacency matrix representation

vertex	1	2	3	4	5	6	7
1	0	1	1	0	0	0	0
2	1	0	1	0	0	0	0
3	1	1	0	0	0	1	0
4	0	0	0	0	1	0	0
5	0	0	0	1	0	0	1
6	0	0	1	0	0	0	0
7	0	0	0	0	1	0	0

An adjacency list, an array of $[1 \dots n]$ lists

$L_1: (\{1, 2\}, \{1, 3\})$
 $L_2: (\{2, 1\}, \{2, 3\})$
 $L_3: (\{3, 1\}, \{3, 2\}, \{3, 6\})$
 $L_4: (\{4, 5\})$
 $L_5: (\{5, 4\}, \{5, 7\})$
 $L_6: (\{6, 3\})$
 $L_7: (\{7, 5\})$

Trade-offs

Adjacency Matrix vs. Adjacency List

- Each entry in a adjacency matrix requires one bit, requiring $\sim n^2/8$ bytes of storage space.
- An adjacency list for an undirected graph requires $8e$ bytes of storage (each edge \rightarrow two entries that use 4 bytes, on a 32-bit computer)
- A graph can have at most n^2 edges (including loops),
- Define density $d=e/n^2$. So, when $8e > n^2/8$, an adjacency list occupies more space, i.e., when $d > 1/64$.
- Thus, only sparse graphs are more efficiently stored as adjacency list.

Trade-offs

Adjacency Matrix vs. Adjacency List (2)

- It is easy to find all vertices adjacent to a given node in an adjacency list. In an adjacency matrix, you must scan an entire row ($O(n)$ time)
- It is easy to find whether two nodes have an edge connecting them in an adjacency matrix. In an adjacency list, you need to scan a row in ($O(n)$ time, where n is minimum degree of nodes in the adjacency list)

Degree distribution of complex networks

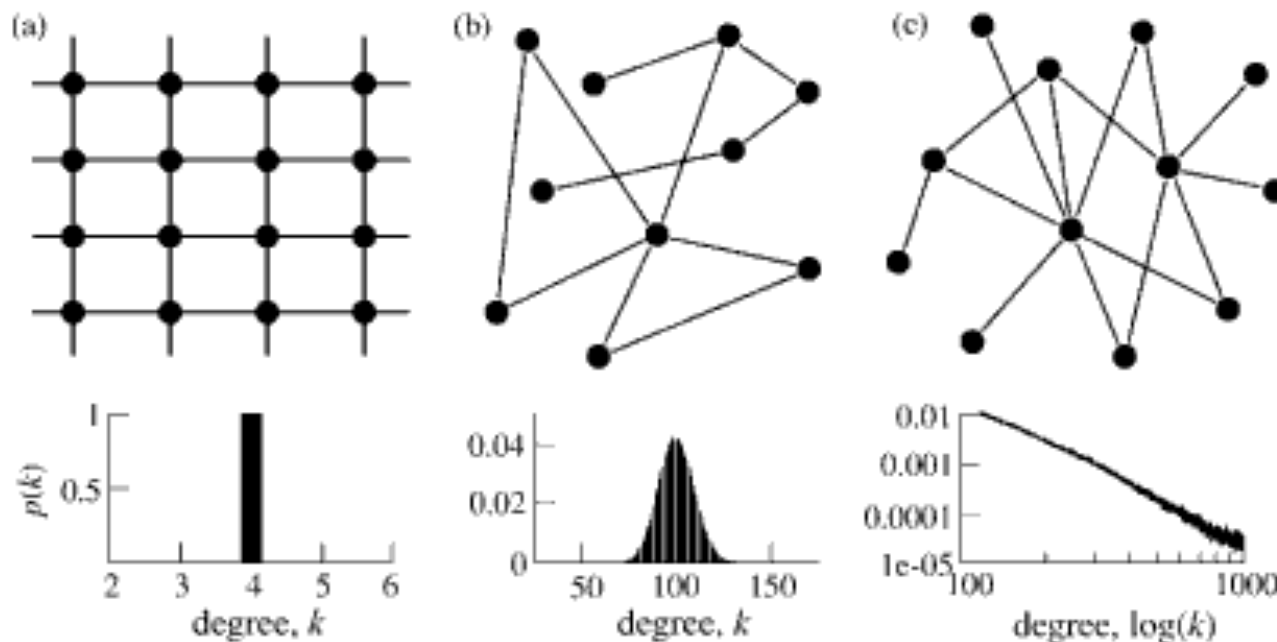


FIGURE 3.3 Degree distributions of complex networks. (a) A lattice-like network. Each vertex has the same degree k (for periodic boundary conditions or large networks, such that vertices at the border can be neglected). (b) An Erdős-Rényi random network. The degree distribution is homogeneous, the degrees of the vertices are centered around the average value. (c) A scale-free network. The degree distribution is highly inhomogeneous and follows a power law of the form $p(k) \sim k^{-\gamma}$, where γ denotes the *degree exponent*. While most vertices have a low number of connections only, a smaller number of vertices is highly connected.

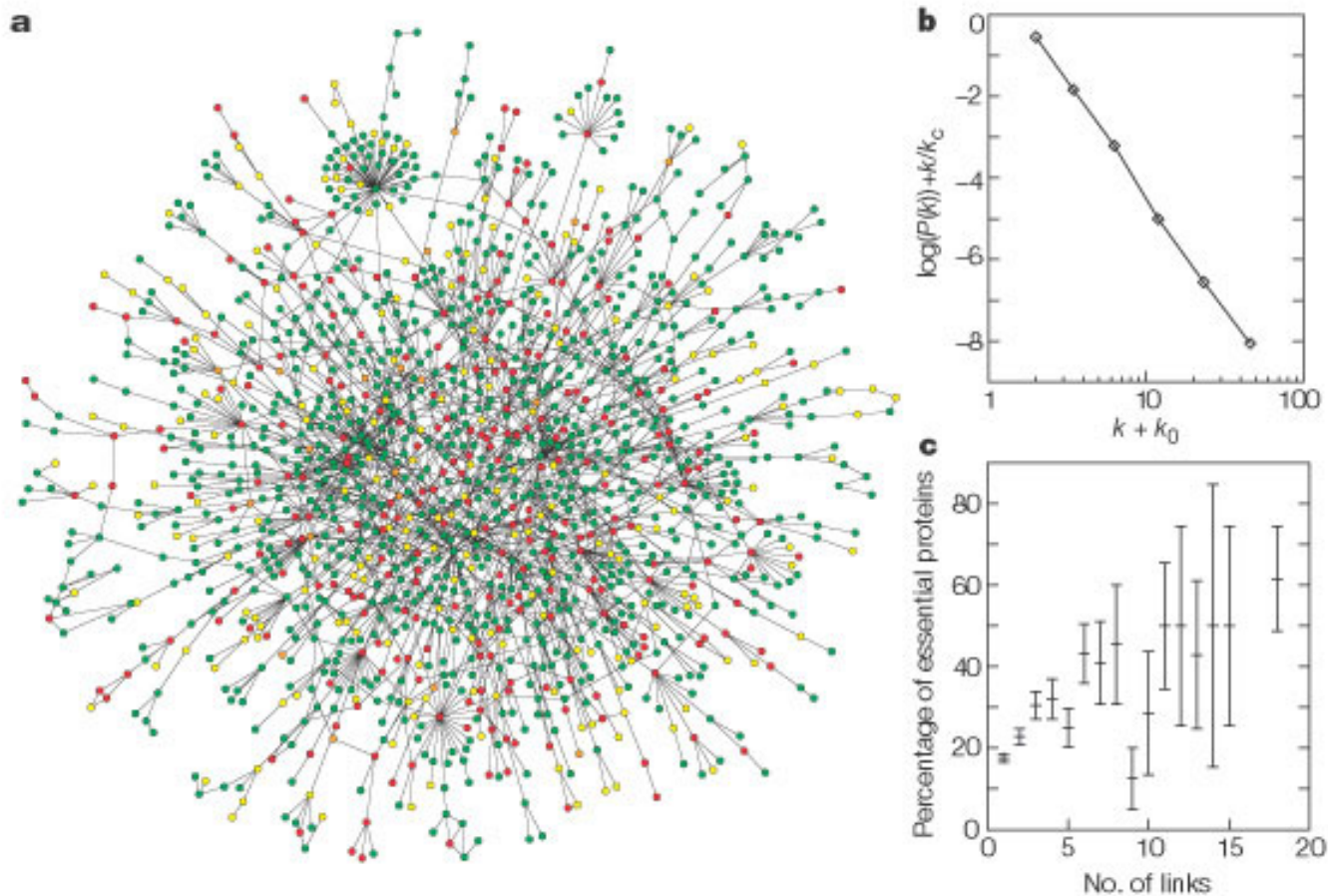
Some scale-free networks

- Social networks, including collaboration networks.
- Protein-Protein interaction networks.
- Sexual partners in humans.
- Many kinds of computer networks, including the World Wide Web.
- Semantic networks.

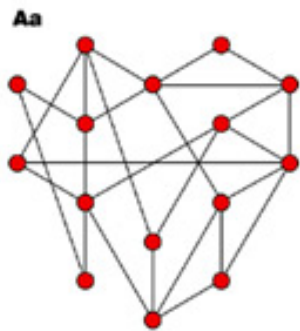
Characteristics of scale-free networks

- Occurrence of "hubs" more frequent than in random networks.
- Tend to remain connected even if a few nodes are lost. However, if hubs are lost then network usually falls apart into several subnetworks.
- The fraction $P(k)$ of nodes in the network having k connections to other nodes $\sim k^{-\gamma}$
- For $2 < \gamma < 3$ will also have ultrasmall diameter $d \sim \ln \ln N$. The diameter of a growing scale-free network might be considered almost constant

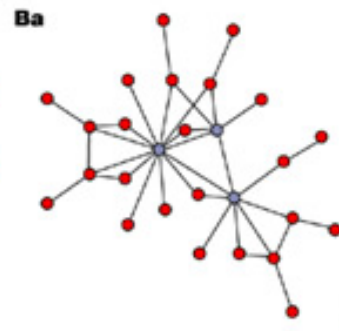
Lethality and centrality in protein networks



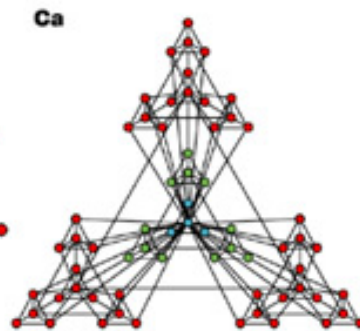
A Random network



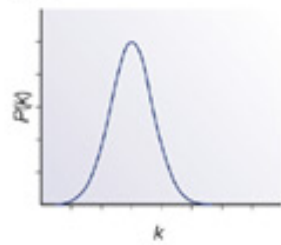
B Scale-free network



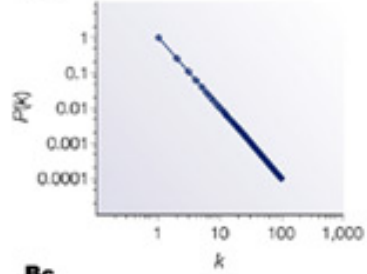
C Hierarchical network



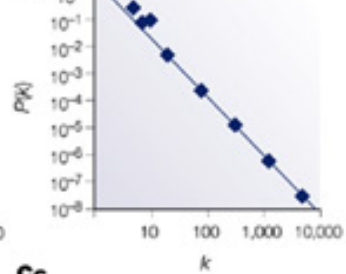
Ab



Bb



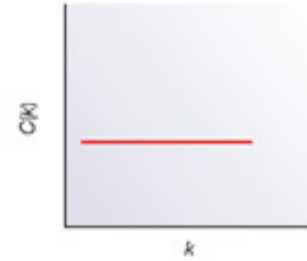
Cb



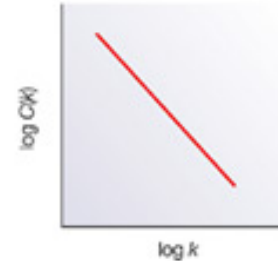
Ac



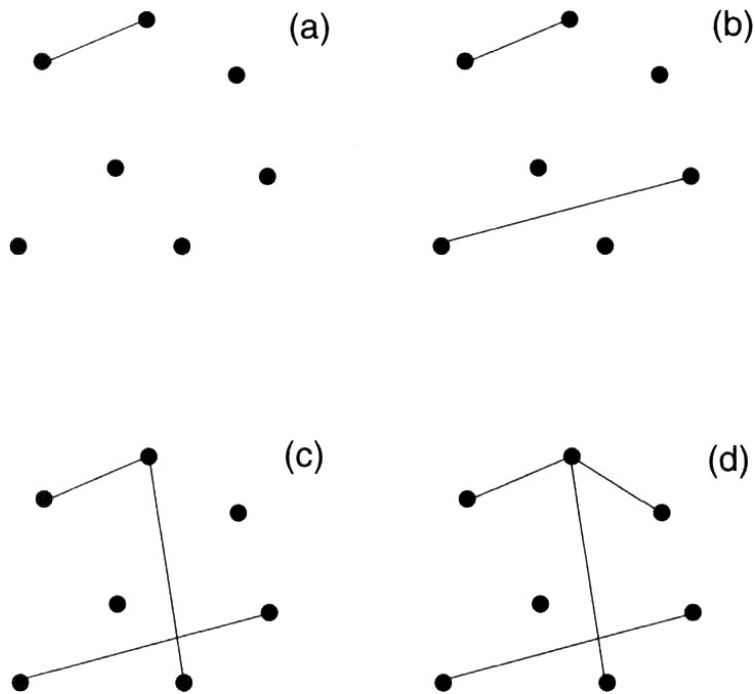
Bc



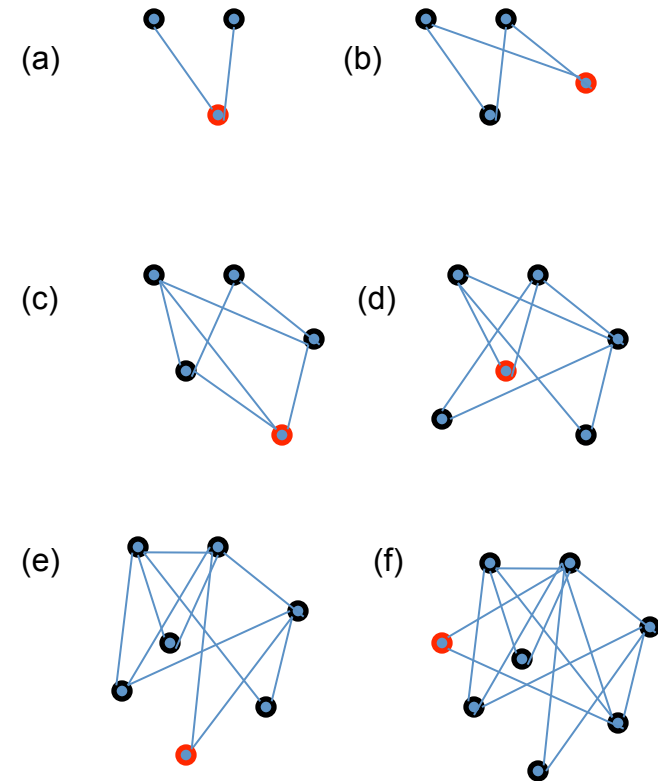
Cc



Generative Models of Networks

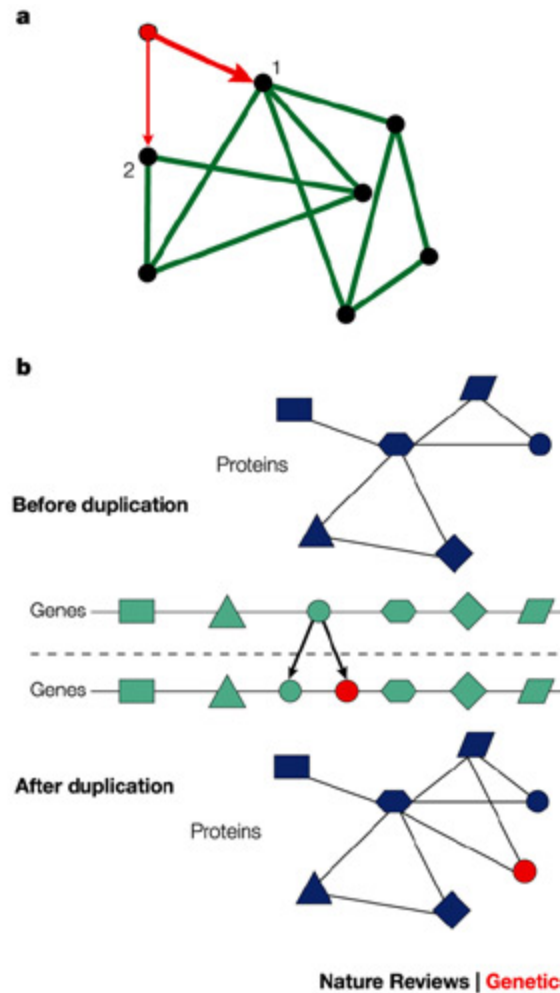


Example of an equilibrium network, a classical random graph (the Erdős-Rényi model). Pairs of randomly chosen vertices are connected by edges. The total number of vertices is fixed.



A scale-free network (Barabasi-Albert model). At each step a new node is added and two new edges from the new node to the old ones.

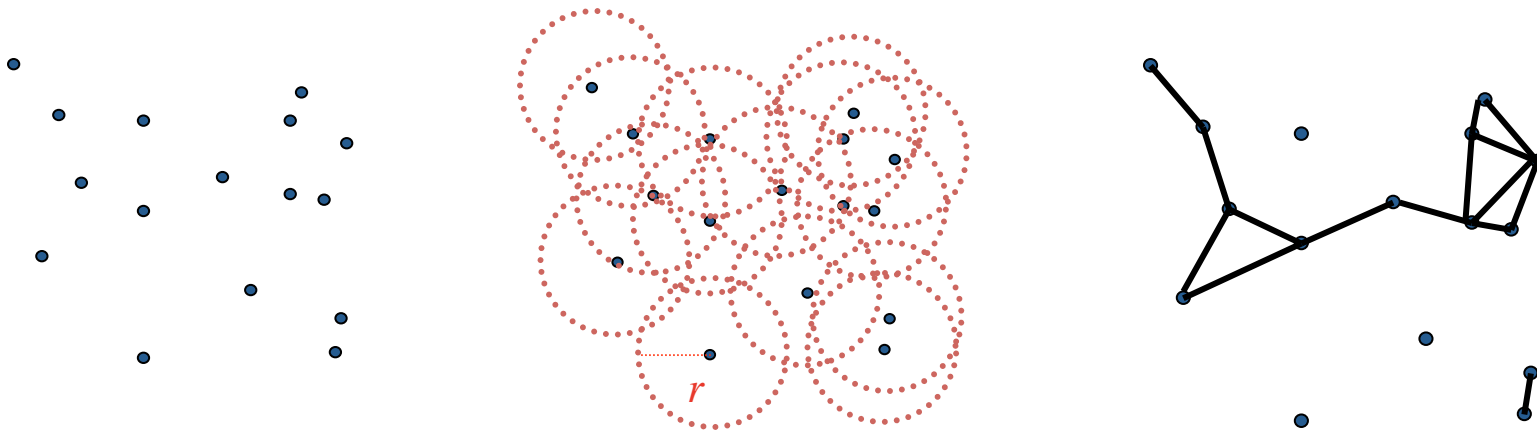
The origin of the scale-free topology and hubs in biological networks



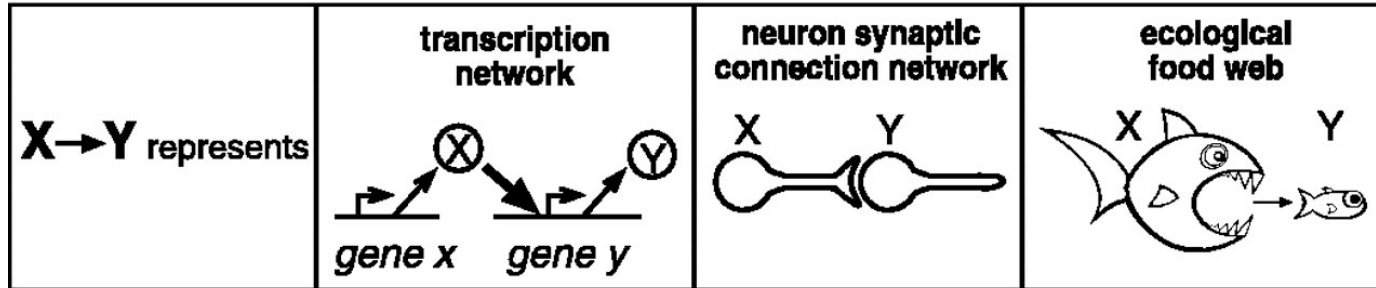
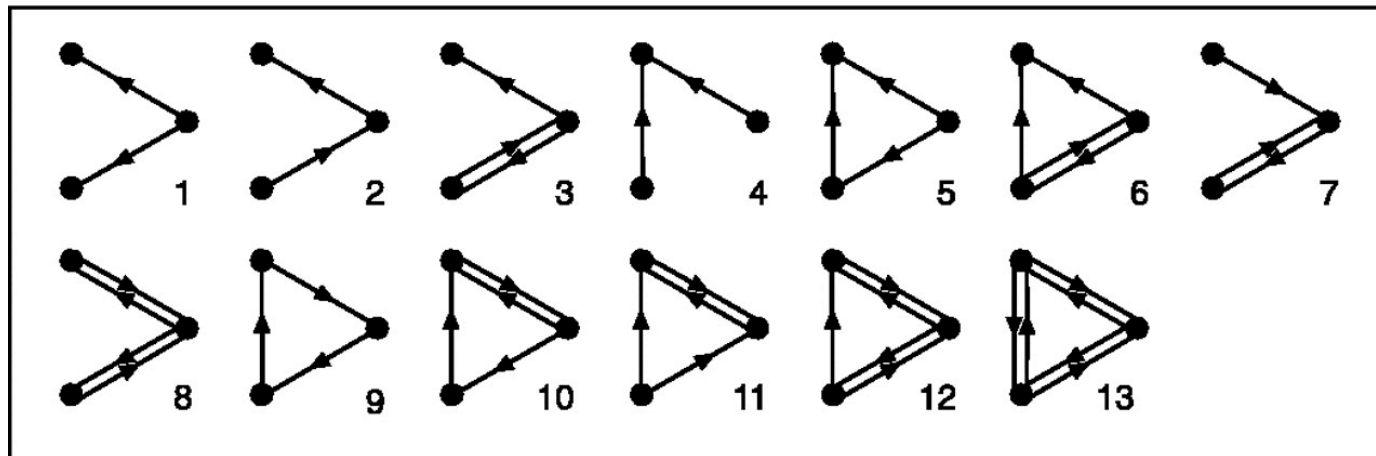
Geometric Network

- Geometric random graphs in 2-D Euclidian space are generated as follows:
- We place N nodes uniformly in the unit square and two nodes are connected by an edge if and only if they are within Euclidian Distance r . (3 and 4-d cases are analogous)
- For protein-protein interactions, compute path lengths up to length K . compute 2 most positive Eigen values and corresponding eigenvectors of the scaling matrix from the connectivity data. Embed nodes in 2-dimensional space. Search for an r such that the resulting geometric graph matches the given network.

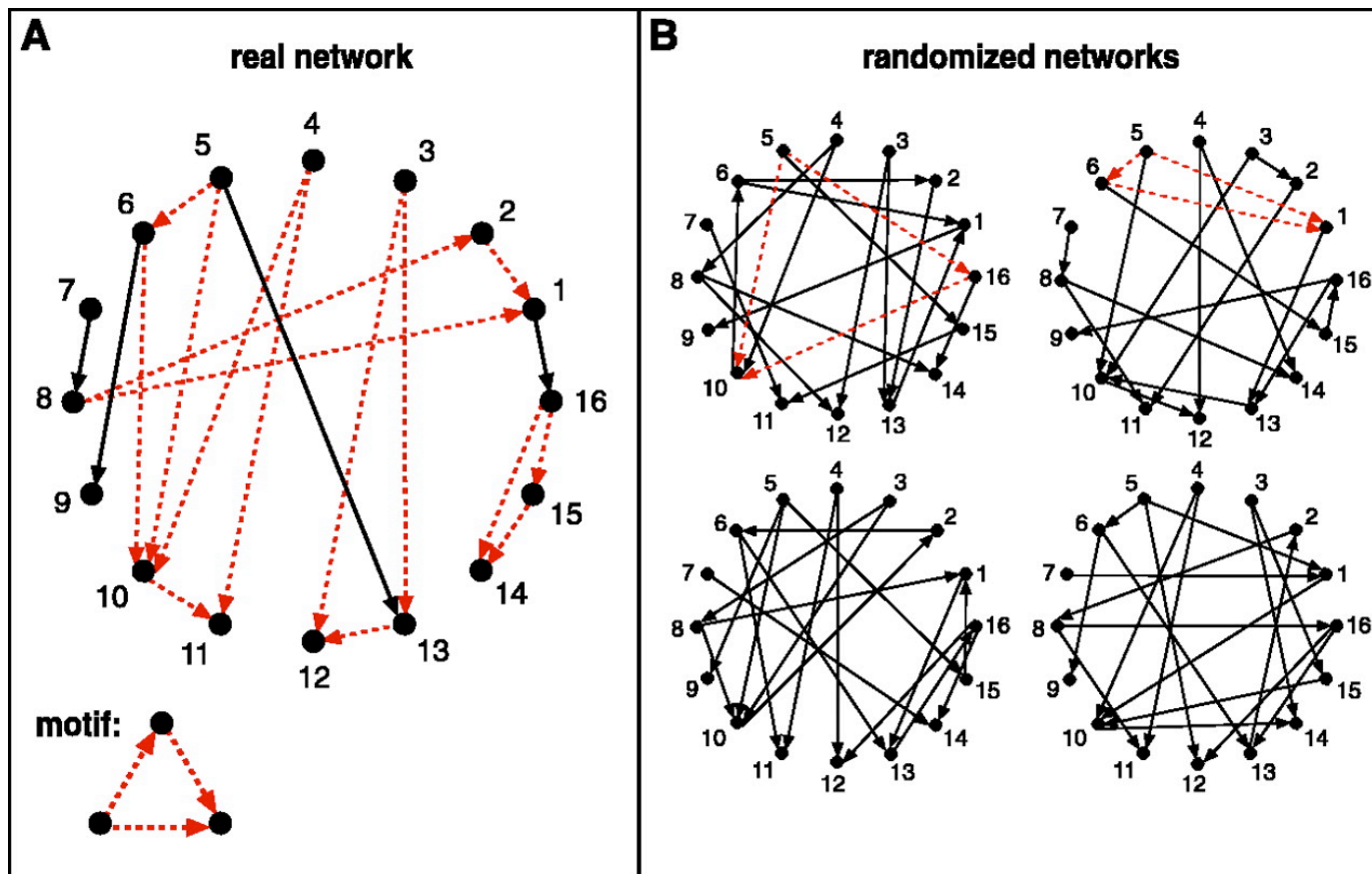
Random geometric network



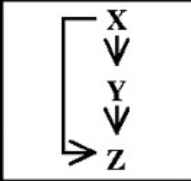
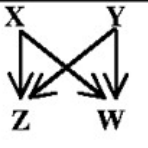
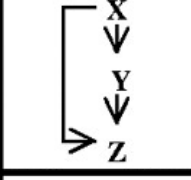
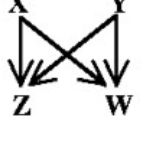
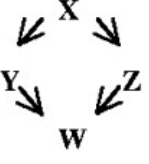
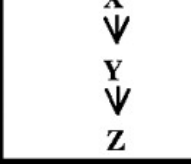
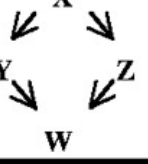
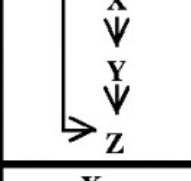
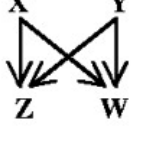
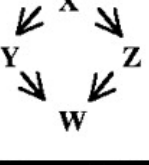

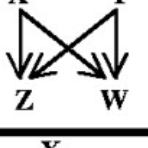
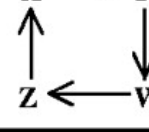
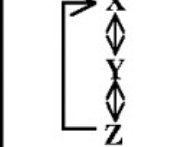
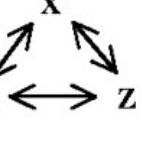
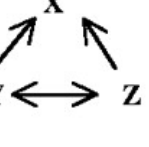
NETWORK MOTIFS

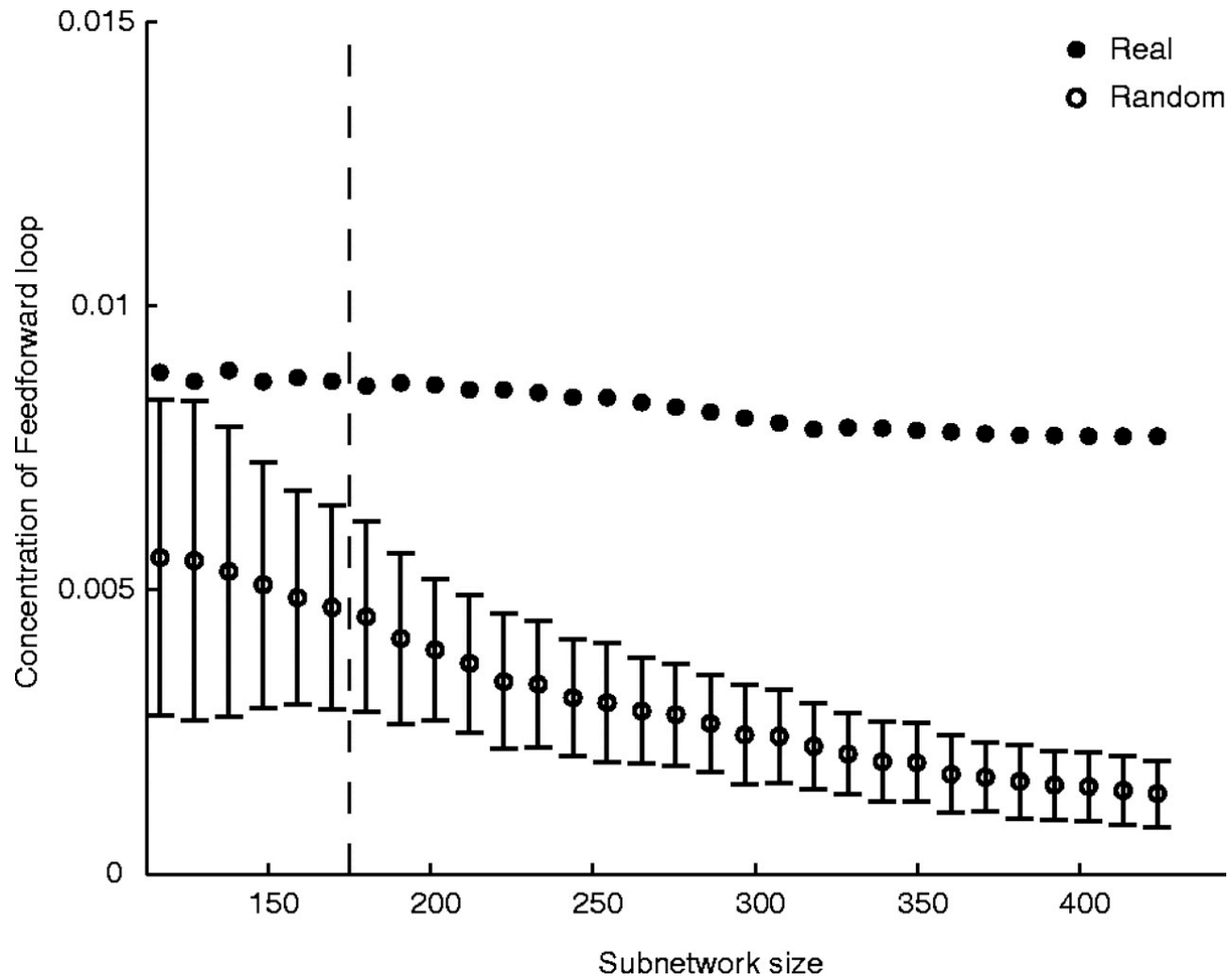
A**B**

How to assess significance of

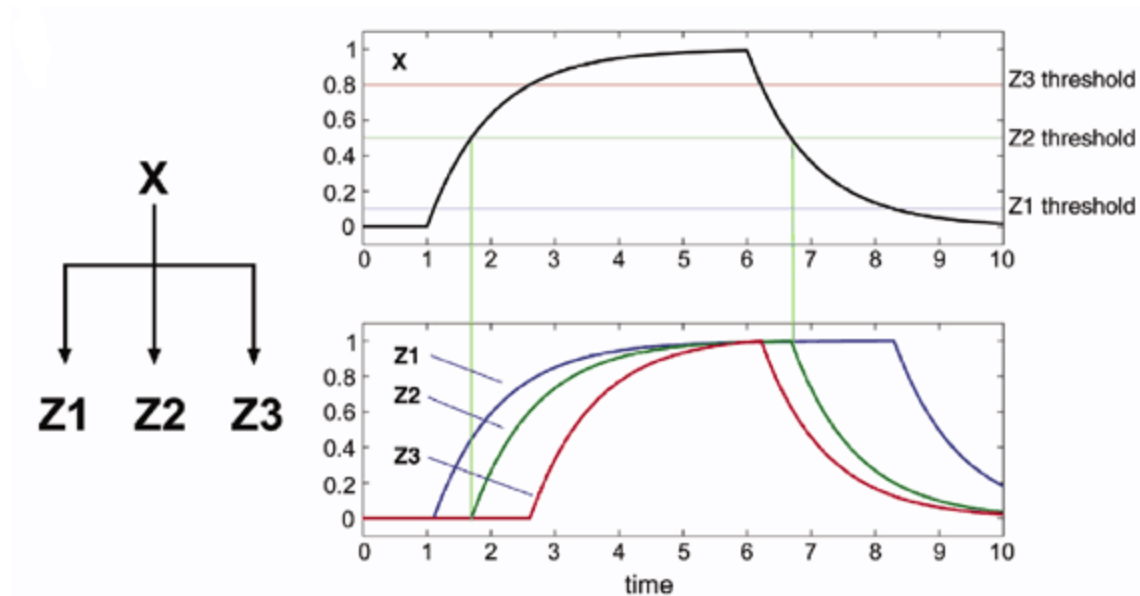


Common network motifs

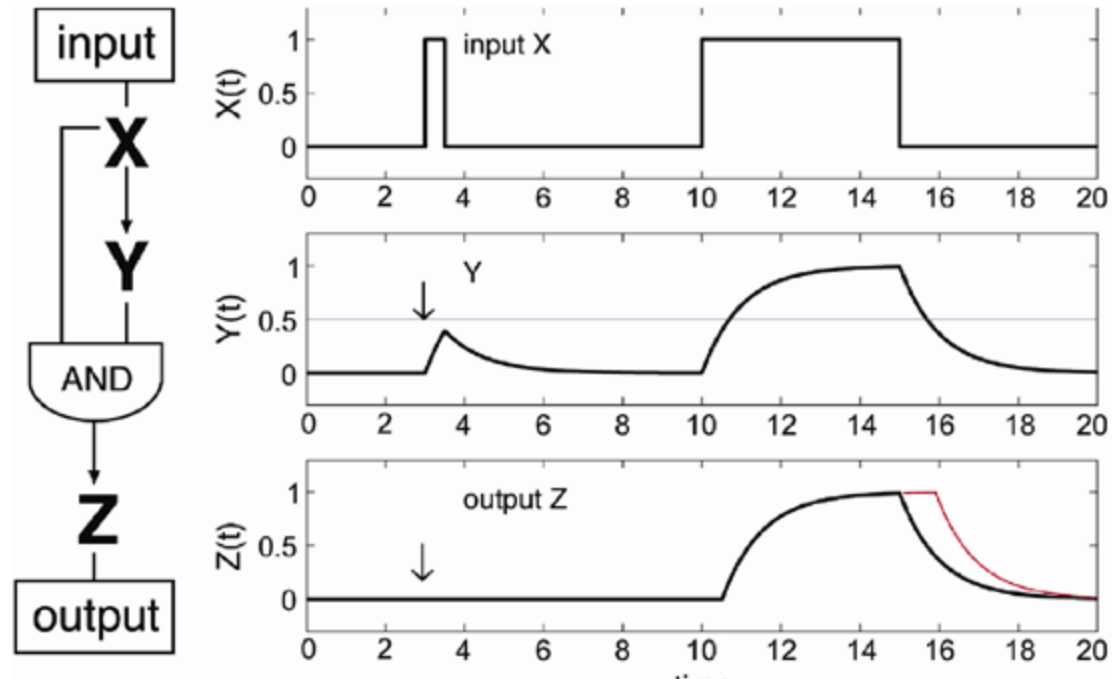
Network	N_{real}		N_{real}		N_{real}	
Gene regulation (transcription)		Feed-forward loop		Bi-fan		
Neurons		Feed-forward loop		Bi-fan		Bi-parallel
Food webs		Three chain		Bi-parallel		
Electronic circuits (forward logic chips)		Feed-forward loop		Bi-fan		Bi-parallel
Electronic circuits (digital fractional multipliers)		Three-node feedback loop		Bi-fan		Four-node feedback loop
World Wide Web		Feedback with two mutual dyads		Fully connected triad		Uplinked mutual dyad



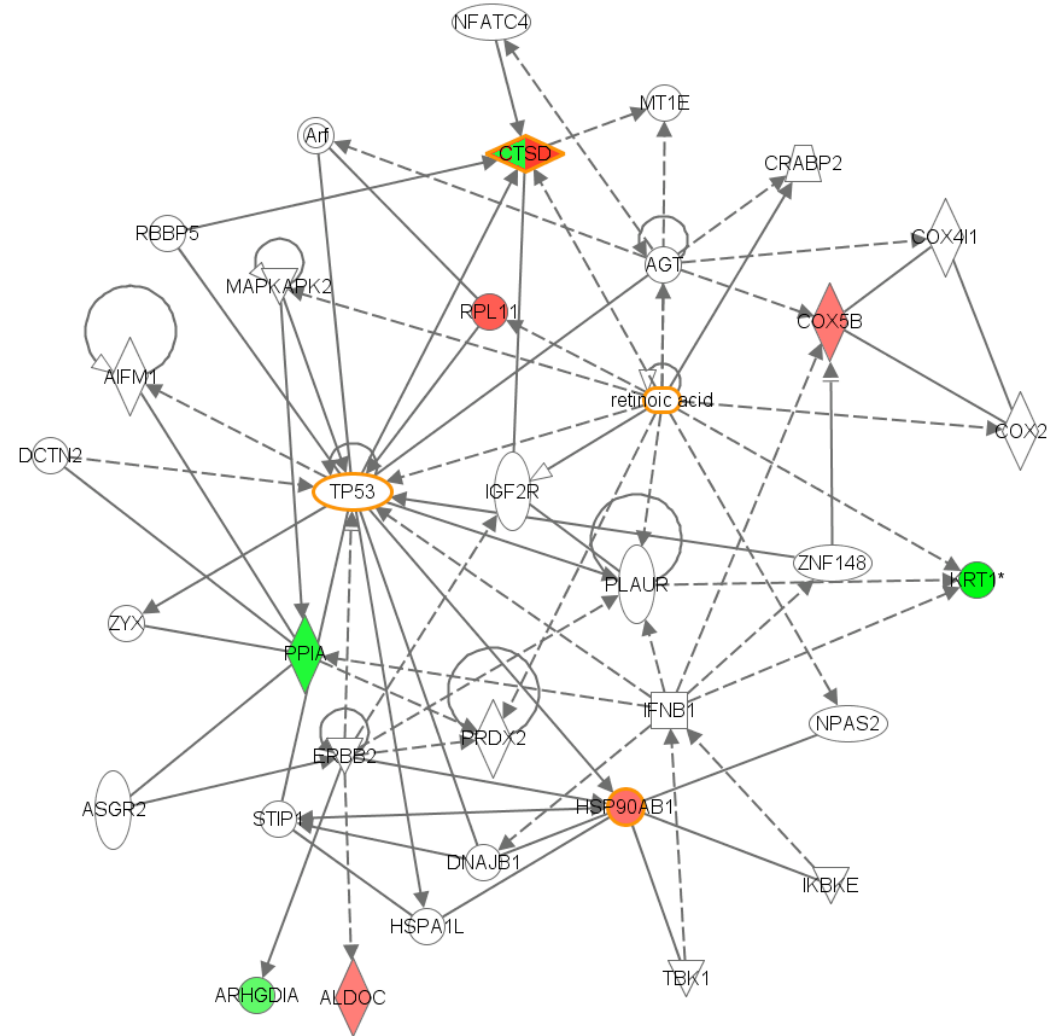
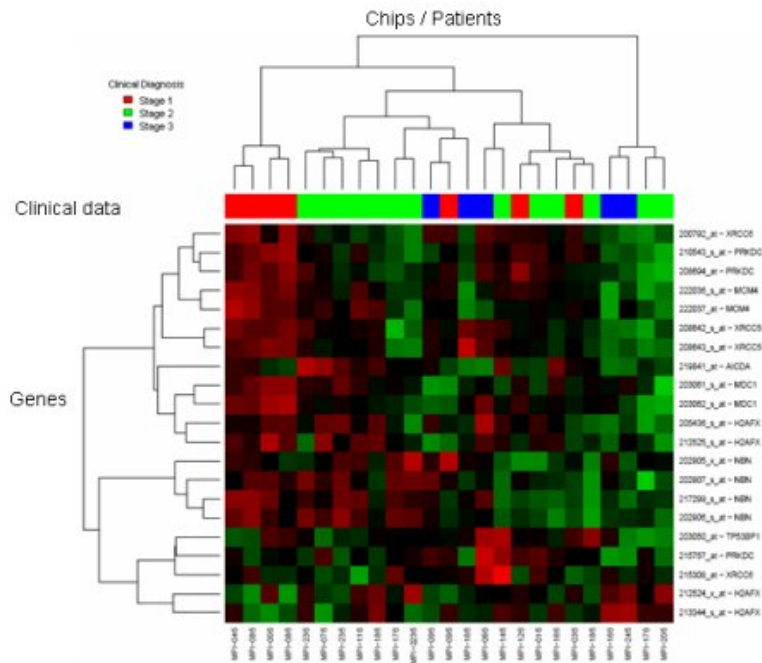
Dynamic features of the SIM motif



Dynamic features of the coherent feedforward loop

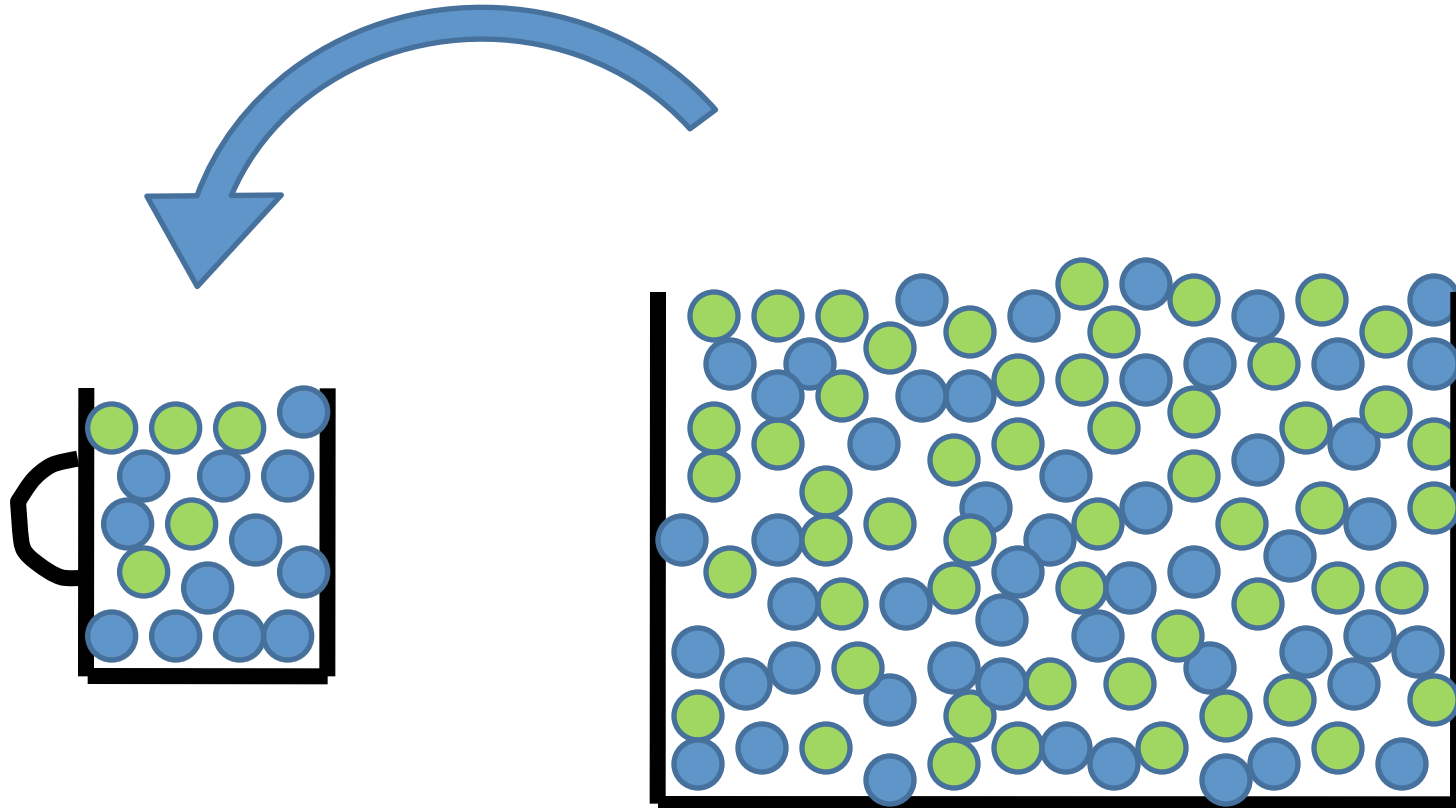


Why Pathway Analysis?



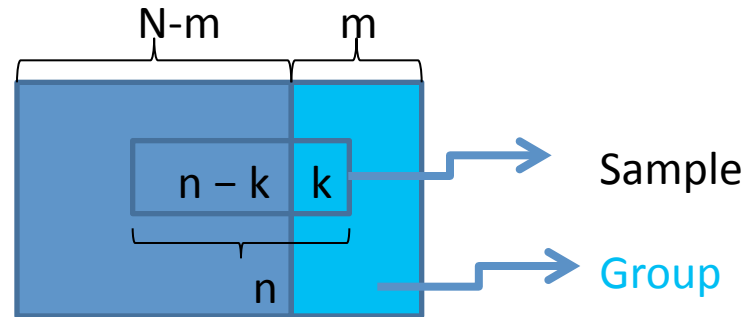
8 list members in 34 node network

Enrichment Analysis



Enrichment Statistics

In probability theory and statistics, the *hypergeometric distribution* is a discrete probability distribution that describes the number of successes in a sequence of n draws from a finite population without replacement.



A database contains N genes of which m belong to a **group** (e.g., a network) The hypergeometric distribution describes the probability that exactly k objects belong to this group in a **sample** of n genes observed in an experiment:

$$f(k; N, m, n) = \frac{\binom{m}{k} \binom{N-m}{n-k}}{\binom{N}{n}}.$$

Hypergeometric distribution

The probability that k **or more** objects belong to this group in a sample of n genes observed in an experiment is the sum of the probabilities for $k, k+1, \dots, n$:

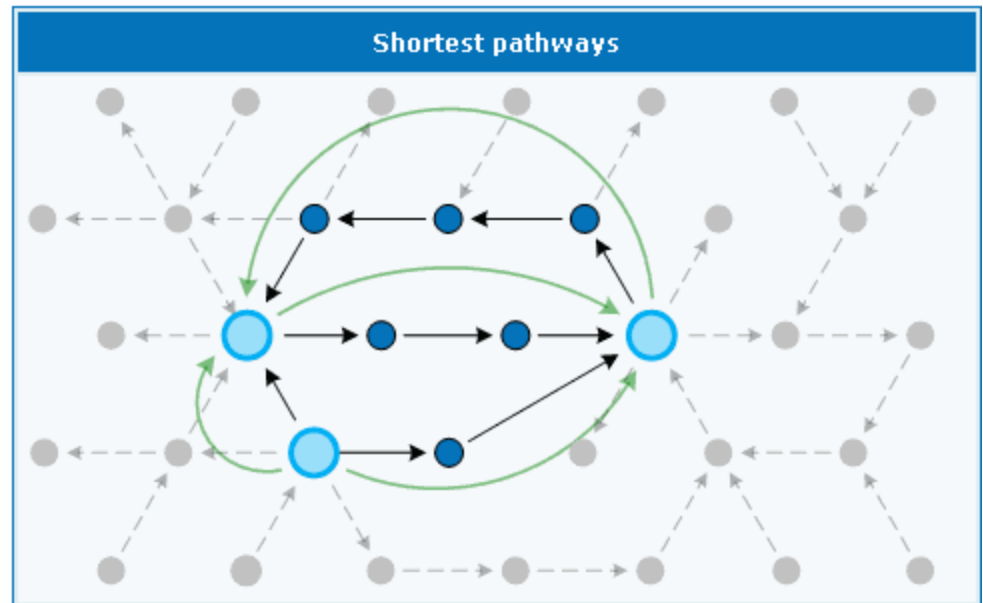
$$P = \sum_k^n \frac{\binom{m}{k} \binom{N-m}{n-k}}{\binom{N}{n}}$$







The hypergeometric test is identical to the corresponding one-tailed version of Fisher's exact test

Note: for small k , it has been suggested to use $k-1$ to get a more robust estimate for P called EASE score. Hosack et al. 2003, Genome Biol.

How to create a dense network

- Assumption: Biological function involves locally dense networks
- A dense network can be created by adding genes along the shortest paths that connect members of your gene list (using Dijkstra's Shortest Path Algorithm).

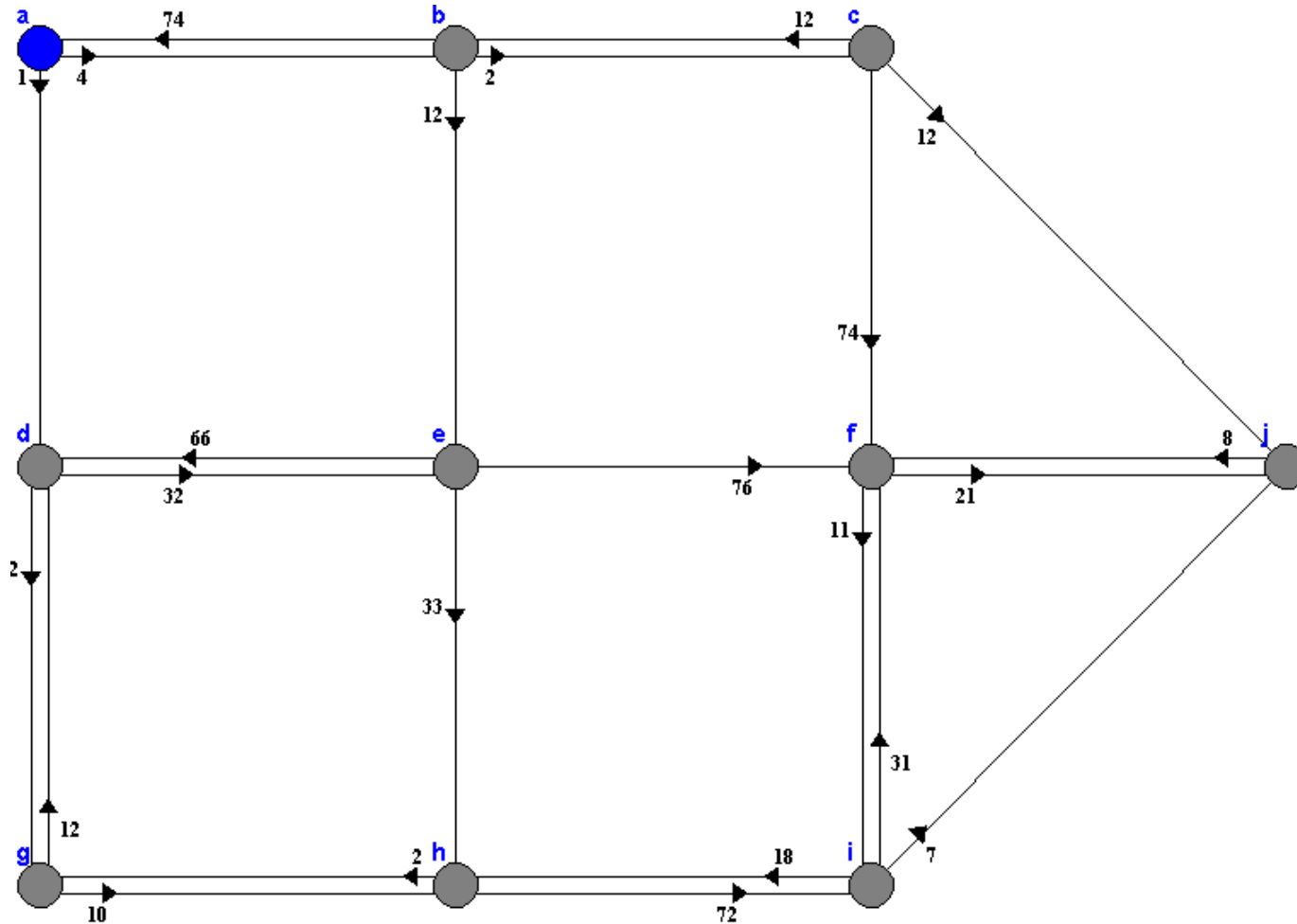


Description	Root nodes	Other nodes	Links
Object included in network			
Object not included in network			

Dijkstra's algorithm

- A graph search algorithm that solves the single-source shortest path problem for a graph with non negative edge path costs, outputting a shortest path tree.

Dijkstra's Shortest Path Algorithm



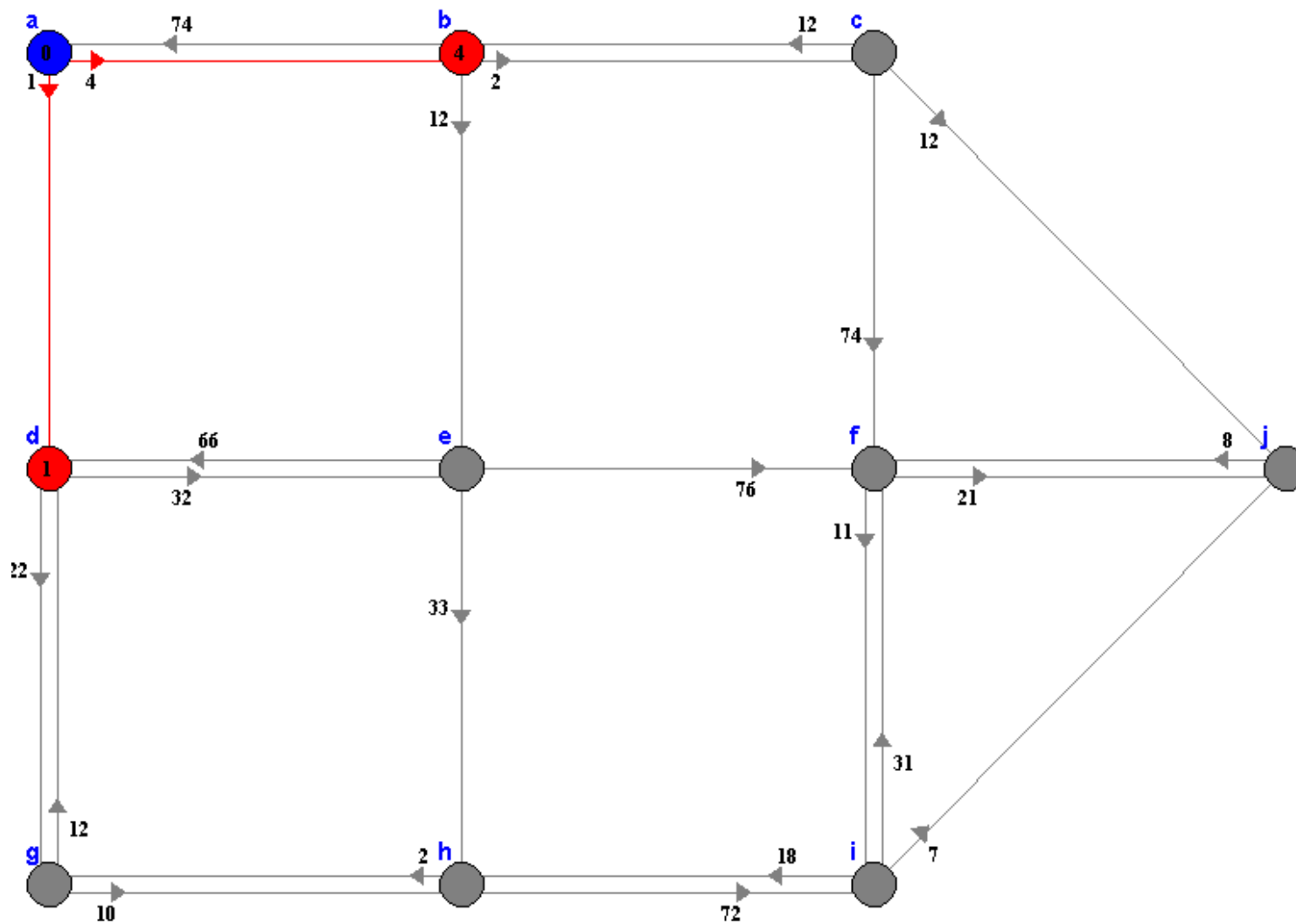
Algorithm running: red arrows point to nodes reachable from the startnode.

v:

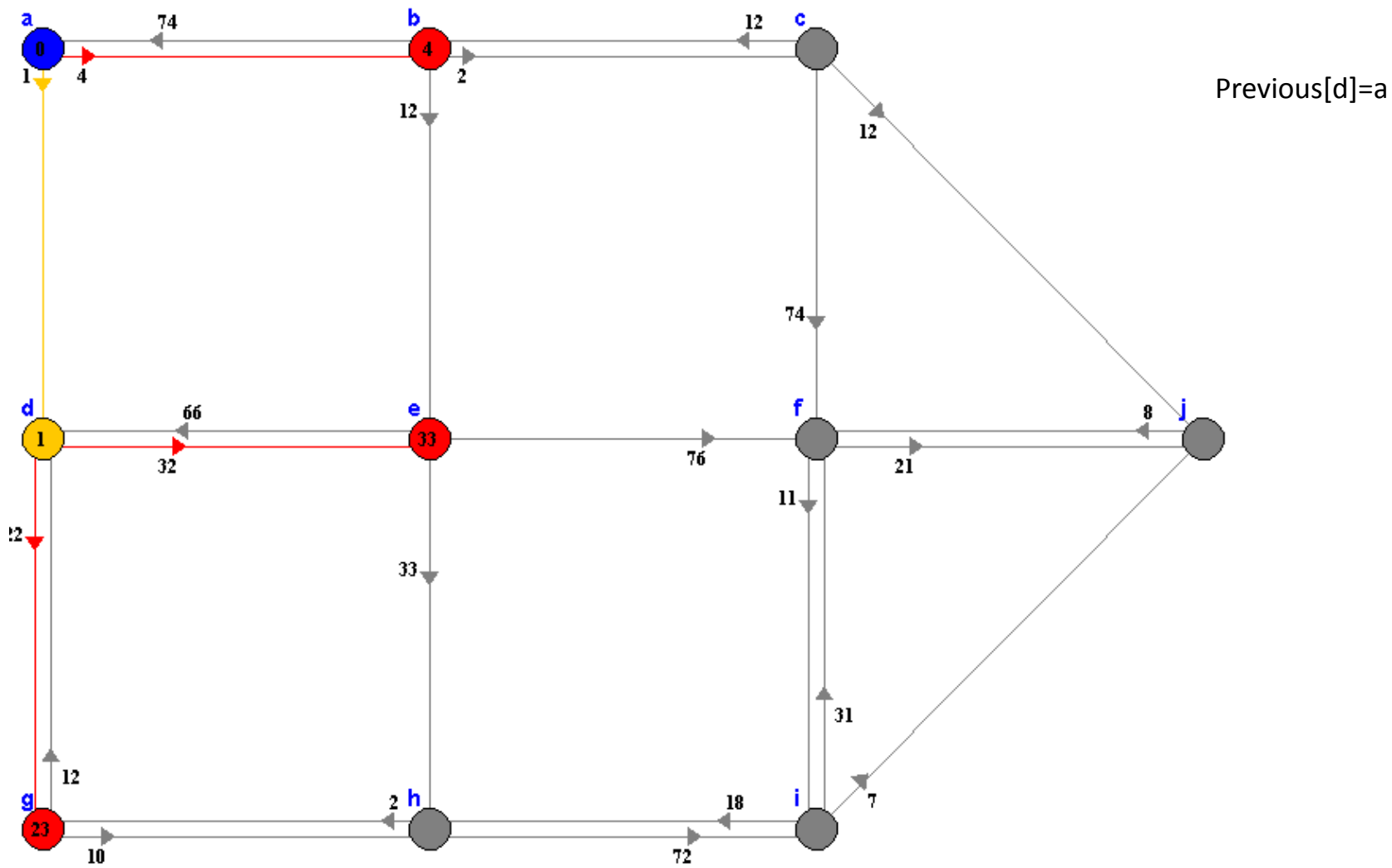
The distance to: $b=4$, $d=1$. Node d has the minimum distance.

Any other path to d visits another red node, and will be longer than 1.

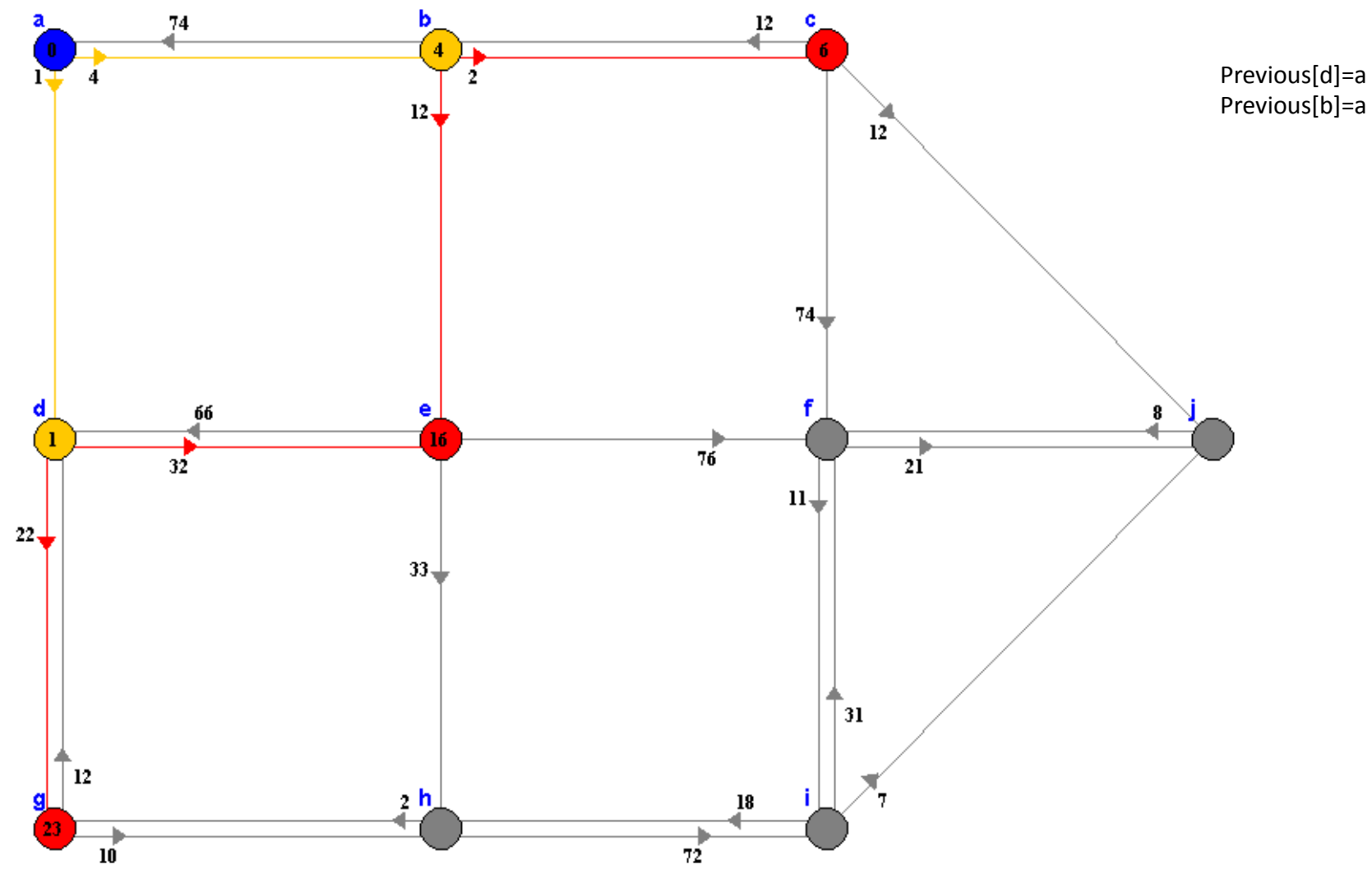
Node d will be colored orange to indicate 1 is the length of the shortest path to d.



Step 2: Red arrows point to nodes reachable from nodes that already have a final distance.
 The distance to: b=4, e=33, g=23. Node b has the minimum distance.
 Any other path to b visits another red node, and will be longer than 4.
 Node b will be colored orange to indicate 4 is the length of the shortest path to b.



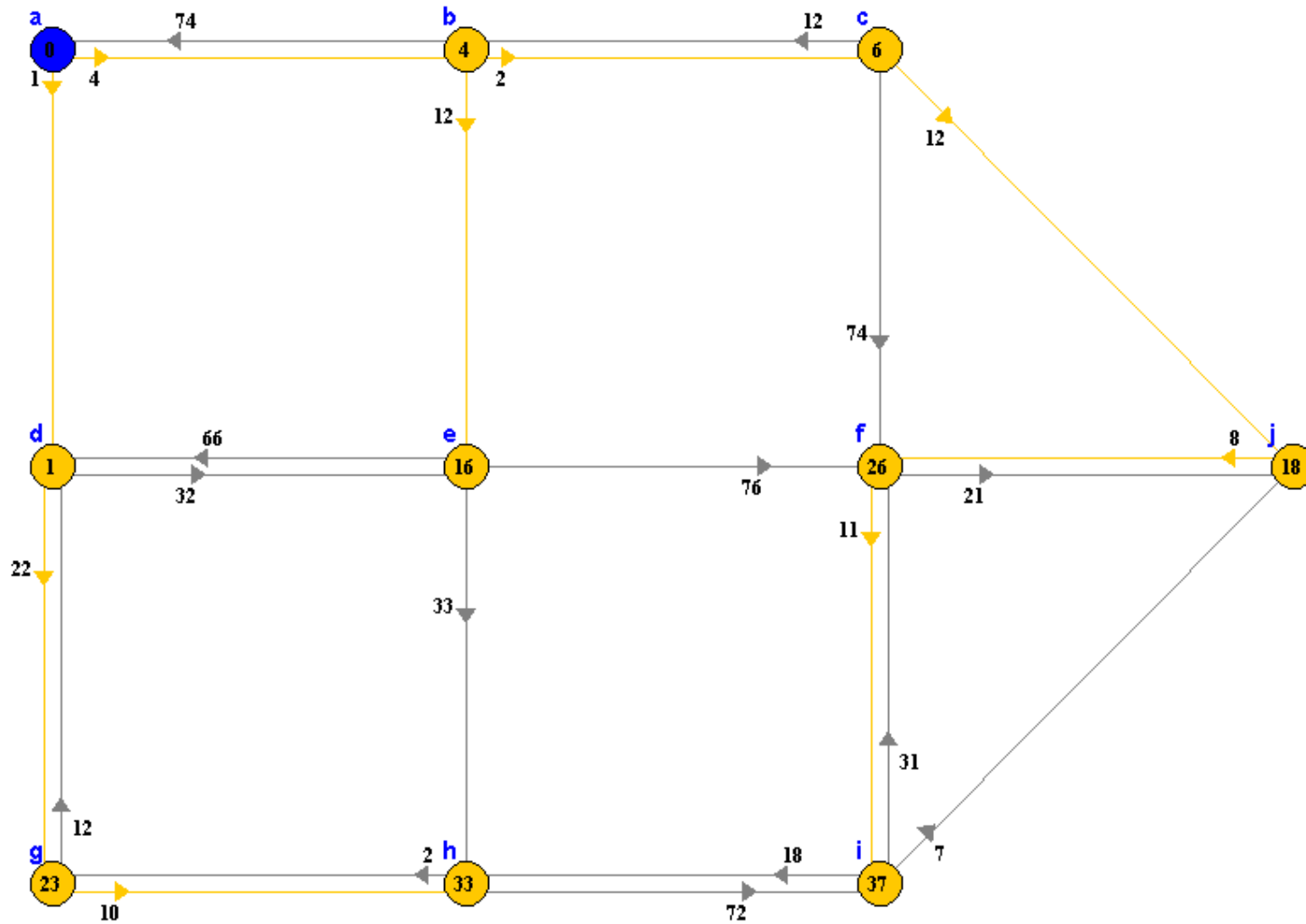
Step 3: Red arrows point to nodes reachable from nodes that already have a final distance.
 The distance to: c=6, e=16, g=23. Notice that the distance to e, has changed!
 Node c has the minimum distance.
 There are no other arrows coming in to c.
 Node c will be colored orange to indicate 6 is the length of the shortest path to c.



- A few more steps ...

IN:

Algorithm has finished, follow orange arrows from startnode to any node to get the shortest path to the node. The length of the path is written in the node.
press <RESET> to reset the graph, and unlock the screen.



Previous{d}=a
Previous{b}=a
Previous{c}=b
Previous{e}=d
Previous{j}=c
Previous{g}=d
Previous{f}=c
Previous{h}=g
Previous{i}=f

Dijkstra's Shortest Path Algorithm

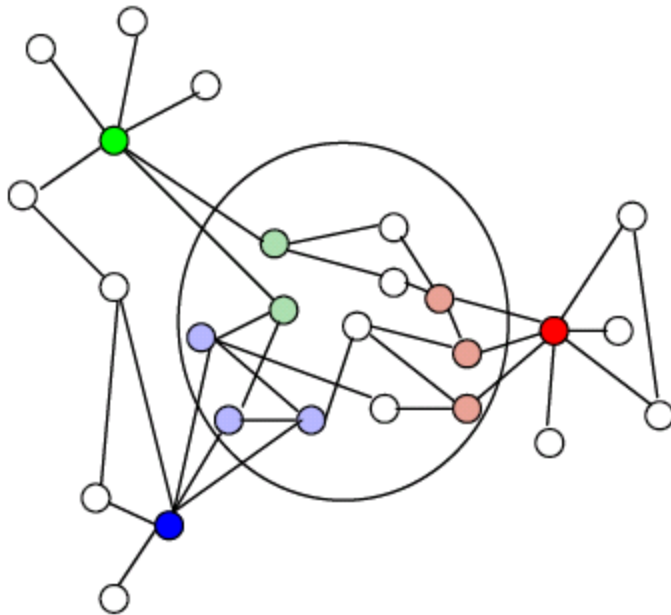
(comments)

- The previous algorithm shows the shortest path from a given node (a) to all the other nodes in the graph.
- It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined
- If there are more than one shortest paths (say going through nodes r and s) between source node a and target node t, then the algorithm would be repeated to find the shortest paths from r to t and s to t.

Network creation algorithm

(Ingenuity Pathway Analysis)

Assumption: Biological function involves locally dense networks



- 1) Sort genes in focus list so we can start with most interconnected genes.
- 2) Select most connected gene. Add to it other genes one at a time, in a way as to have the most connected pathway, until maximum network size (35) is reached.
- 3) If maximum size cannot be reached, combine smaller networks into larger ones.
- 4) If network size is < 35, add some more genes to provide biological context.
- 5) User may combine several small networks to one of < 210 nodes.

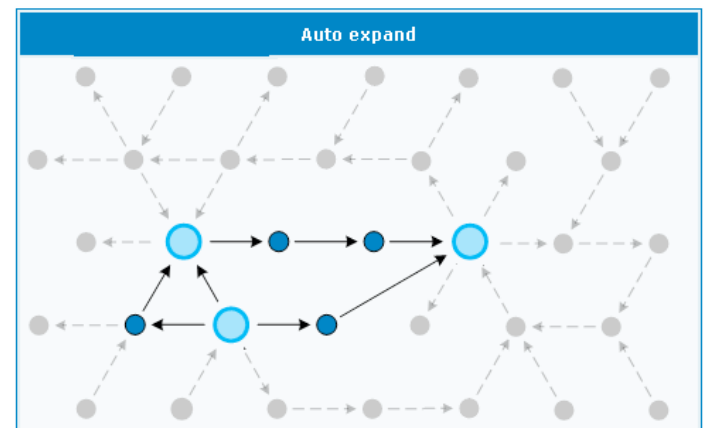
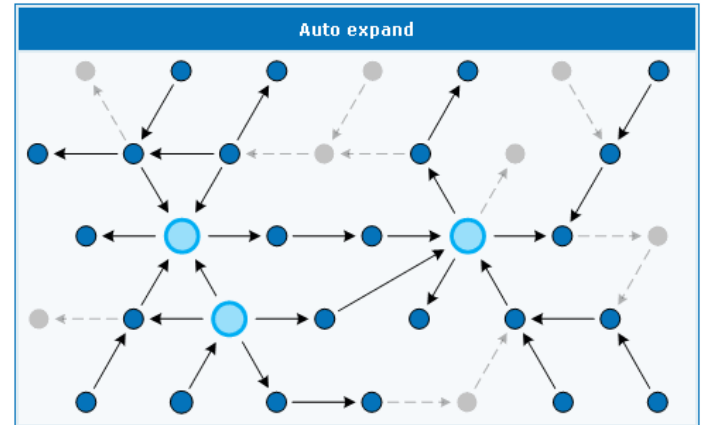
Color	Genes in neighborhood	Genes in network	Specific Connectivity
Red	8	3	0.18
Green	7	2	0.12
Blue	7	3	0.19

← most connected

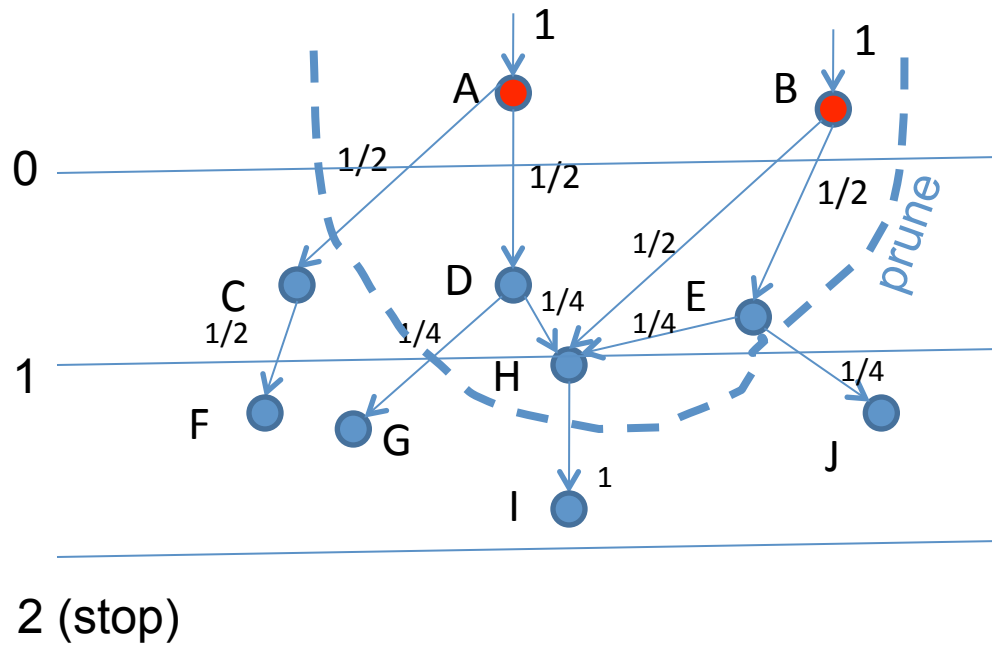
How to expand a network

Auto-expand algorithm

- Builds sub-networks around every object from the uploaded set consisting of nearest neighbors.
- The expansion halts when the sub-networks intersect.
- The objects that do not contribute to connecting sub-networks are automatically truncated and there is no user control over the size of the network.



Auto-expand algorithm

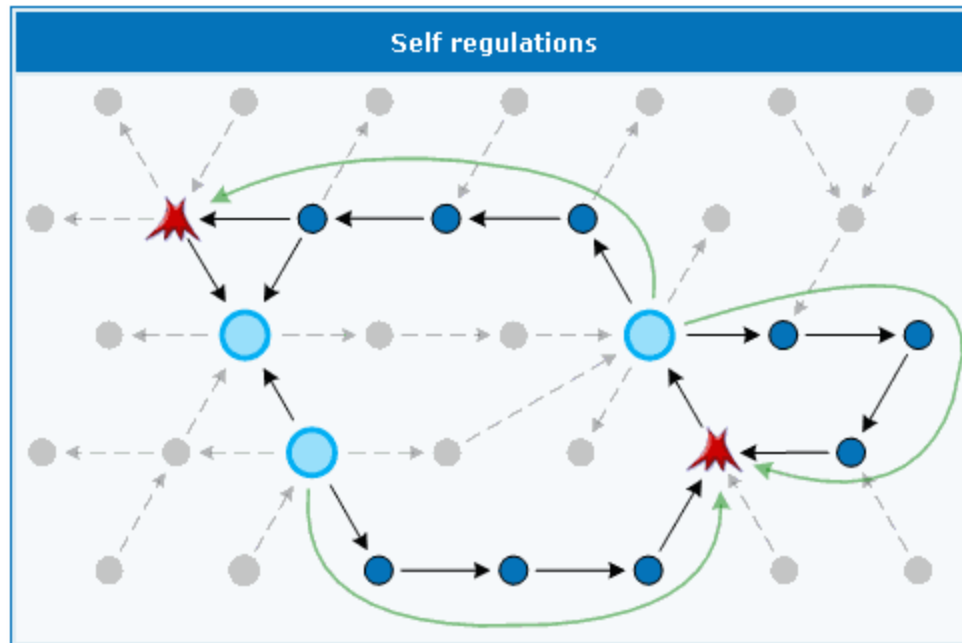


Node	flow	origin	dest
A	1	A	H
B	1	B	H
H	1	(A, B) =H	H
I	1	H	
C	1/2	A	-
D	1/2	A	
E	1/2	B	H
F	1/2	A	
G	1/4	A	

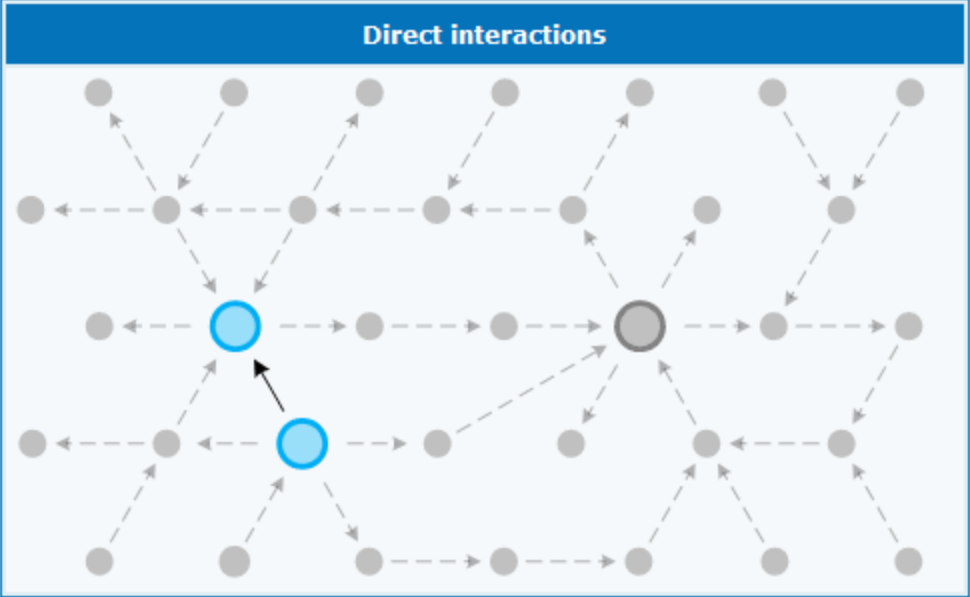
prune → A
B
D
E
H

(Number of new nodes limited to a preset max)

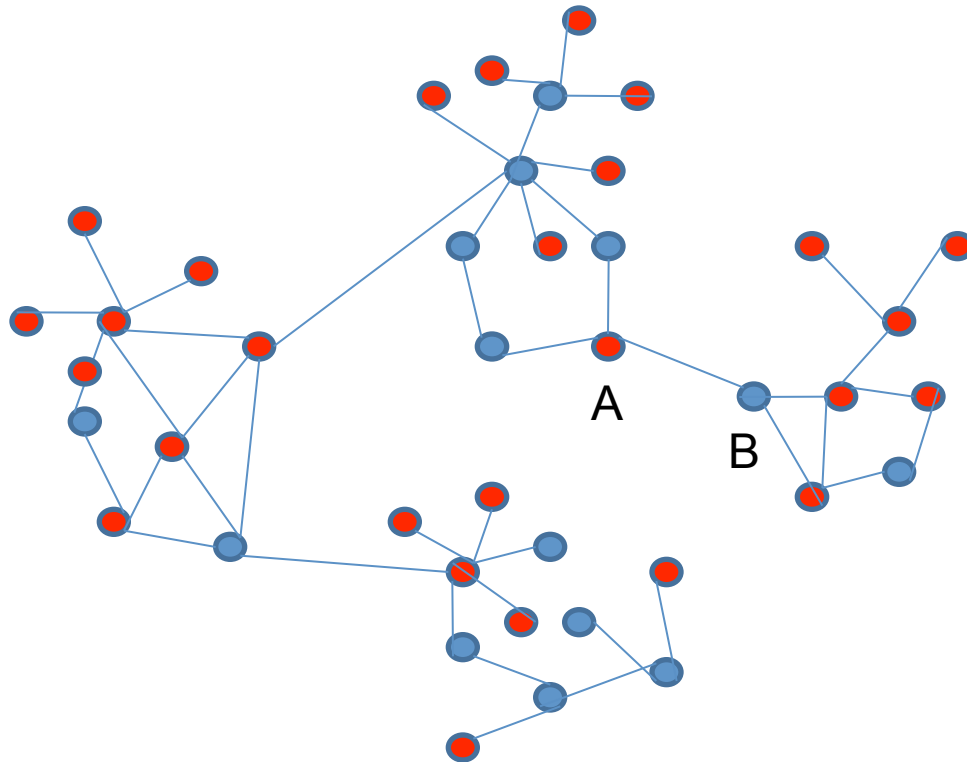
Modification of Shortest Path Algorithm



require that a transcription factor be included in the path.



Community Detection



Girvan-Newman algorithm

The algorithm's steps for community detection are summarized below:

- 1) The betweenness of all existing edges in the network is calculated first.
- 2) The edges with the highest betweenness are removed.
- 3) The betweenness of all edges affected by the removal is recalculated.
- 4) Steps 2 and 3 are repeated until no edges remain.