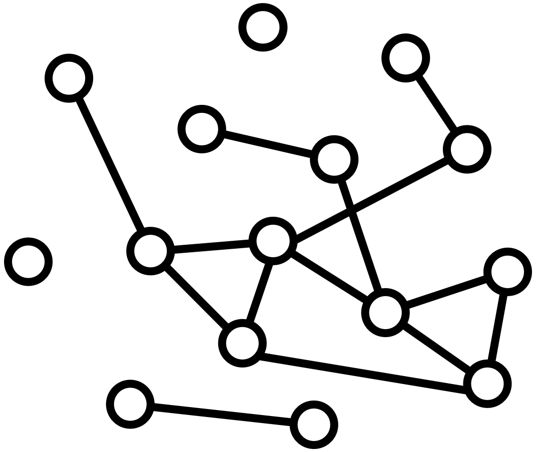
Simulation I:
Dynamics of Networks

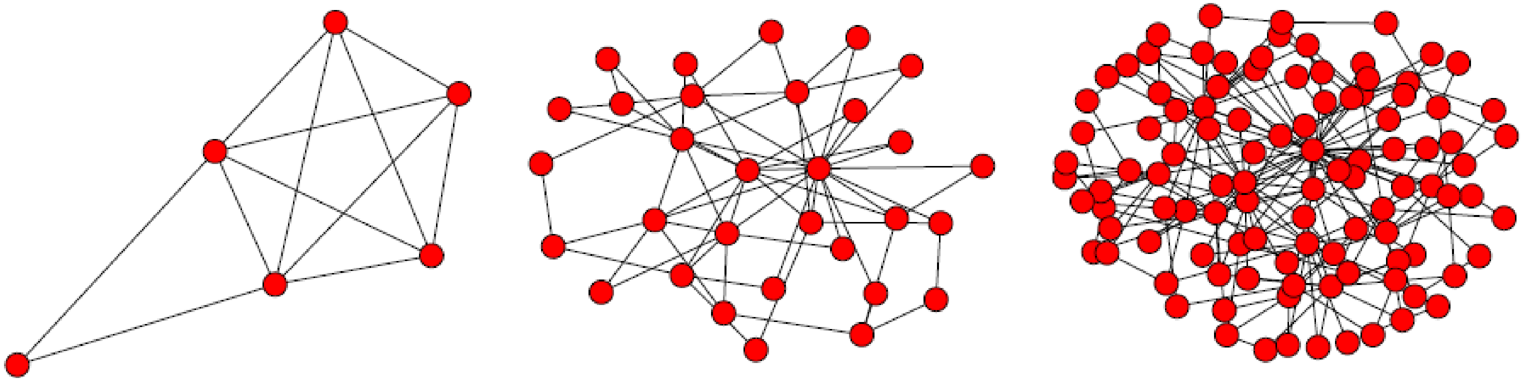


Hiroki Sayama
sayama@binghamton.edu

Modeling and Simulation of Dynamical Networks

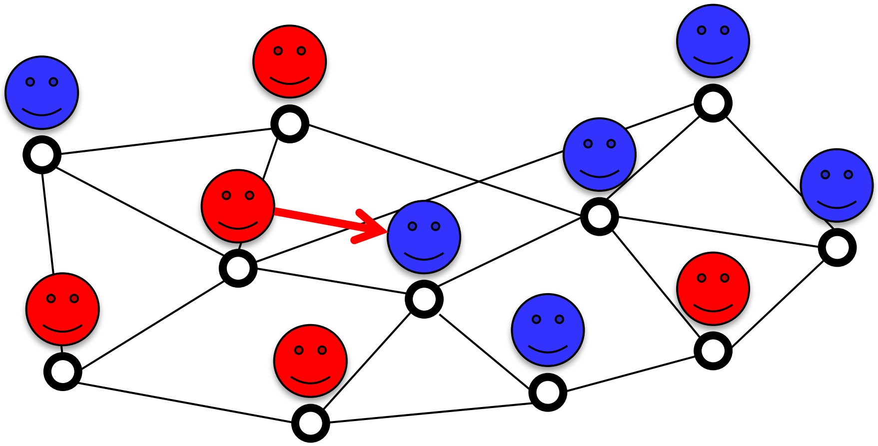
Dynamics of networks

- Dynamic growth and transformation of network topologies
  - Social network formation
  - Growth of the Internet and WWW
  - Growth of scientific citation networks



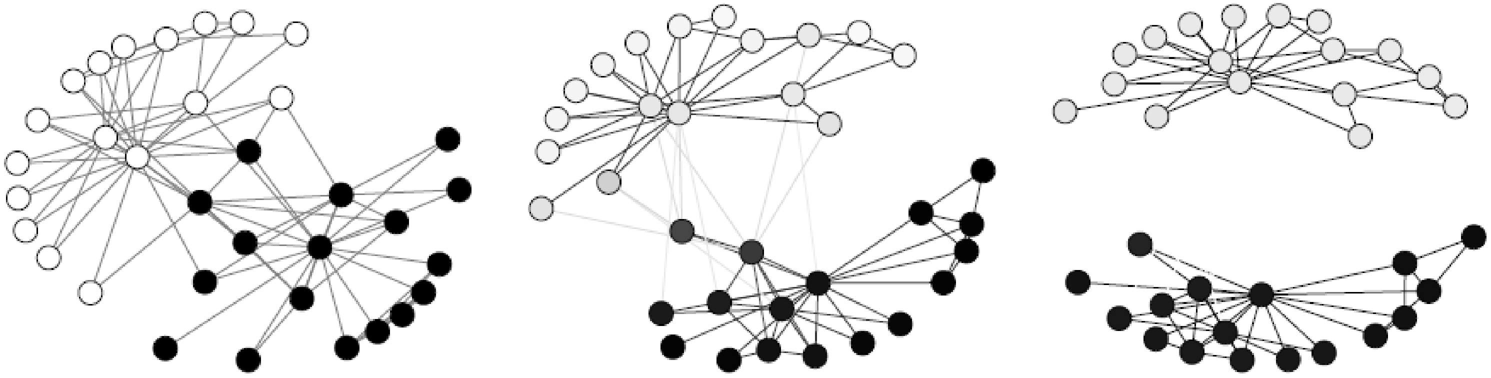
Dynamics on networks

- Dynamic state changes taking place on a static network topology
  - Gene/protein regulatory networks
  - Population dynamics on food webs
  - Spread of disease/opinion/failure



Adaptive Networks

- Complex networks whose states and topologies co-evolve, often over similar time scales
  - Link (node) states adaptively change according to node (link) states



Modeling and Simulation of Dynamics of Networks

Dynamics of networks

- Dynamic growth and transformation of network topologies
  - Social network formation
  - Food web formation over ecological/ evolutionary time scales
  - Growth of the Internet and WWW
  - Growth of scientific citation networks
  - Effects of node/link removal or rewiring

Network Percolation

Percolation in random networks

- Number of connected components decreases with increasing link probability

- Above a critical probability p_c, a giant connected component emerges

Giant connected component

- Largest connected component whose size (relative to the total number of nodes N) remains positive even if N is very large

$$\lim_{N \to \infty} |GCC|/N > 0$$

  - If LCC is not giant, $\lim_{N \to \infty} |LCC|/N = 0$

Exercise

- Simulate the emergence of a giant connected component by randomly introducing edges one by one

- Monitor the process and see how the giant connected component emerges

Exercise

- Plot (1) the size of the largest connected component and (2) the number of connected components of a random network made of 10,000 nodes over varying p

Review: Degree distribution of ER networks

- Degree distribution of an ER random network is given by a binomial distribution:

$$P(k) = {}_{N-1}C_k \, p^k \, (1-p)^{N-1-k}$$

- With large N (with fixed Np), it approaches a Poisson distribution:

$$P(k) \sim (Np)^k \, e^{-Np} \, / \, k!$$

Percolation threshold

- Let s be the probability for a node to belong to LCC (i.e., |LCC| = sN)
- Degree dist.: $P(k) = (Np)^k e^{-Np} / k!$
- Probability for a node to be separated from LCC is given by:

$$1-s = \Sigma_{k=0\sim\infty} P(k) (1-s)^k$$

  - $(1-s)^k$ is the probability for all k neighbors to be separated from LCC

Exercise

- Using the following equations, show that s can take positive values if and only if $\langle k \rangle = Np > 1$

$$P(k) = (Np)^k e^{-Np} / k!$$

$$1-s = \Sigma_{k=0\sim\infty} P(k) (1-s)^k$$

Exercise

- Choose two link probabilities, one below and one above p_c
- Create ER random networks for each probability with varying N, and see how the size of their LCCs scales along N

Exercise

- If $1-s < 1/N$, that means all nodes are essentially included in LCC, and thus the network is made of just one connected component

- Obtain the critical threshold of $\langle k \rangle$ above which this occurs

Edge Rewiring and Growth

Exercise: Rewiring for “small-world”

- Create a ring-shaped network made of n nodes; connect each node to k nearest neighbors
- Visualize the network by coloring nodes using their closenesses
- Randomly rewire edges one-by-one
- Monitor what happens to the network topology and node colors

Exercise: Preferential attachment

- Simulate the growth process of the Barabasi-Albert network growth model with m = 1, m = 3 and m = 5

- See how the process is affected by variation of this parameter

Exercise

- Modify the simulation code so that the node selection preference is:
  - Independent of the node degree
  - Proportional to the square of the node degree
  - Inversely proportional to the node degree
- Conduct simulations and compare the resulting network topologies

Exercise

- Modify the simulation code of the preferential attachment model so that a node whose degree exceeds a certain capacity limit splits into two (and each node inherits about half of the original connections)
- Conduct simulations and compare the resulting network topologies

Robustness and Vulnerability of Complex Networks

Robustness and vulnerability

- How do these networks respond to dynamic topological changes caused by external forces?
  - Input: Removal of nodes
  - Output: Changes in characteristic path length and connectivity

Two types of node removals

- Error: Random removal of nodes
  - Occurs stochastically
  - Same error probability for all nodes
- Attack: Selective removal of most connected nodes
  - Occurs deterministically
  - The attacker knows network hubs

Examples in real-world networks

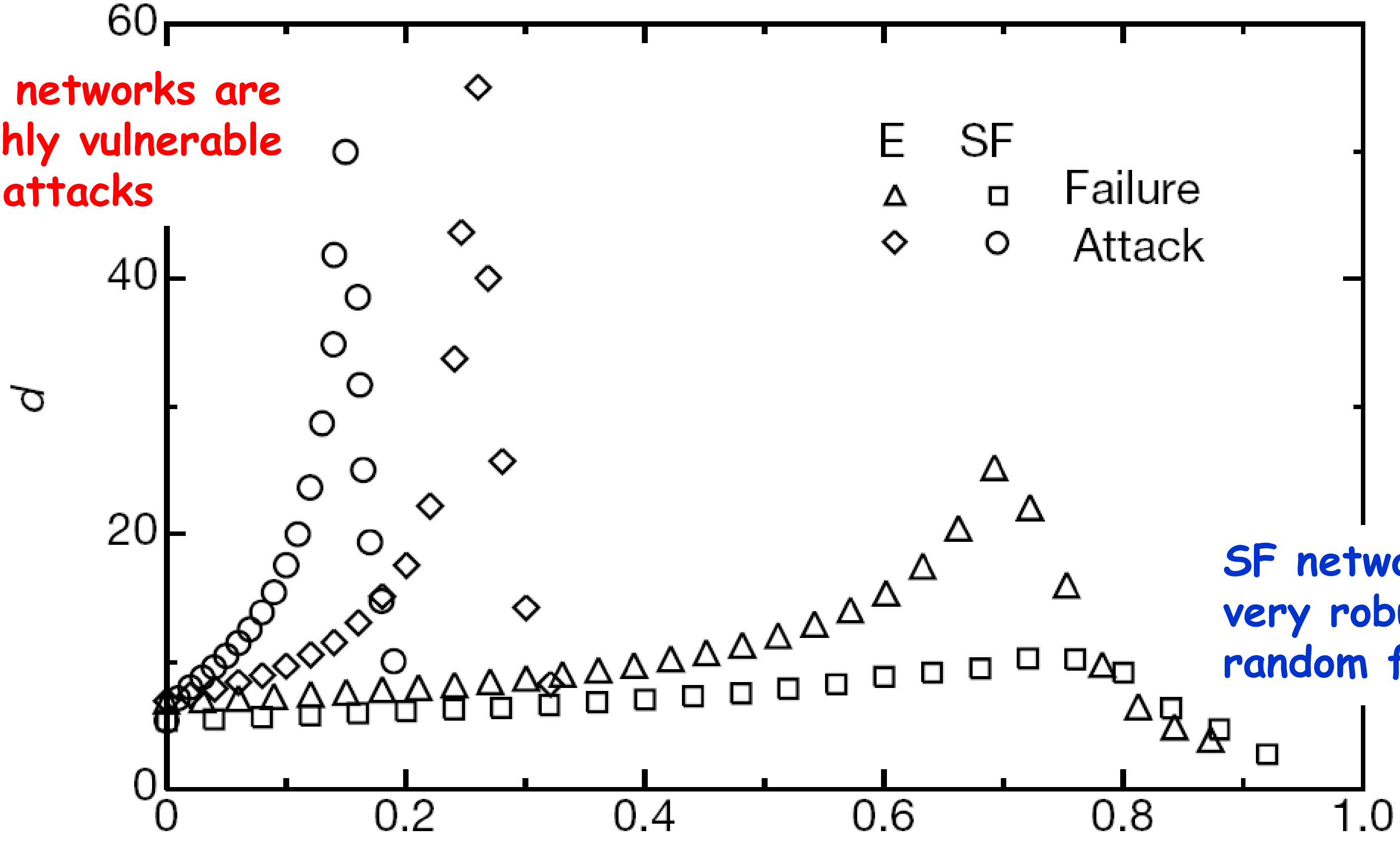
- **WWW:**
  - Error ⇒ Occasional server breakdown
  - Attack ⇒ Server breakdown due to DoS etc.
- **Warfare:**
  - Error ⇒ Accidental local actions
  - Attack ⇒ Strategic actions to hit the central core of opponents
- **Marketing:**
  - Error ⇒ Indiscriminate direct mail, spam
  - Attack ⇒ Targeting on influential customers

Robustness and vulnerability of scale-free networks

- R. Albert, H. Jeong & A.-L. Barabasi, Error and attack tolerance of complex networks, Nature 406:378-382, 2000.
  - Considered the effects of random errors & targeted attacks on scale-free networks (both simulated and actual ones)

Change of diameter (experiments with artificially generated networks)

Characteristic path length



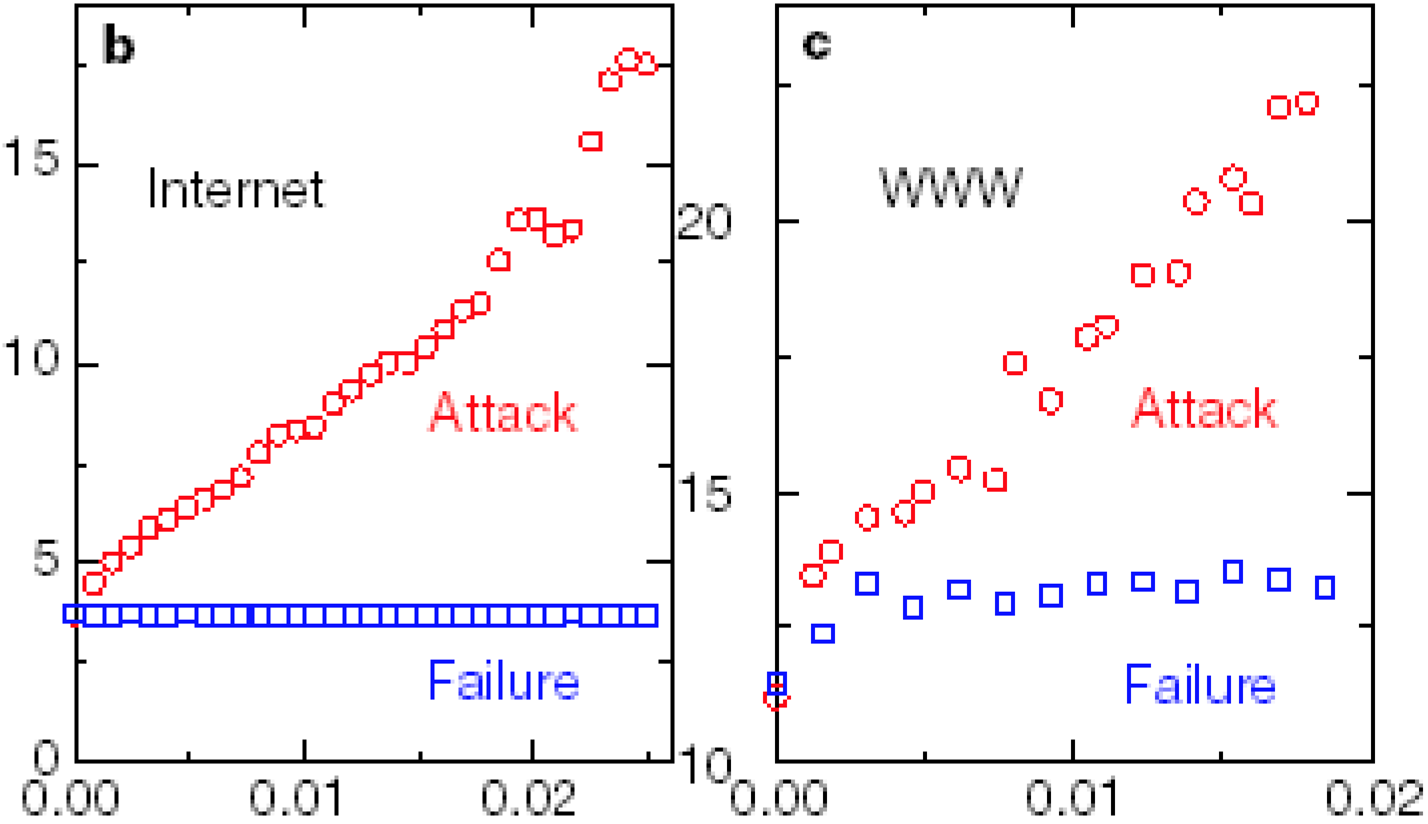
E	SF	
△	□	Failure
◇	○	Attack

SF networks are highly vulnerable to attacks

SF networks are very robust to random failures

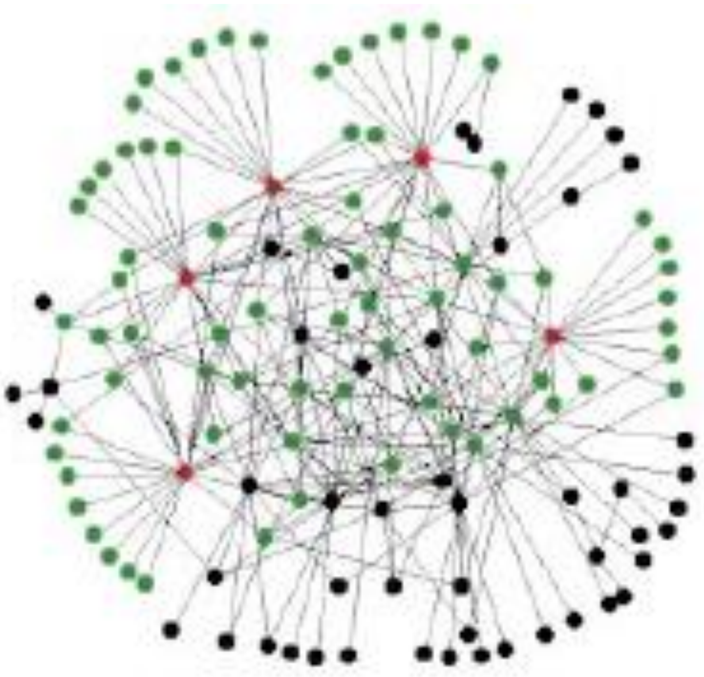
Change of diameter (experiments with networks based on real data)

Characteristic path length



Fraction of removed nodes

Why such robustness / vulnerability occurs?



A scale-free network has a few hub nodes and a lot of non-hub (mostly terminal) nodes

→Random errors are likely to hit non-hub nodes, causing only limited influence

→Attacks always hit hubs, causing great impacts on the whole

Fragmentation analysis

- If node removal goes on further, the network will eventually fall apart (fragmentation)

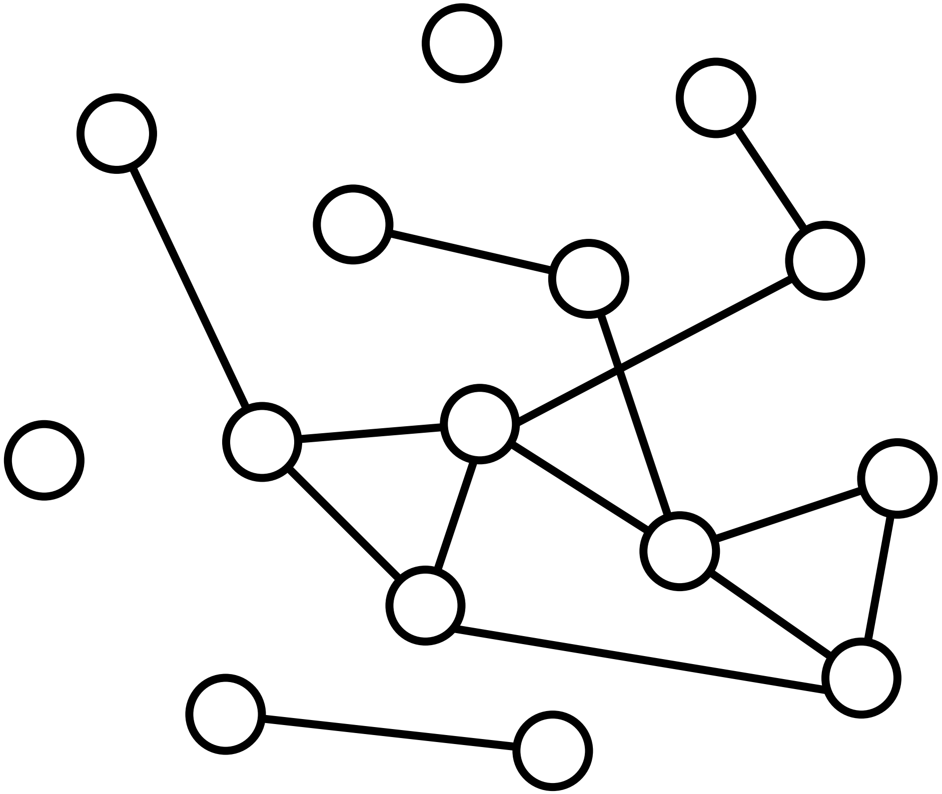
  One can detect such fragmentation by monitoring the following:

  S : Size of LCC

  <s> : Average size of all other smaller connected components

Exercise

- Calculate S and <s> for this graph



What S and <s> tells us

- While individual nodes drop out one by one from the largest connected body:

  → S decreases slowly, <s> ~ 1

- When the LCC falls apart:

  → S drops suddenly, <s> > 1

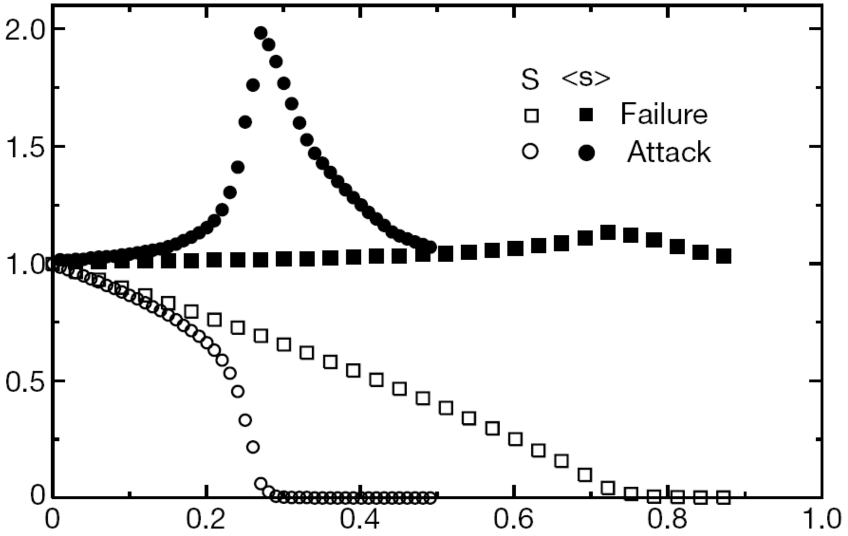
The latter indicates a critical moment of network fragmentation

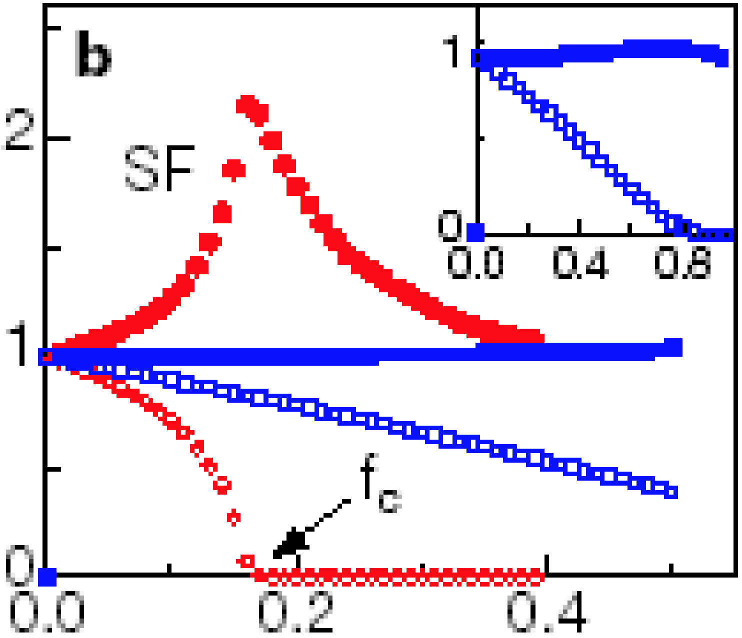
Fragmentation process (experiments with artificially generated networks)

Random

Scale-free

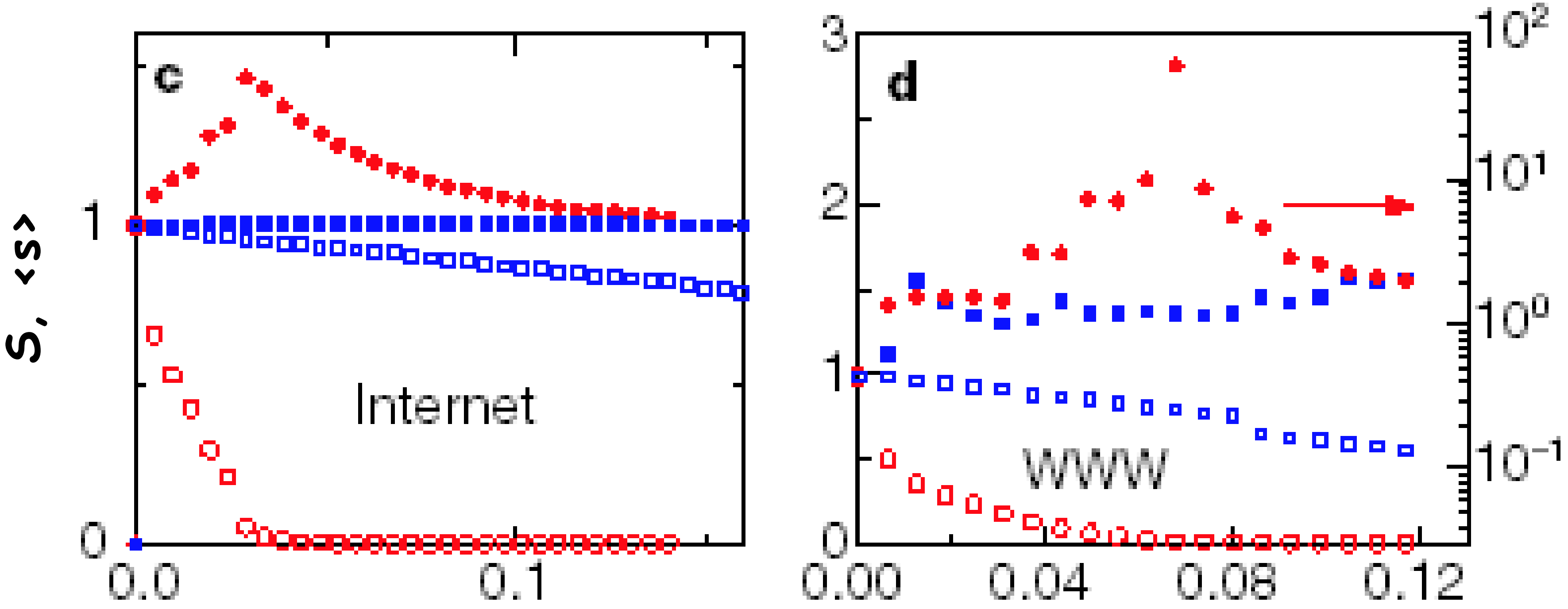
S, <s>





Fraction of removed nodes

Fragmentation process (experiments with networks based on real data)



Fraction of removed nodes

Exercise

- Replicate Albert, Jeong & Barabasi's network fragmentation experiments for ER random networks and BA scale-free networks

Exercise

- Conduct fragmentation analysis on Mark Newman's Political Blogs network data
- Try several different attack strategies and see which one would be most effective in disrupting the connectivity of the network

Network vulnerability

- Scale-free networks are robust to stochastic errors, but quite fragile against intentional attacks targeted to hubs

- This conclusion directly applies to real-world networks
  - DoS attacks to key servers, terrorisms at commercial hubs, etc…

- Then, what can we do?

A Potential Solution:

(1,0) networks

Our attempt

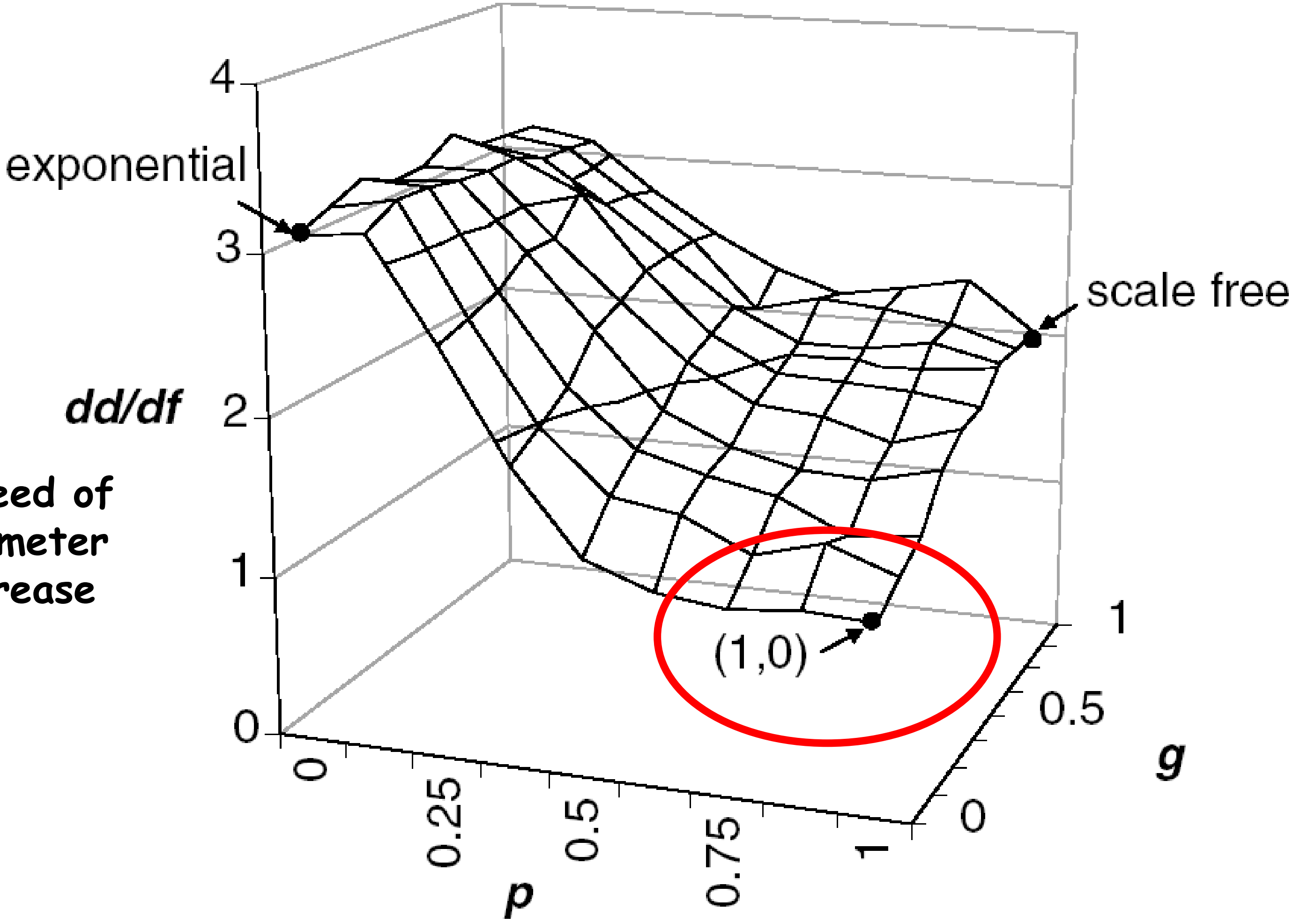
- B. Shargel, H. Sayama, I. R. Epstein & Y. Bar-Yam, Optimization of robustness and connectivity in complex networks, Phys. Rev. Lett. 90:068701, 2003.
  - Reconsidered the details of network development and proposed (1,0) networks that are more robust to both errors and attacks than pure scale-free networks

Two parameters for network development

- Preference parameter p ($0 \le p \le 1$)
  - Specifies how much the selection of nodes is affected by their degrees
- Growth parameter g ($0 \le g \le 1$)
  - Specifies the fraction of nodes that are added through the developmental process to the total number of nodes

Seeking a more robust network in the p-g parameter space

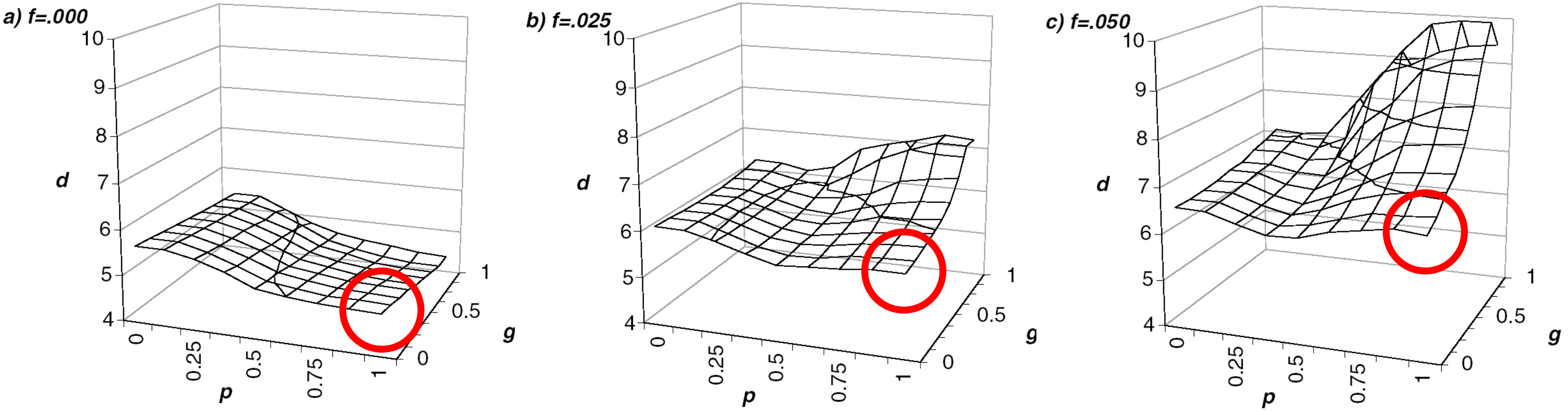
Response to random errors



Speed of diameter increase

Response to targeted attacks

d : characteristic path lengths



Intensity of attacks

Why robust to attacks?

- During the development of a (1,0) network, well-connected hubs can be connected to each other

→ Tightly connected clusters of hubs will emerge

(In the scale-free network growth with preferential attachment, isolated hubs cannot be connected to each other)

Exercise: Preferential attachment

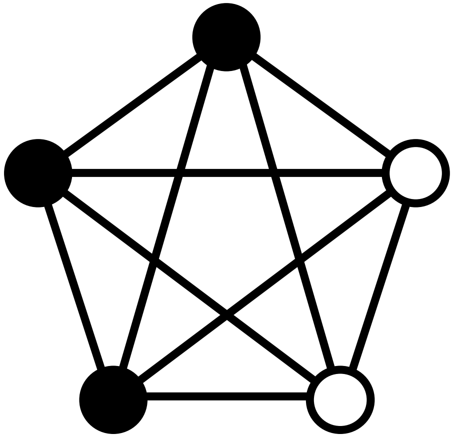
- Simulate the development process of the (p, g) network model

- See how the resulting network topology differs among the following
  - (p, g) = (0, 0) (random)
  - (p, g) = (1, 1) (preferential attachment)
  - (p, g) = (1, 0)

Implications

Networks that continue to reinforce connections between their internal parts can be more robust in many situations than other networks whose internal connections are enhanced only by the addition of newcomers

Simulation II:
Dynamics on Networks



Hiroki Sayama

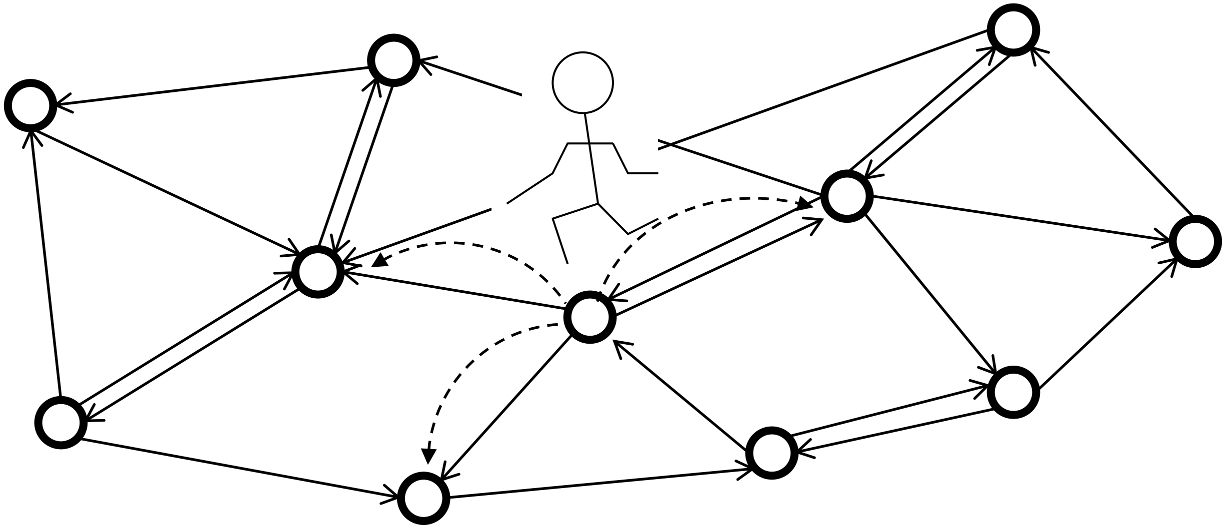
sayama@binghamton.edu

Dynamics *on* networks

- Dynamic state changes taking place on a static network topology
  - Regulatory dynamics on gene/protein networks
  - Population dynamics on ecological networks
  - Disease infection on social networks
  - Information/culture propagation on organizational/social networks

Simple example: Random walk on a network

- An agent (or a set of agents) moving on a network
- An agent jumps randomly to one of the neighbor nodes at each time step



Exercise

- Simulate random walk of an agent on a directed random network made of 50 nodes

- Count how many times each node was visited by the agent over time

TPM and asymptotic probability distribution (review)

- $|\lambda| <= 1$ for all eigenvalues
- If the original network is strongly connected (with some additional conditions), the TPM has one and only one eigenvalue 1 (no degeneration)

→ This is a unique dominant eigenvalue; the probability vector will converge to its corresponding eigenvector

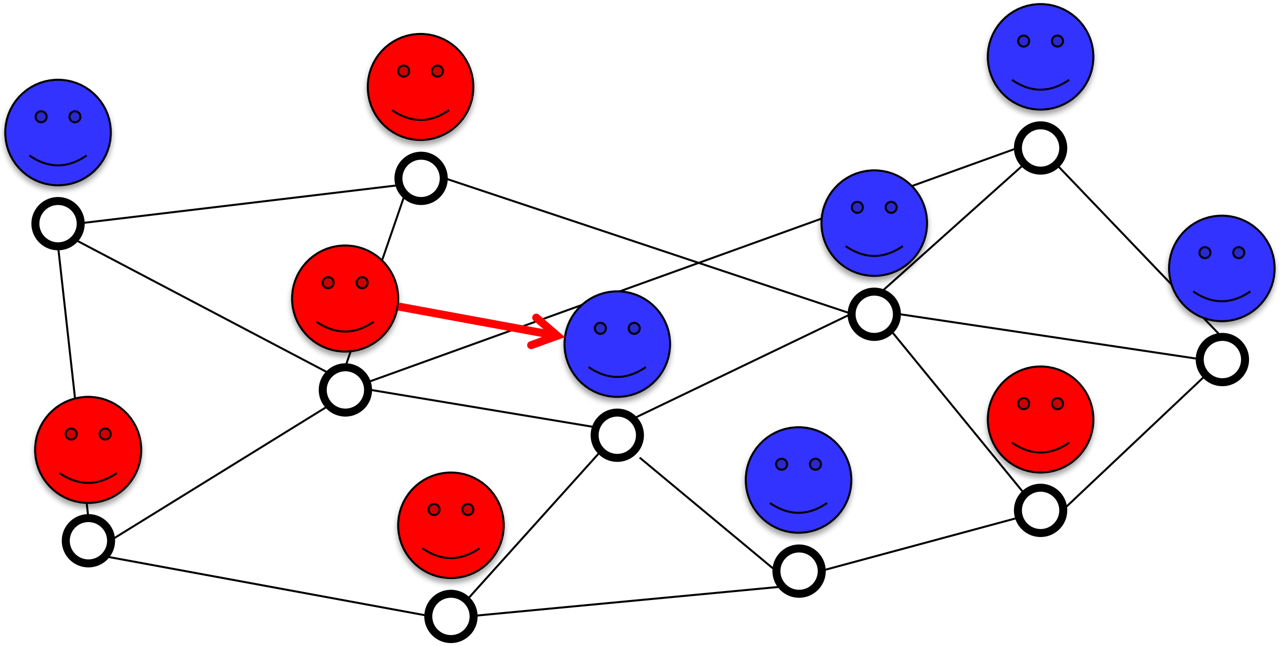
Exercise

- Construct the transition probability matrix of the random network used in the previous exercise
- Find its dominant eigenvector with $\lambda = 1$
- Compare the results with the previous "counting" results

Dynamics on Networks with Discrete Node States

Opinion formation (Voter model)

- A simple model of opinion formation in society
  - Opinions = discrete states



Three versions of voter models

- Original voter model
  - A randomly selected node copies the opinion of one of its neighbors
- Reverse voter model
  - A randomly selected node “pushes” its opinion into one of its neighbors
- Link-based voter model
  - An opinion is copied through a randomly selected link

Exercise

- Simulate the three different versions of the voter model (original, reverse and link-based) on a Barabasi-Albert scale-free network
- Compare the speed of opinion homogenization between the three models
  - Why different?

Epidemics (SIS/SIR model)

- Initially, a small fraction of nodes are infected by a disease
- If a susceptible node has an infected neighbor, it will be infected with probability p_i (per infected neighbor)
- An infected node will recover and become susceptible (SIS) or recovered (SIR) with probability p_r

Exercise

- Study the effects of infection/ recovery probabilities on the fixation of a disease on a random social network
  - In what condition will the disease remain within society?
  - In what condition will it go away?
  - Is the transition smooth, or sharp?

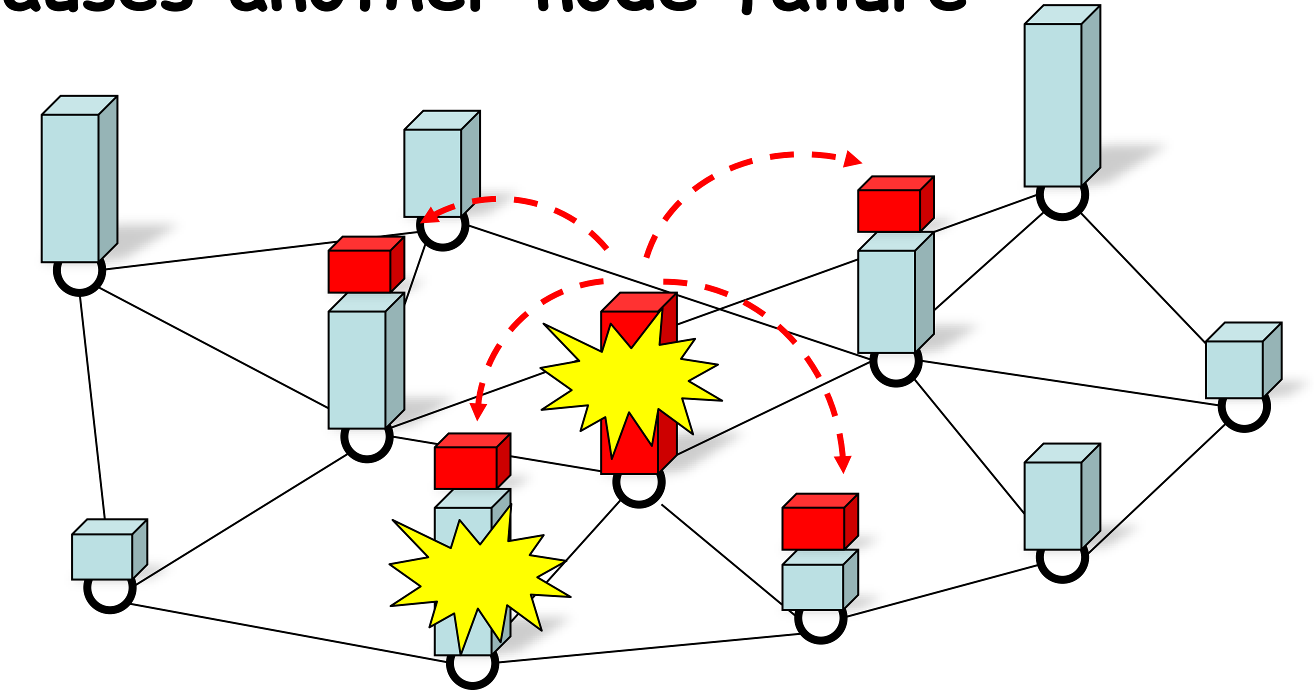
Exercise

- Do the same experiments with WS small-world networks and BA scale-free networks

- Compare their properties

Cascade of failure

- Load on a failing node is divided and distributed to its neighbors
- If the load exceeds capacity of each node, it causes another node failure

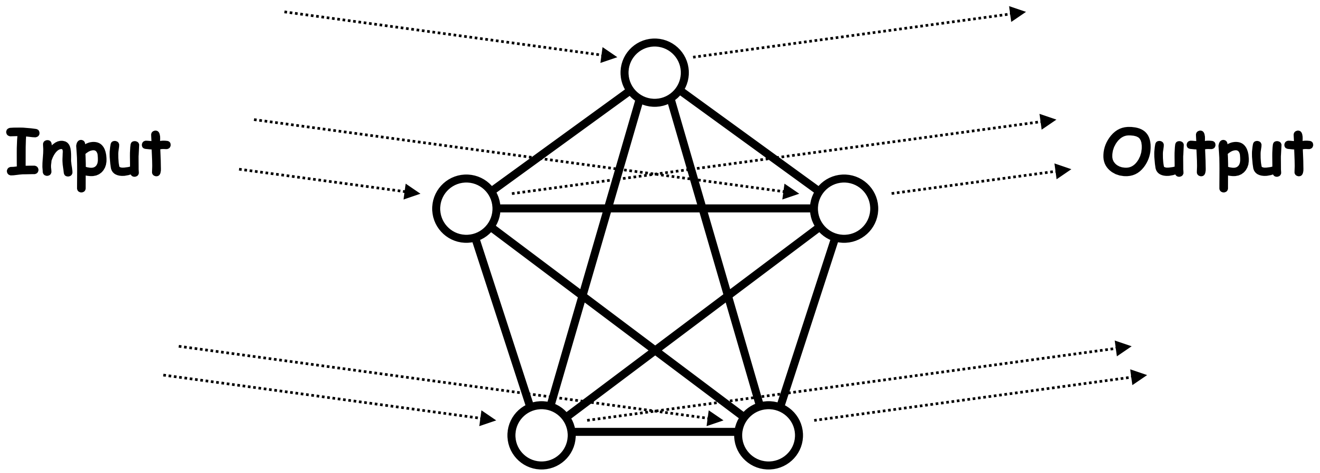


Exercise

- Simulate a cascade of failure on a scale-free network made of 100 nodes with random node capacities and load assignments

- Investigate which node has the most significant impact when it fails

Hopfield network



- A.k.a. “**attractor network**”
- Neurons connected in a shape of an undirected weighted complete graph
- Each neuron takes either 1 or -1, and updates its state in discrete time

State-transition rule

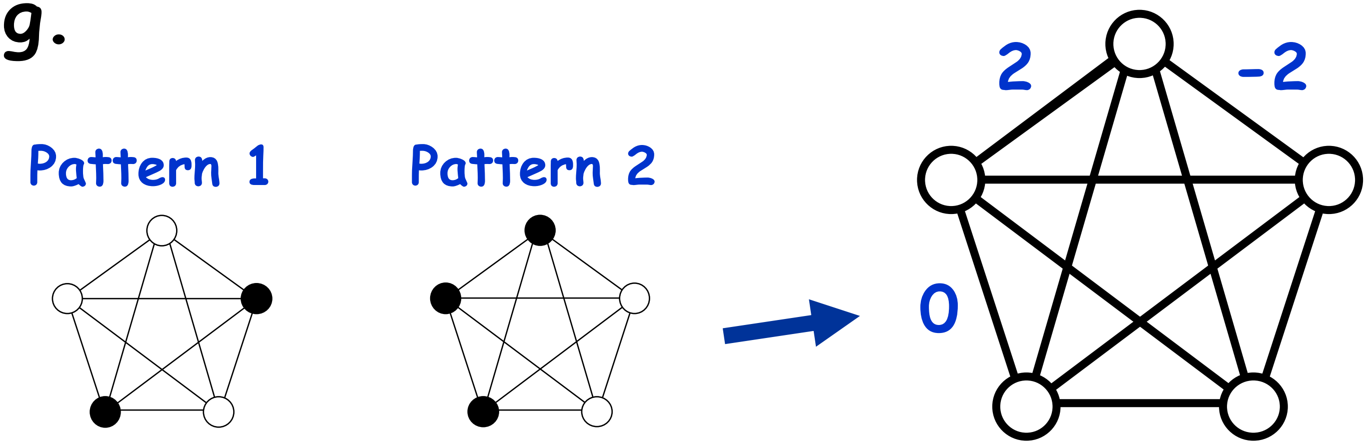
$$s_i(t+1) = \mathrm{sign}\left(\Sigma_j \, w_{ij} \, s_j(t) \right)$$

- w_{ij} : connection weight between neuron i and neuron j
- $w_{ij} = w_{ji}$ (symmetric interaction)
- $w_{ii} = 0$ (no feedback to itself)

Setting weights by "imprinting"

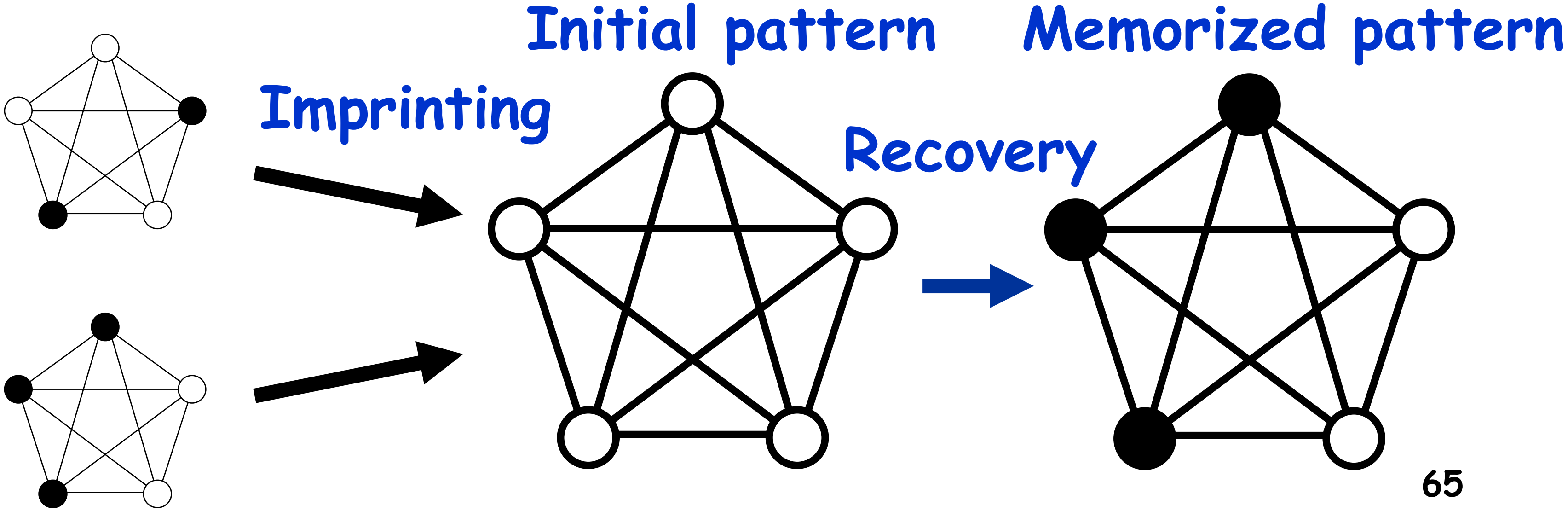
$$w_{ij} = \Sigma_k s^k_i s^k_j$$

- k : index of patterns memorized
- s^k_i : state of neuron i in pattern k
  - e.g.



Recovering patterns

- When started with some initial pattern, the network "remembers" the closest pattern in its memory (or its reversal)
  - Can be applied to content addressable memory, pattern recognition, etc.



Imprinting

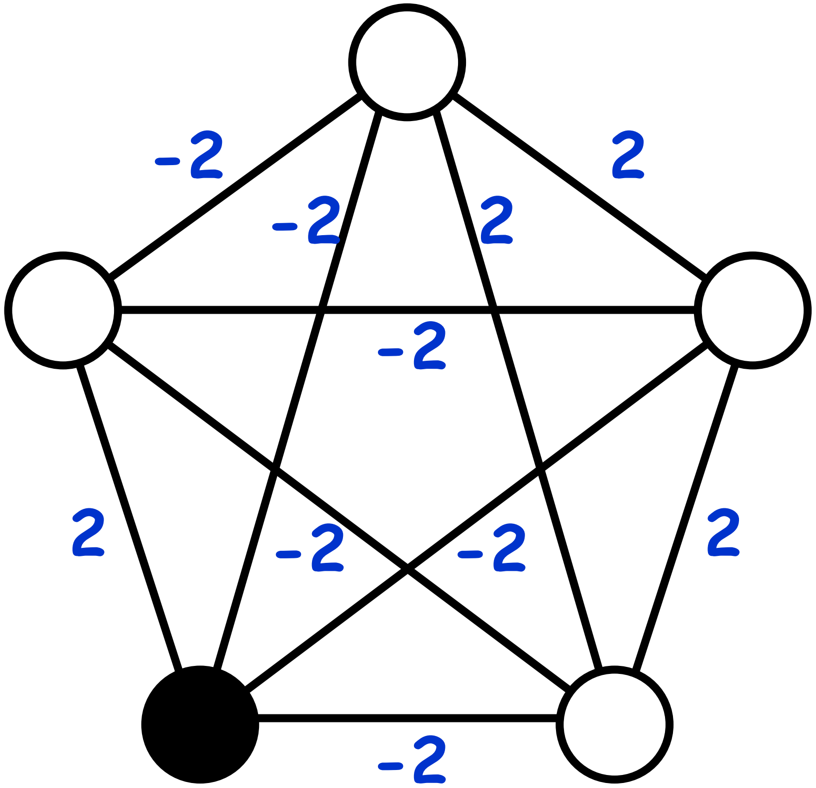
Initial pattern

Recovery

Memorized pattern

Exercise

- Simulate the behavior of the following Hopfield network



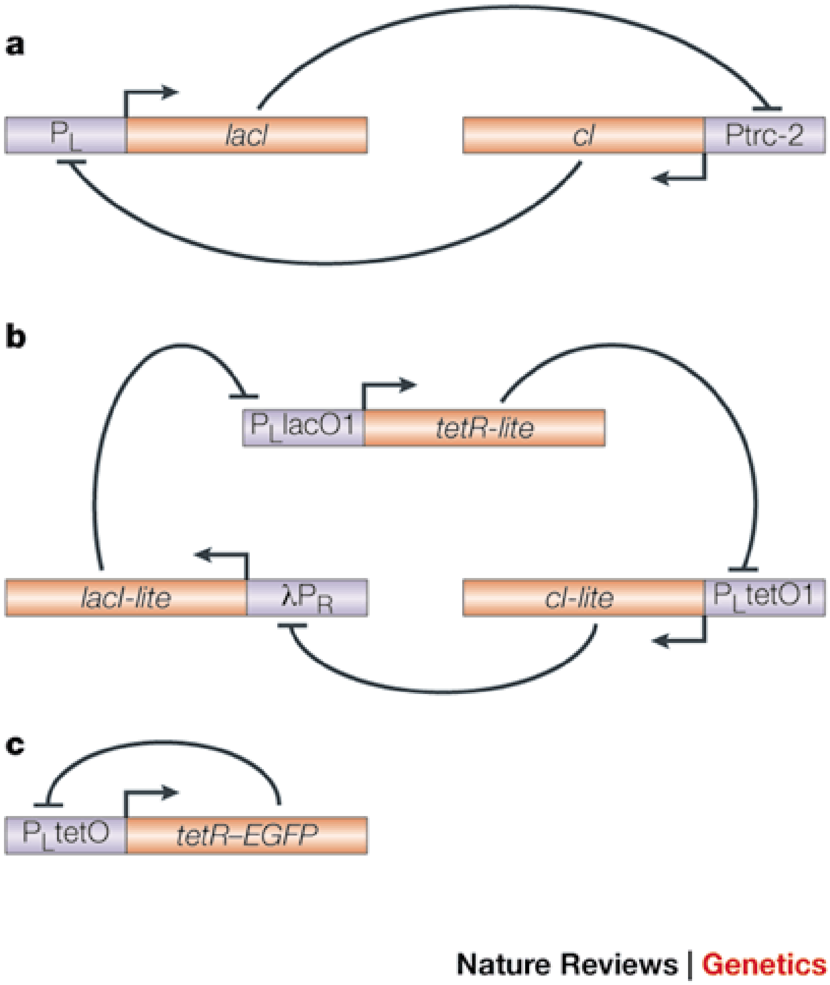
○ : +1

● : -1

2, -2 : weights

Gene regulatory network

- Each gene is activated or inhibited by other genes
  - Forming a network of "logic gates"
  - Each gene takes binary state (on/off)



(from Hasty et al., Nature Reviews Genetics 2, 268-279, 2001)

Boolean network

- Mathematical abstraction of gene regulatory networks
  - Binary node states
  - Each node determines next state using its own Boolean state transition function (referring to neighbors' states)
- Random Boolean network:
  - Network topology and state transition functions are both randomly generated

Example of transition functions

- 2-input functions ($2^{2^2}=16$ possibilities)

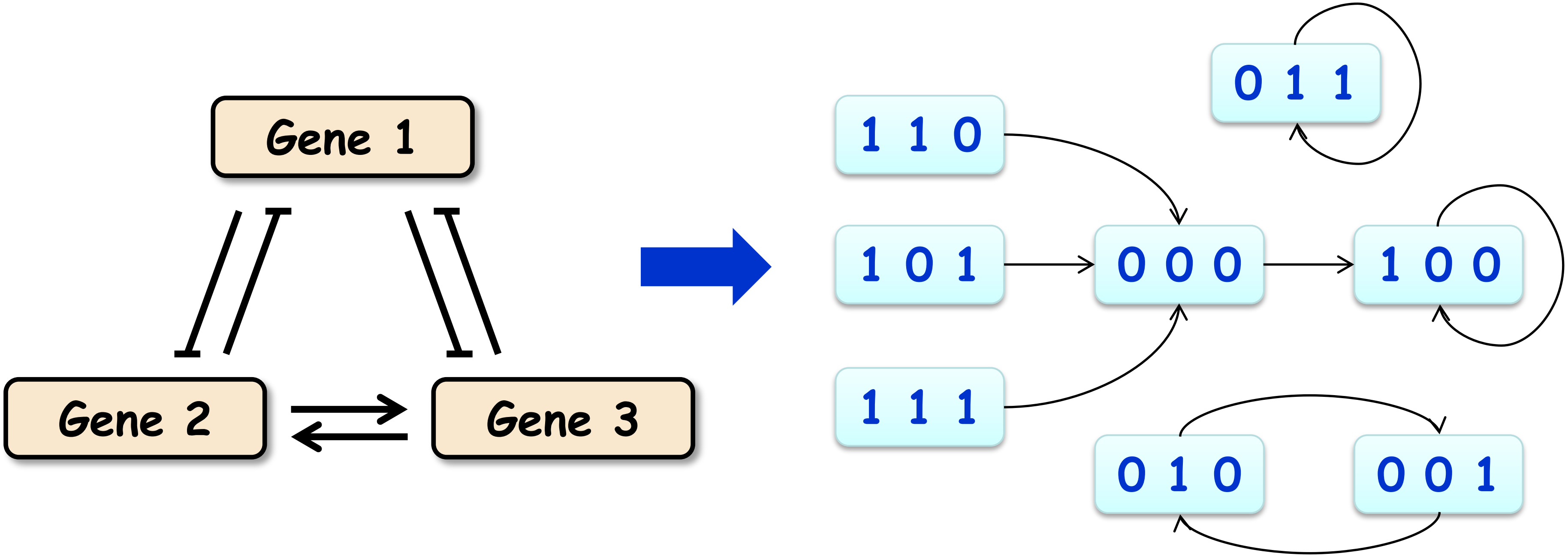
										AND						OR	
X	Y	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Kauffman's NK networks

- N: # of nodes
- K: # of inputs to each node
  - Topologies and state-transition functions are both random
  - Similar to, but not the same as, the NK fitness landscape (NK model) often used in mathematical biology and management sciences

NK network's attractors

- Total # of macro-states: 2^N
- The network eventually falls into one of its "attractors"



Exercise

- Create a Python code that generates the NK network's state-transition diagram (i.e., a directed network whose nodes are the network's macro-states)
- Count how many attractors exist
- Study how # of attractors change when you vary N and K

Dynamics on Networks with Continuous Node States

Simple diffusion

- Individually:

$$\frac{ds_i}{dt} = D \Sigma_{j \text{ in } N_i} (s_j - s_i)$$

- Collectively (with Laplacian L):

$$\frac{ds}{dt} = - D L s$$

Exercise

- Simulate a diffusion process of continuous node states on a Barabasi-Albert scale-free networks with n = 100 and m = 1

Exercise

- Calculate the eigenvalues and eigenvectors of Laplacian matrices of several different network topologies
- Interpret their meanings in the context of diffusion
- Confirm your interpretation by numerical simulation of the diffusion processes

Synchronization

- Linear coupling model:

$$\frac{ds_i}{dt} = F(s_i) + \Sigma_j \left(c_{ij} H(s_j) \right)$$

  - F(s): internal dynamics
  - $C = (c_{ij})$: coupling matrix
  - H(s): output function

– If $s_i(t) = s(t)$ for all i, then the network is synchronized

Synchronization and Laplacian

- If coupling depends only on the difference of outputs across a link:

$$\frac{ds_i}{dt} = F(s_i) + \sigma \Sigma_{j \text{ in } N_i} (H(s_j) - H(s_i))$$

  - I.e., $C = - \sigma L$
  - Laplacian's "spectral gap" (first non-zero eigenvalue) is critical in determining synchronizability of the network

Exercise

- Simulate the following nonlinear Kuramoto model:

$$\frac{ds_i}{dt} = w_i + K/|N_i| \Sigma_{j \text{ in } N_i} \sin(s_j - s_i)$$

  - w_i: inherent angular velocity
  - N_i: neighbors of node I

– What kind of networks synchronize most easily?

Exercise

- Measure and plot the following "phase coherence" in the simulation of the Kuramoto model:

$$r = | \Sigma_j e^{i\theta_j} / n |$$

Synchronizability

Synchronizability

- Synchronizability of a simple coupled dynamical network can be studied by conducting stability analysis

$$\frac{dx_i}{dt} = R(x_i) + \alpha \sum_{j \in N_i} (H(x_j) - H(x_i))$$

$R(x)$: Local reaction term (homogeneous)
$H(x)$: Output function

Exercise

- Consider adding a small perturbation to the general solution of the dynamical equation (w/o interactions)

$$\frac{dx}{dt} = R(x) \quad \rightarrow \quad x_s(t)$$

- Conduct stability analysis by assuming:

$$x_i(t) = x_s(t) + \Delta x_i(t)$$

Condition for synchronizability

- Solution $x_s(t)$ is stable (i.e., the network is synchronized) if

$$\alpha \lambda_i H'(x_s(t)) > R'(x_s(t))$$

for all i and t

(you need to consider only λ_2 and λ_n)

Exercise

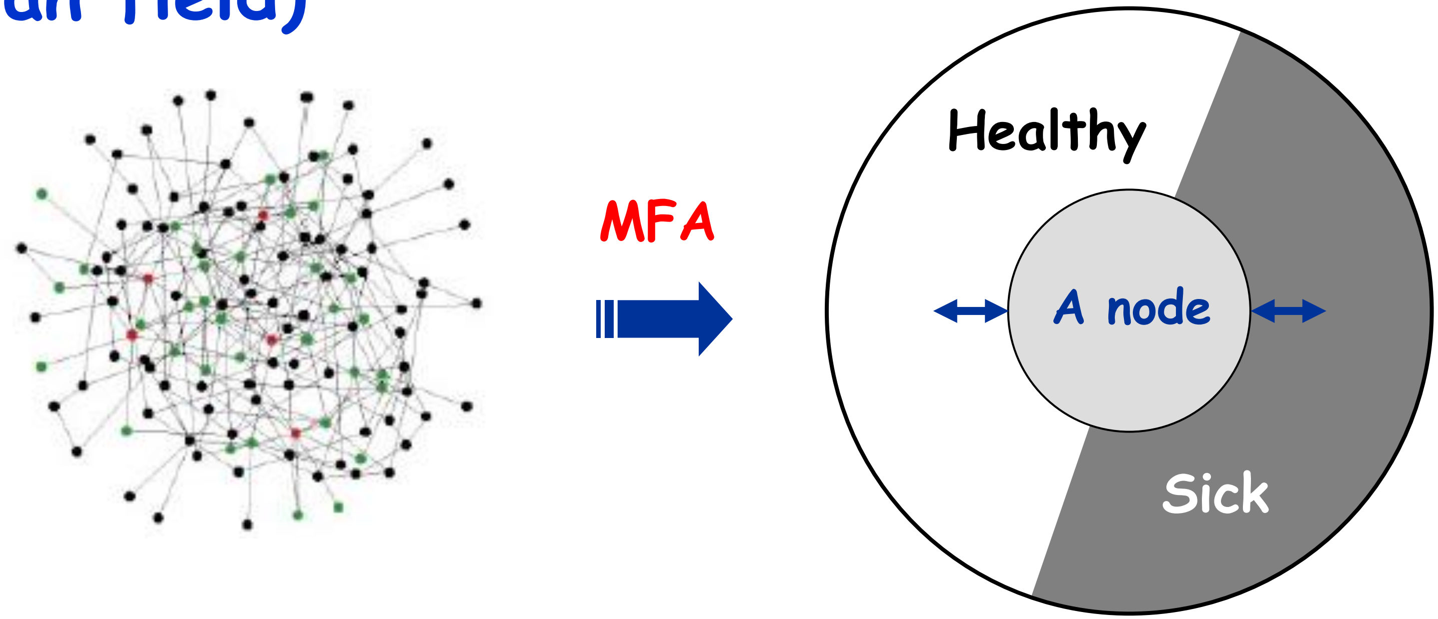
- **Analyze the synchronizability condition of the following coupled oscillator model:**

$$\frac{d\theta_i}{dt} = \beta\theta_i + \alpha \sum_{j \in N_i} (\theta_j - \theta_i)$$

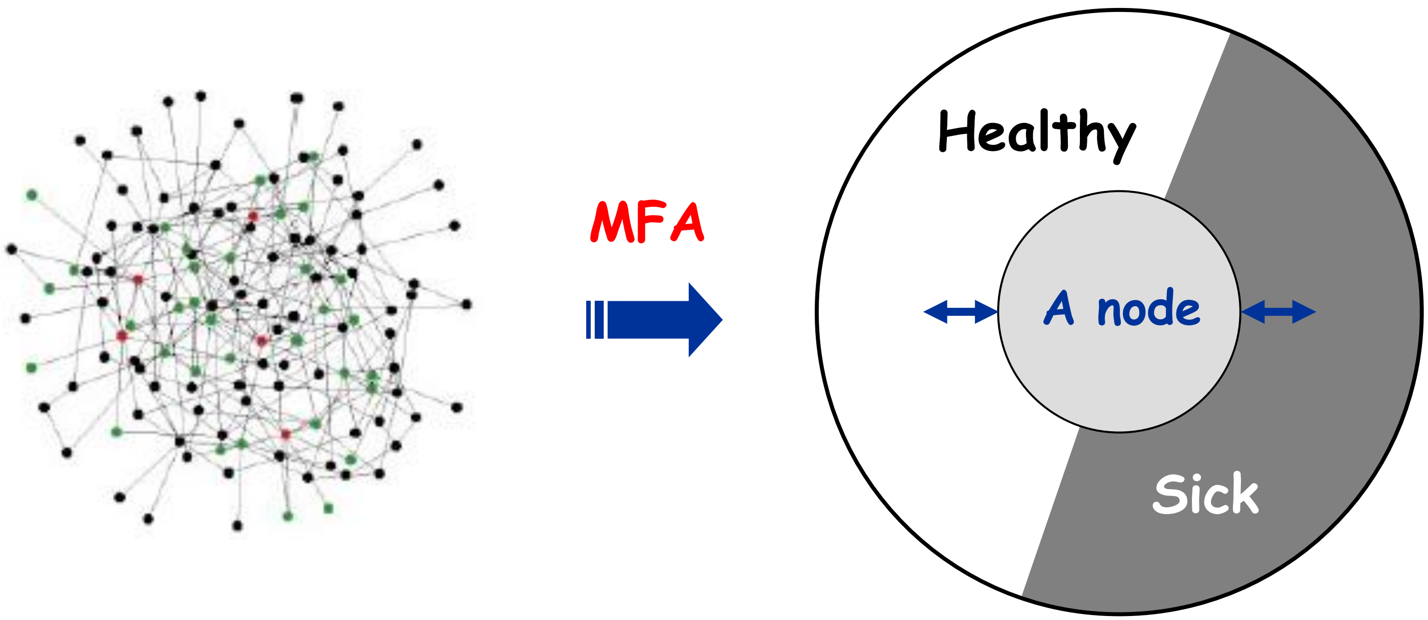
Mean-Field Approximation

Mean-field approximation

- An approximation to drastically reduce the dimensions of the system by reformulating the dynamics in terms of "a state of one node" and "the average of all the rest (= mean field)"



How MFA works



1. Make an approximated description about how one node changes its state through the interaction with the average of all the rest (= mean field)
2. Assume that 1. uniformly applies to all the nodes, and analyze how the mean field itself behaves

Mathematical description of MFA (difference equations)

- Original equations:

$$x^i_t = F^i(\{ x^i_{t-1} \})$$

- Approximate equations with MFA:

$$x^i_t = F'^i(x^i_{t-1}, \langle x \rangle_{t-1})$$

$$\langle x \rangle_t = \Sigma_i \, x^i_{t-1} / n$$

Each state-transition function
takes only two arguments:
its own state and the "mean field"

Example: SIS on a random network

- Infection probability p_i
- Recovery probability p_r
- Edge probability p_e

- Write down a difference equation that describes how the probability of infected nodes, q_t (mean field), changes over time

Example: SIS on a random network

Current state	Next state	Probability of this transition
0 (susceptible)	0 (susceptible)	$(1-q)(1-p_e q p_i)^{n-1}$
0 (susceptible)	1 (infected)	$(1-q)\left(1-(1-p_e q p_i)^{n-1}\right)$
1 (infected)	0 (susceptible)	$q p_r$
1 (infected)	1 (infected)	$q(1-p_r)$

- Find equilibrium states
- Study the stability of those equilibrium points
  - When does the equilibrium $q = 0$ become unstable (i.e., epidemic occurs)?

Example: SIS on a SF network

- Infection probability p_i
- Recovery probability p_r
- Degree distribution $P(k)$

- Write down a difference equation that describes how the probability of infected nodes with degree k, $q_t(k)$ (many mean fields), changes over time

Degree-dependent infection

- Probability for a node with degree k to get infected from its neighbor:

$$\sum_{k'} P_n(k'|k)q(k')p_i$$

Pn : neighbor degree probability distribution

If the network is nonassortative:

$$P_n(k') = \frac{k'}{\langle k \rangle} P(k')$$

FYI: Friendship paradox

- "Your friends have more friends than you do, on average"

$$\sum_{k'} k' P_n(k') = \sum_{k'} \frac{k'^2 P(k')}{\langle k \rangle} = \frac{\langle k^2 \rangle}{\langle k \rangle}$$

$$= \frac{\langle k \rangle^2 + \sigma(k)^2}{\langle k \rangle} = \langle k \rangle + \frac{\sigma(k)^2}{\langle k \rangle}$$

Calculation...

Current state	Next state	Probability of this transition
0 (susceptible)	0 (susceptible)	$(1-q(k))\left(1-\sum_{k'}\frac{k'}{\langle k\rangle}P(k')q(k')p_i\right)^k$
0 (susceptible)	1 (infected)	$(1-q(k))\left(1-\left(1-\sum_{k'}\frac{k'}{\langle k\rangle}P(k')q(k')p_i\right)^k\right)$
1 (infected)	0 (susceptible)	$q(k)p_r$
1 (infected)	1 (infected)	$q(k)(1-p_r)$

$$\begin{aligned} q_{t+1}(k) &= (1-q(k))\left(1-\left(1-\sum_{k'}\frac{k'}{\langle k\rangle}P(k')q(k')p_i\right)^k\right)+q(k)(1-p_r) \\ &= (1-q(k))\left(1-(1-q_n p_i)^k\right)+q(k)(1-p_r), \end{aligned}$$

$$q_n = \frac{\sum_{k'} k'P(k')q(k')}{\langle k\rangle}$$

Calculation…

$$\begin{aligned} q_{t+1}(k) &= (1-q(k))\,(1-(1-kq_np_i)) + q(k)(1-p_r) \\ &= (1-q(k))kq_np_i + q(k) - q(k)p_r = f(q(k)) \end{aligned}$$

$$q_{\mathrm{eq}}(k) = \frac{kq_np_i}{kq_np_i + p_r}$$

With this:

$$q_n = \frac{1}{\langle k \rangle} \sum_{k'} k' P(k') \frac{k' q_n p_i}{k' q_n p_i + p_r}$$

Calculation…

$$q_n = \frac{1}{\langle k \rangle} \sum_{k'} k' P(k') \frac{k' q_n p_i}{k' q_n p_i + p_r}$$

- For BA SF networks, this becomes:

$$q_n = \frac{1}{2m} \sum_{k'=m}^{\infty} k' \cdot 2m^2 k'^{-3} \frac{k' q_n p_i}{k' q_n p_i + p_r},$$

$$1 = m p_i \sum_{k'=m}^{\infty} \frac{1}{k'(k' q_n p_i + p_r)}.$$

$$q_n \approx \frac{p_r}{(e^{\frac{p_r}{m p_i}} - 1) m p_i}$$

Calculation…

- **Final stability analysis:**

$$\left.\frac{df(q(k))}{dq(k)}\right|_{q(k)=\frac{kq_np_i}{kq_np_i+p_r}} = -kq_np_i + \frac{p_r}{kq_np_i+p_r}\frac{k^2P(k)p_i}{\langle k\rangle} + 1 - p_r = r(k)$$

$$\begin{aligned} r(k) &= -k\frac{p_r}{(e^{\frac{p_r}{mp_i}}-1)mp_i}p_i + \frac{p_r}{k\frac{p_r}{(e^{\frac{p_r}{mp_i}}-1)mp_i}p_i+p_r}\frac{k^2\cdot 2m^2k^{-3}p_i}{2m} + 1 - p_r \\ &= -\frac{kp_r}{(e^{\frac{p_r}{mp_i}}-1)m} + \frac{mp_i}{\frac{k^2}{(e^{\frac{p_r}{mp_i}}-1)m}+k} + 1 - p_r \end{aligned}$$

Conclusion

- **If $p_i \to 0$:**

$$\lim_{p_i \to 0} r(k) = -\frac{k p_r}{\left(\left[e^{\frac{p_r}{m p_i}} \to \infty\right] - 1\right) m} + \frac{m\,[p_i \to 0]}{\frac{k^2}{\left(\left[e^{\frac{p_r}{m p_i}} \to \infty\right] - 1\right) m} + k} + 1 - p_r$$

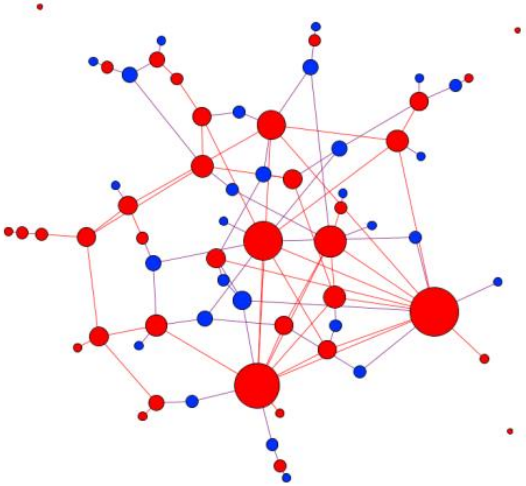
$$= 1 - p_r$$

- **Since $0 < 1 - p_r < 1$, the non-zero equilibrium state (i.e., epidemic) is still stable even if $p_i \to 0$ on scale-free networks!!**

Take-home lesson

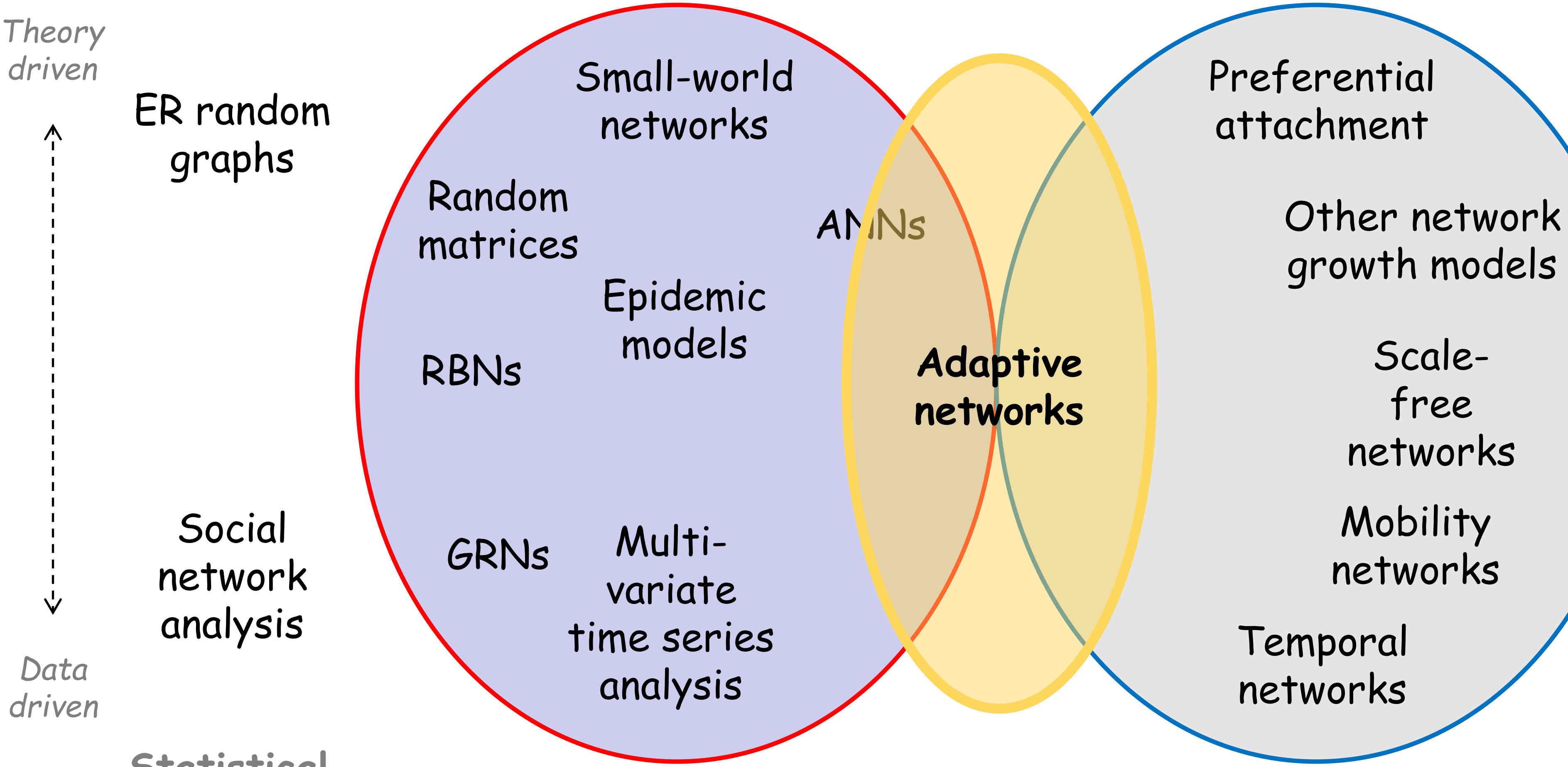
- Dynamics on networks can be influenced significantly by network topology

Simulation III: Adaptive Networks



Hiroki Sayama
sayama@binghamton.edu

A map of network science



Dynamics *on* networks

Dynamics *of* networks

Theory driven

Data driven

ER random graphs

Social network analysis

Small-world networks

Random matrices

Epidemic models

RBNs

GRNs

Multi-variate time series analysis

ANNs

Adaptive networks

Preferential attachment

Other network growth models

Scale-free networks

Mobility networks

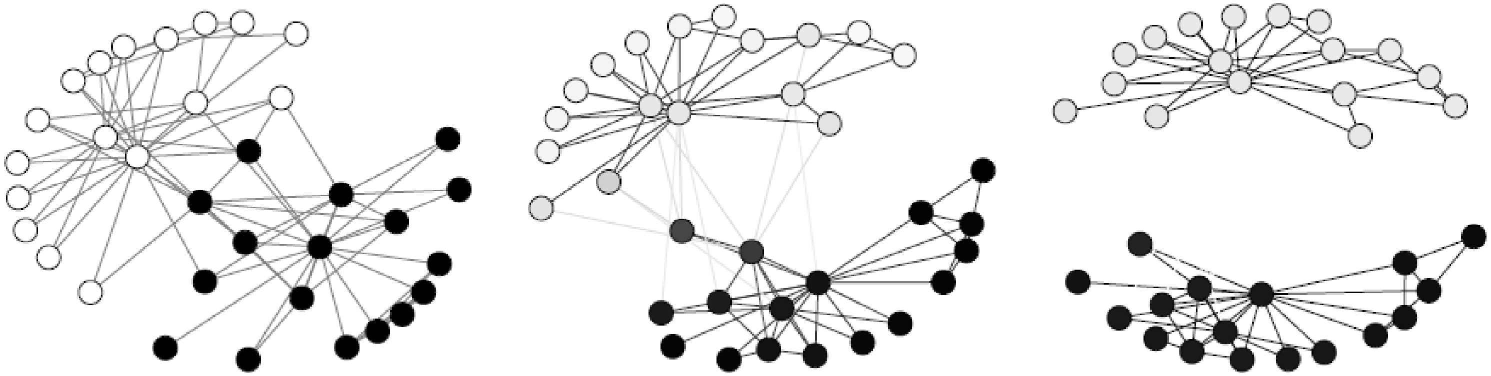
Temporal networks

Statistical approaches

Dynamical approaches

Adaptive networks

- Complex networks whose states and topologies co-evolve, often over similar time scales
  - Link (node) states adaptively change according to node (link) states



Adaptive networks in action

- **Many real-world complex systems show coupling between “dynamics *of* networks” and “dynamics *on* networks”**

System	Nodes	Edges	States of nodes	Topological changes
Organism	Cells	Intercellular communication channels	Gene/protein activities	Fission and death of cells during development
Ecological community	Species	Interspecific relationships	Population	Speciation, invasion, extinction of species
Human society	Individual	Conversations, social relation-ships	Social, professional, economical, political, cultural statuses	Changes in social relationships, entry and withdrawal of individuals
Communica-tion network	Terminals, hubs	Cables, wireless connections	Information stored and transacted	Addition and removal of terminal or hub nodes

Simulation of Adaptive Networks

Simulating state-topology coevolution

- Technically, very easy; not so much different from other network simulation models

- One minor problem:
  How to handle topological changes while state changes are also ongoing?
  → Asynchronous updating

Example: Epidemics on adaptive networks

- Original epidemic network model

  + adaptive changes of links

- A susceptible node that has a link to an infected node will cut the link and reconnect it to another susceptible node with probability p_c
- Does the disease stay in the network?

Exercise

- Study the effects of rewiring probability on the disease fixation on and the global network structure of an initially random social network
  - In what condition will the disease remain within society?
  - How will the topology of the network be reformed through the disease propagation process?

Example: Adaptive voter model

- Original voter model

  + adaptive disconnection of links

- A link that connects two nodes with different opinion states may be cut with probability p_c

- How will the social network and opinions evolve?

Exercise

- Study the effects of the link disconnection probability on the consensus formation in the adaptive voter model
    - Plot the final number of opinions as a function of p_c
    - How will the topology of the network be changed by the diversity of opinions?

Example: Adaptive diffusion model

- Original diffusion model

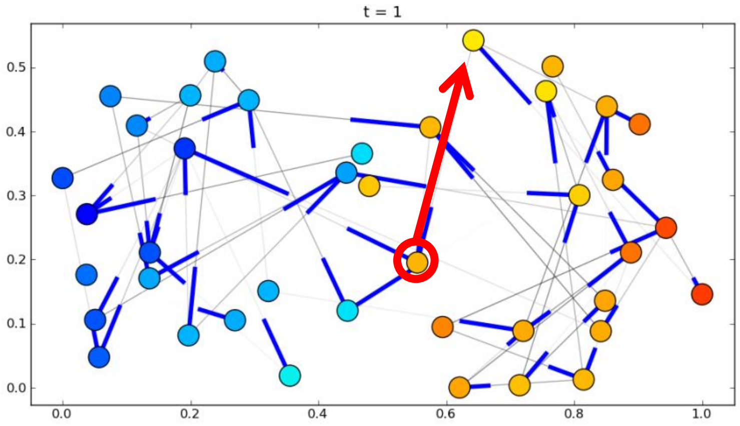
  + adaptive disconnection of links

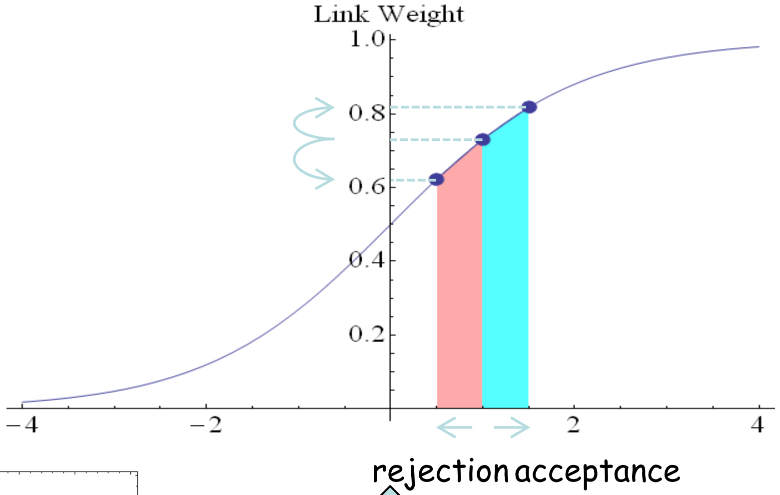
- Link weights will increase or decrease based on the similarity/dissimilarity of node states across the links

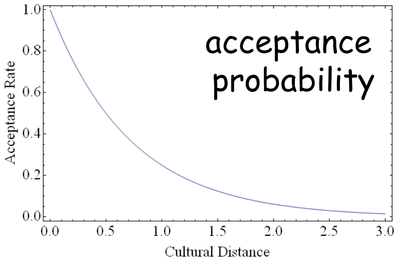
  - Conceptually similar to the adaptive voter model

Application 1: Corporate merger

- **Modeling and simulation of cultural integration in two merging firms**

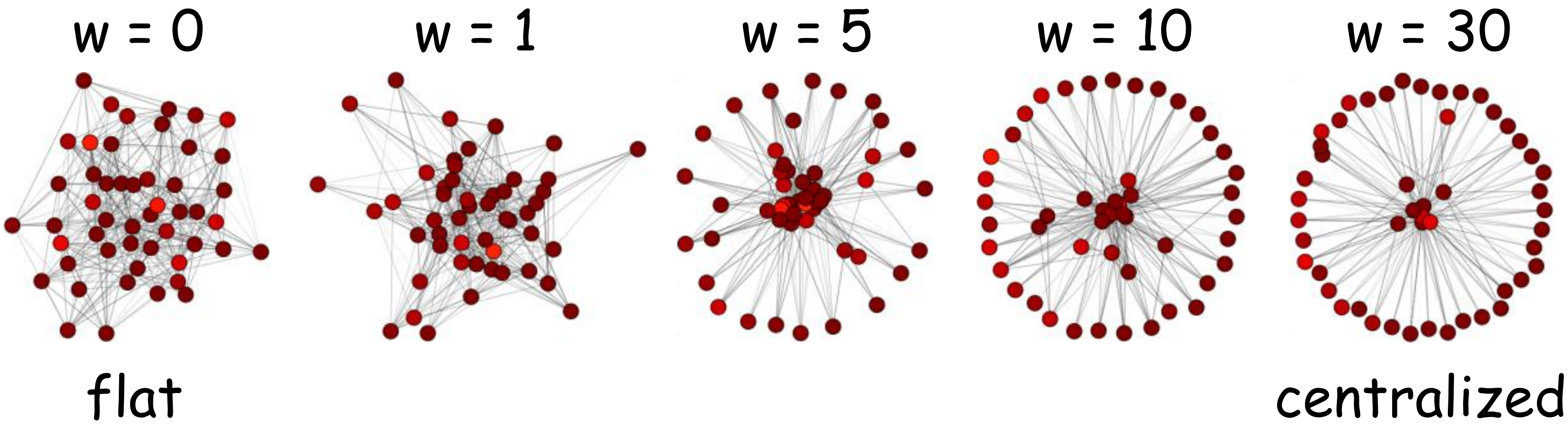






Yamanoi & Sayama, *Comput. Math. Org. Theory* 19, 516-537, 2013.

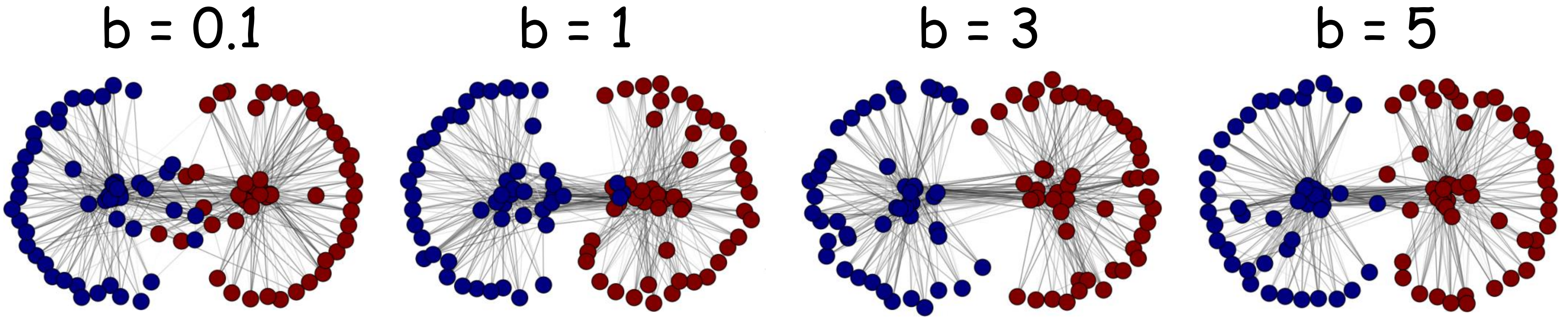
“Within-firm” concentration (w)



- **Prob. for node i to become an info source:**

$P_w(i) \sim (i/n)^w$ (i = 1, 2, …, n; n = firm size)

“Between-firm” concentration (b)



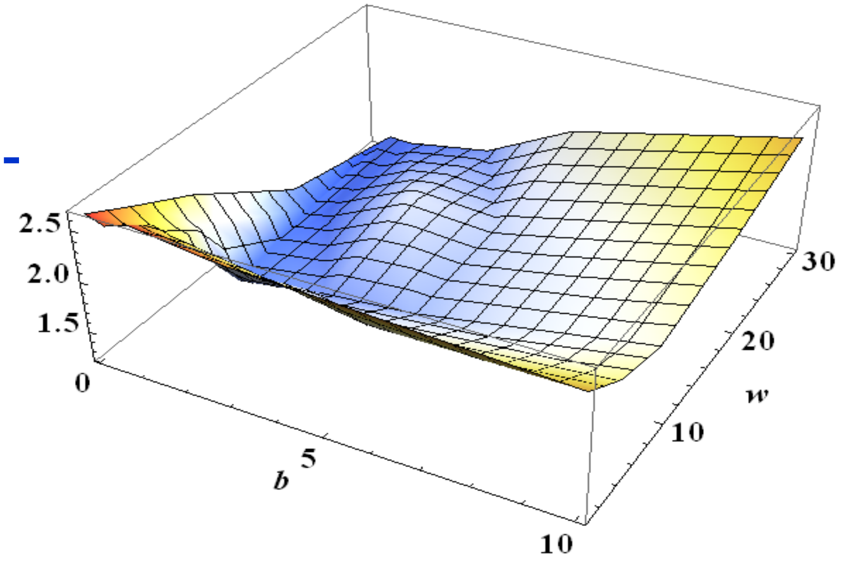
b = 0.1 b = 1 b = 3 b = 5

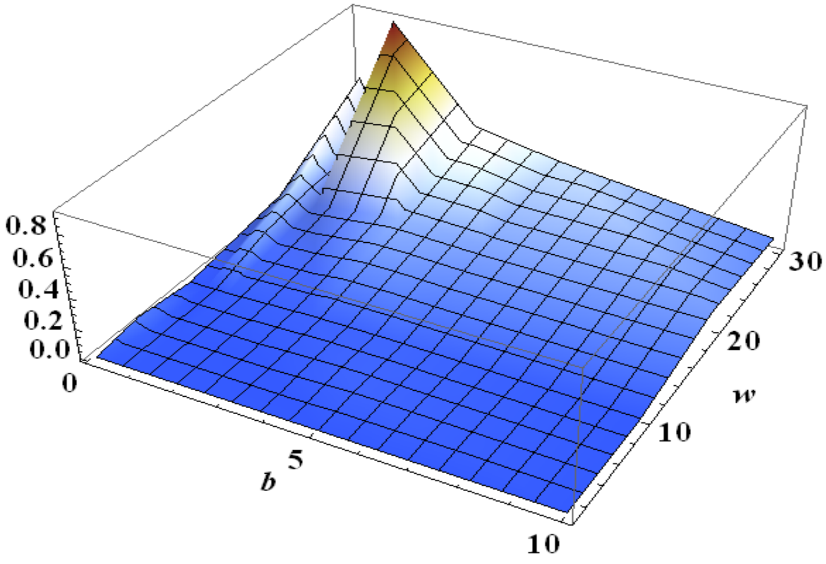
nearly random executive-level

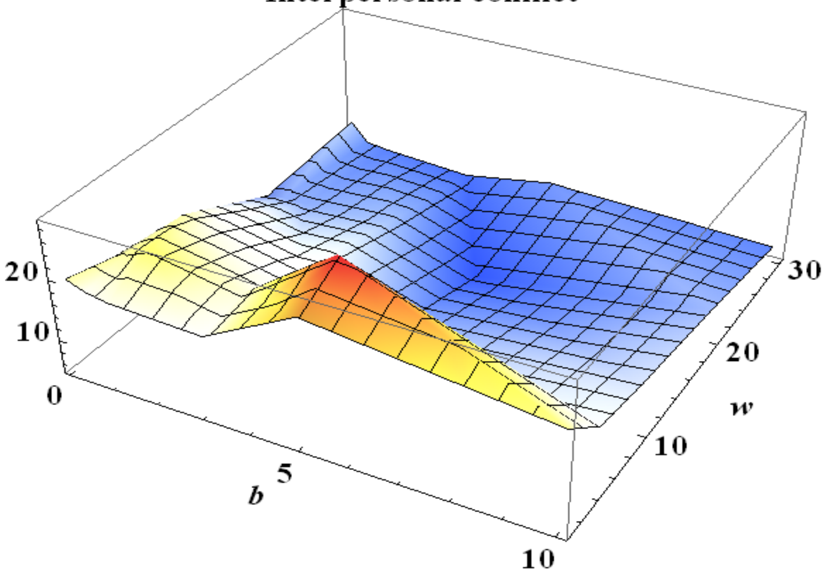
- **Prob. for node i to have an inter-firm tie:**

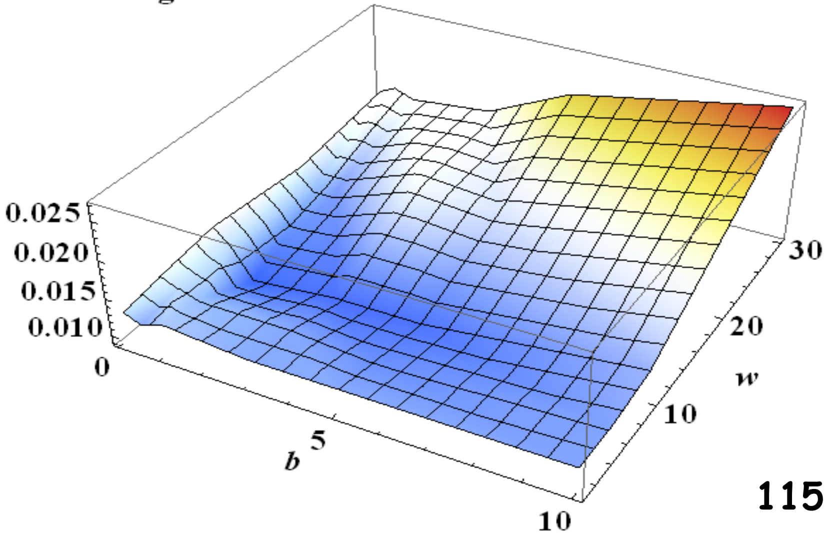
$$P_b(i) \sim c_i^{\,b}$$

(c_i = within-firm closeness centrality of i)


Cultural distance


Turnover


Interpersonal conflict


Organizational communication ineffectiveness

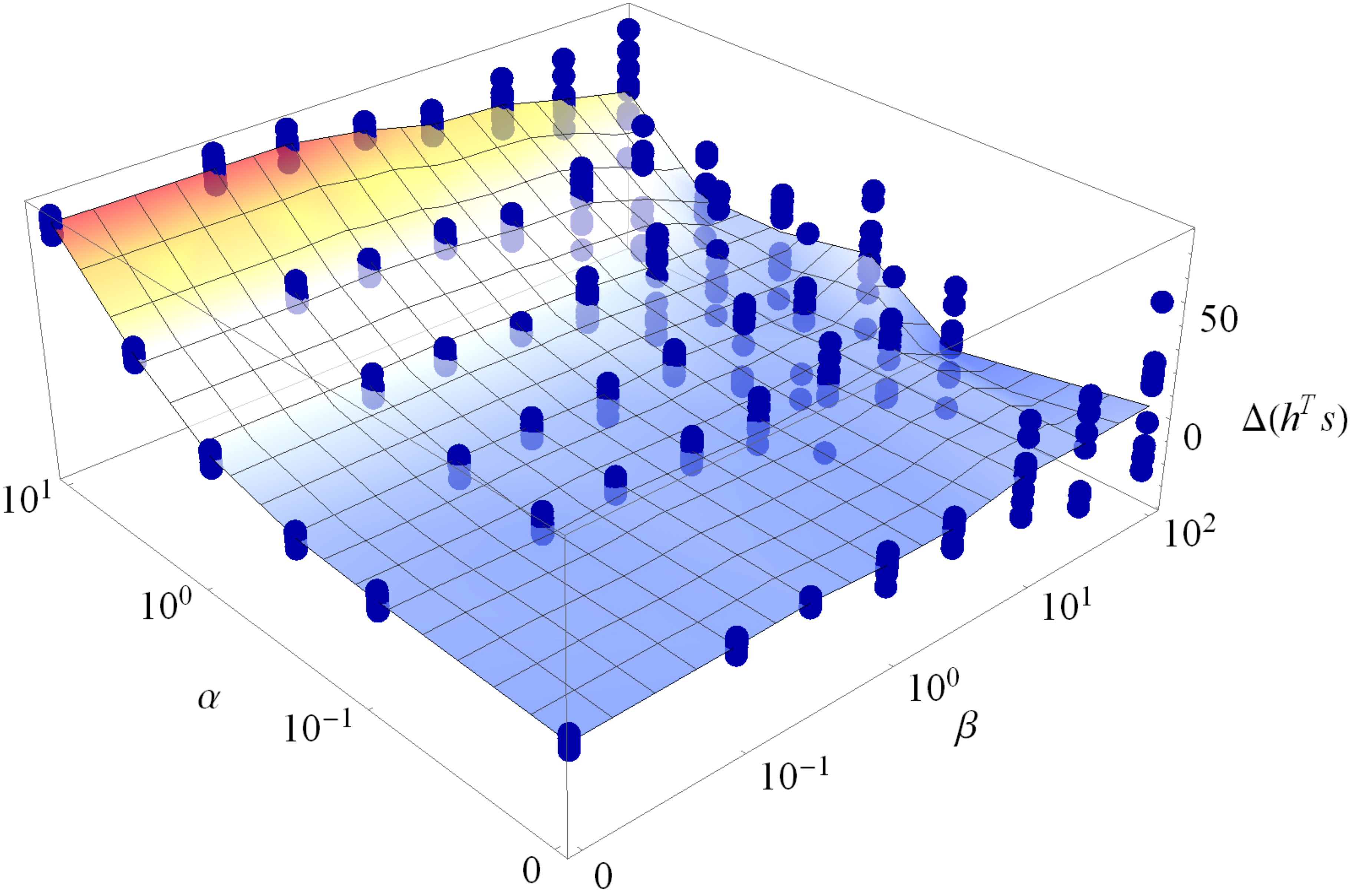
Application 2: Social diffusion and global drift

- Sayama & Sinatra, PRE 91, 032809, 2015

$$\frac{ds_i}{dt} = c\left(\langle s_j \rangle_j^i - s_i\right)$$

Adaptive link weight adjustment:

$$\frac{da_{ij}}{dt} = a_{ij}\left[\alpha \frac{s_i + s_j - 2\langle s \rangle}{2\sigma_s} - \beta \frac{(k_i - \langle k \rangle)(k_j - \langle k \rangle)}{\sigma_k^2}\right]$$


10^1
10^0
α
10^{-1}
0
0
10^{-1}
10^0
β
10^1
10^2
0
50
$\Delta(h^T s)$

Exercise

- Change the rule of link weight adjustment in the adaptive diffusion model
  - E.g., Sayama & Sinatra (2015)
- Simulate the revised model and see how the network topology and state co-evolve

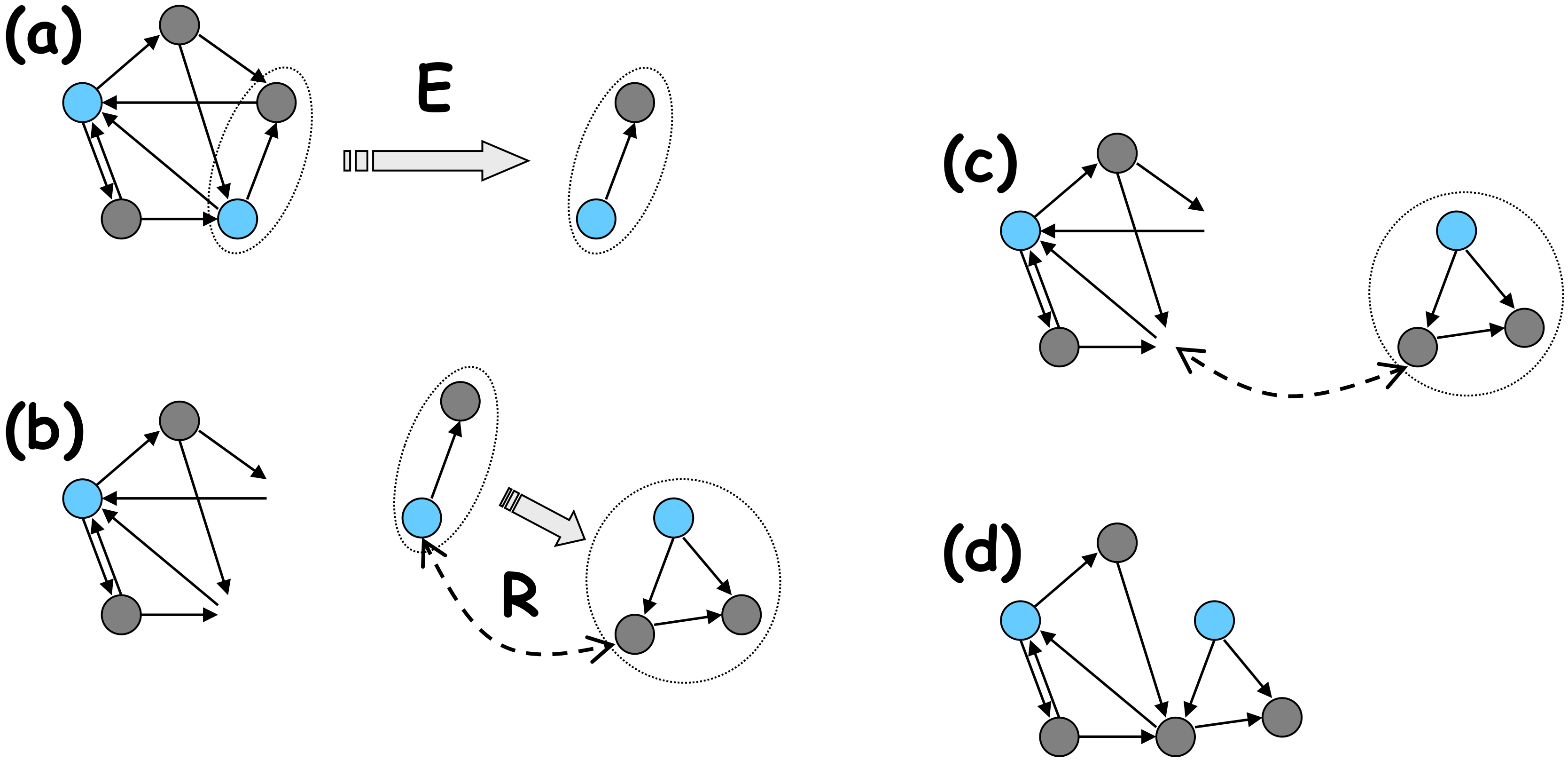
Theoretical Framework:
Generative Network Automata

Generative network automata

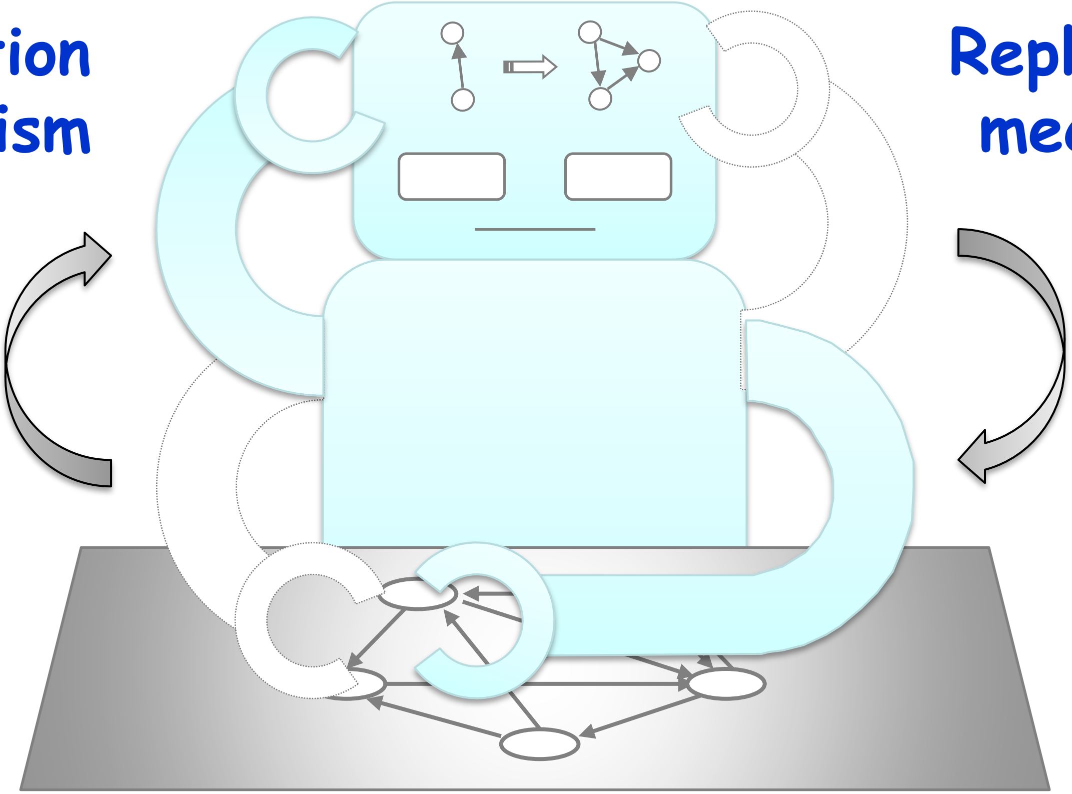
- Unified representation of dynamics *on* and *of* networks using graph rewriting
- Defined by <E, R, I>:
  - E : Extraction mechanism — When, Where
  - R : Replacement mechanism — What
  - I : Initial configuration

Sayama, *Proc. 1st IEEE Symp. Artif. Life*, 2007, pp.214-221.

GNA rewriting example



Actually, it's a generative network automat~~a~~-on



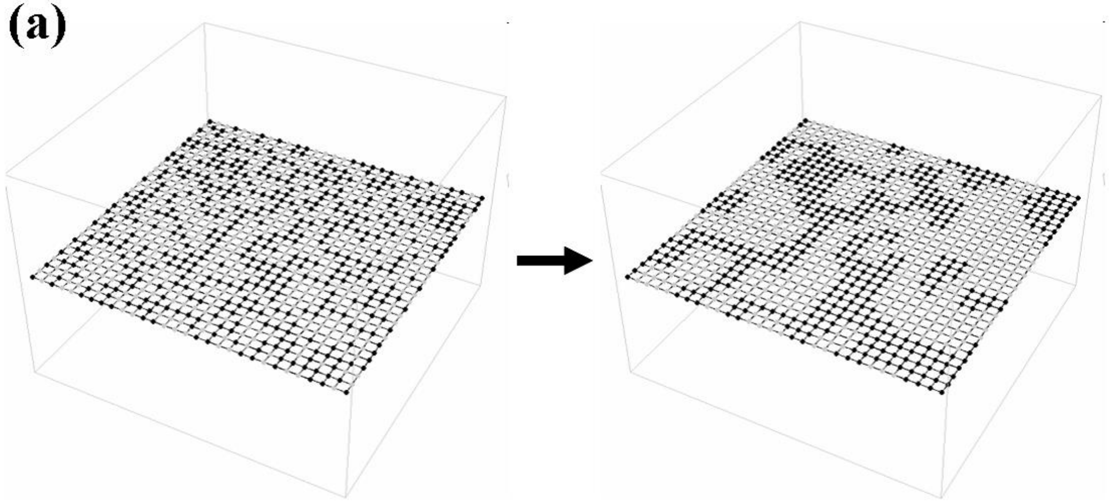
E :
Extraction mechanism

R:
Replacement mechanism

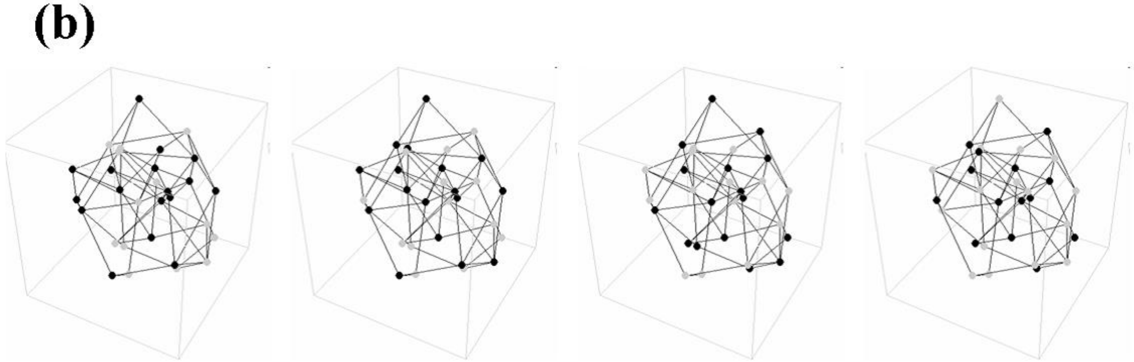
Generality of GNA

- GNA can uniformly represent in <E, R, I>:
  - Conventional dynamical systems models
    - If R always conserves local network topologies and modifies states of nodes only
    - E.g. cellular automata, Boolean networks
  - Complex network generation models
    - If R causes no change in local states of nodes and modifies topologies of networks only
    - E.g. small-world, scale-free networks

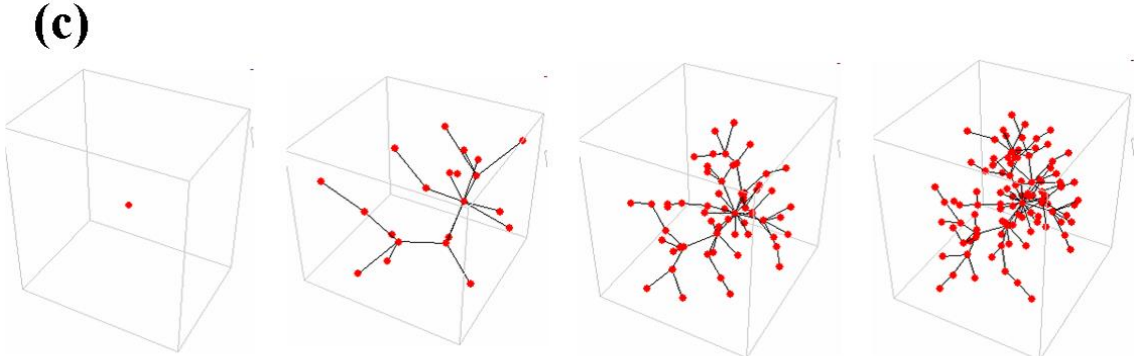
Cellular automata



Random Boolean network



BA scale-free network



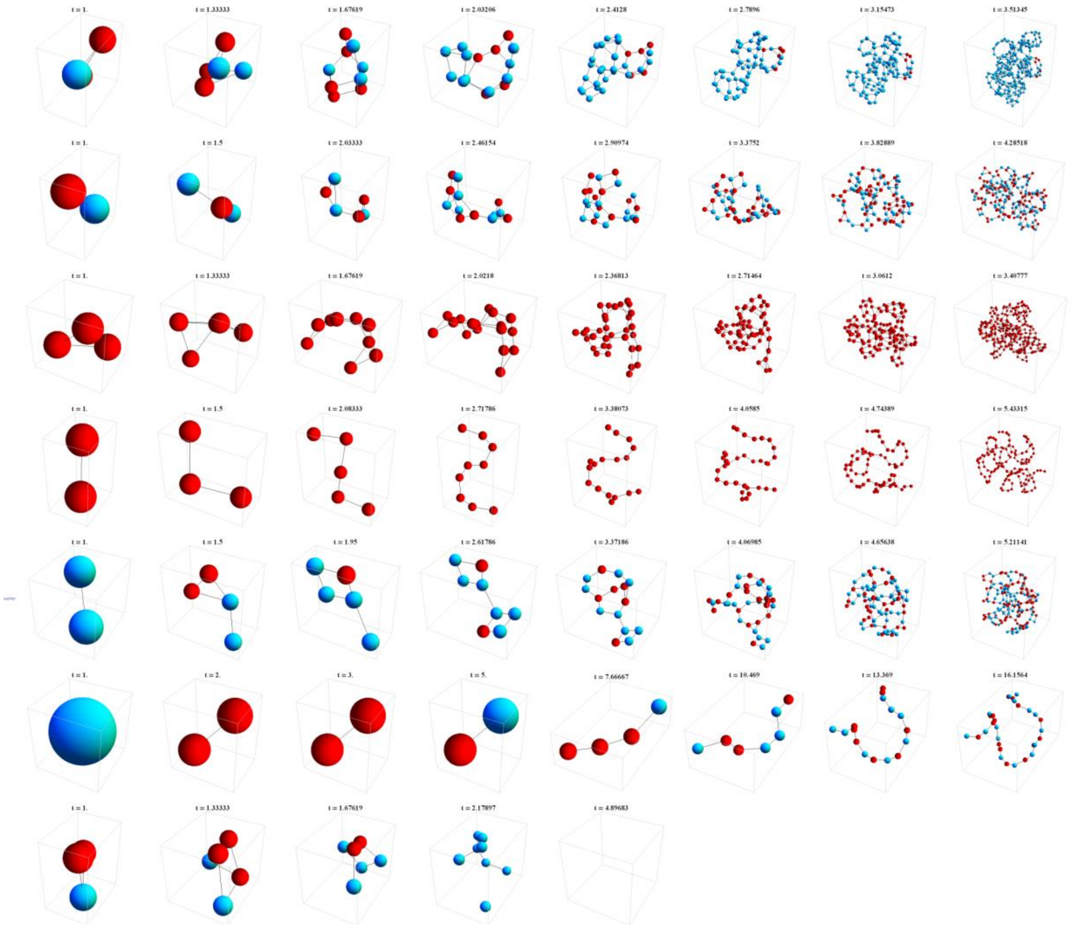
Exhaustive search of rules

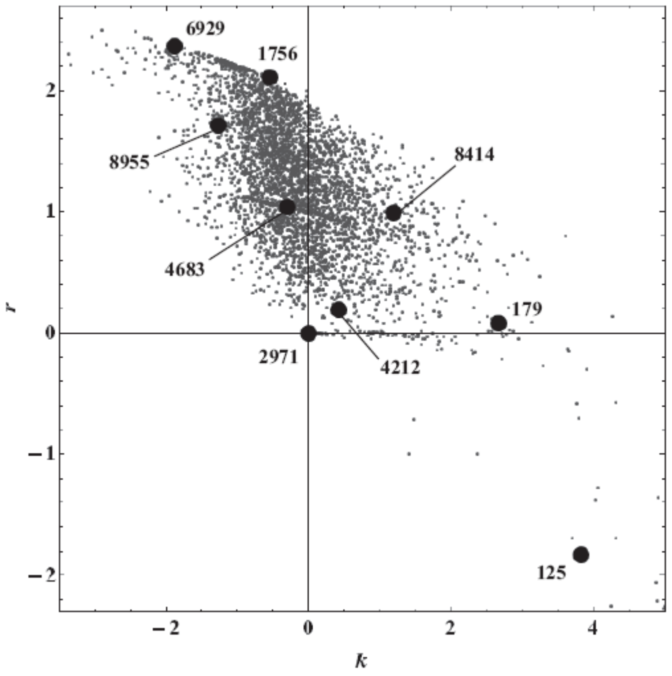
- E samples a node randomly and then extracts an induced subgraph around it
- R takes 2-bit inputs (states of the node and neighbors) and makes 1-out-of-10 decisions
  - Total number of possible R's: $10^{2^2} = 10{,}000$
- **"Rule Number" rn(R)** is defined by

$$rn(R) = a_{11}\ 10^3 + a_{10}\ 10^2 + a_{01}\ 10^1 + a_{00}\ 10^0$$

  - $a_{ij} \in \{0,\ 1,\ \dots\ 9\}$: Choices of R when state of u is i and local majority state is j

Exhaustive search of rules



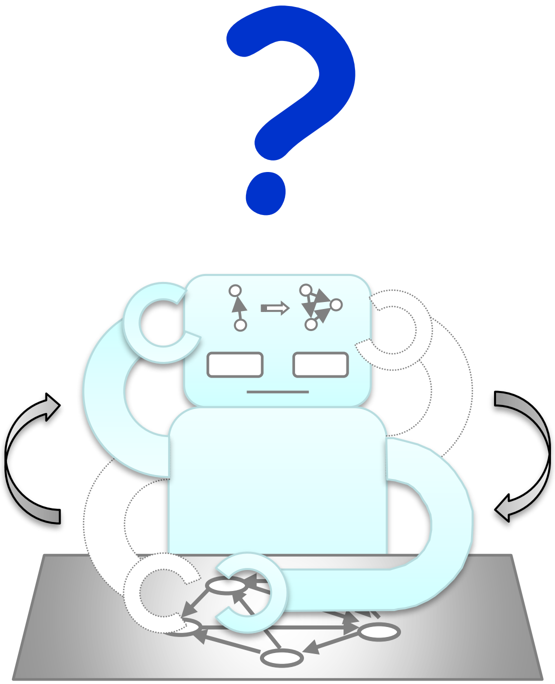


Sayama & Laramee, *Adaptive Networks*, Springer, 2009, pp.311-332.

Developing Adaptive Network Models from Empirical Data

A challenge

- How to derive a set of dynamical rules directly from empirical data of network evolution?
- Separation of extraction and rewriting in GNA helps the rule discovery



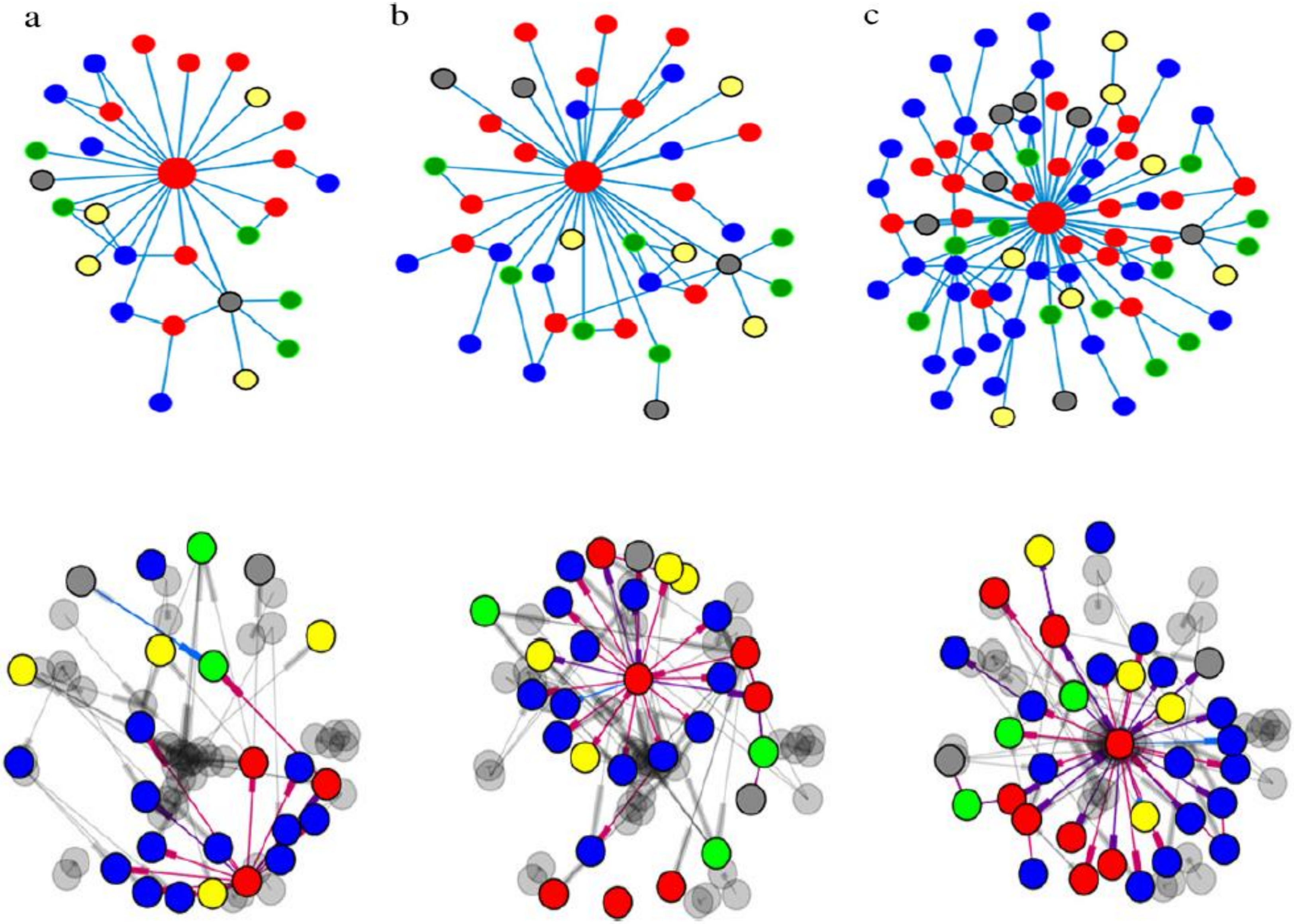
Pestov, Sayama, & Wong, *Proc. 9th Intl. Conf. Model. Simul. Visual. Methods*, 2012.

Schmidt & Sayama, *Proc. 4th IEEE Symp. Artif. Life*, 2013, pp.27-34.

Application to operational network modeling

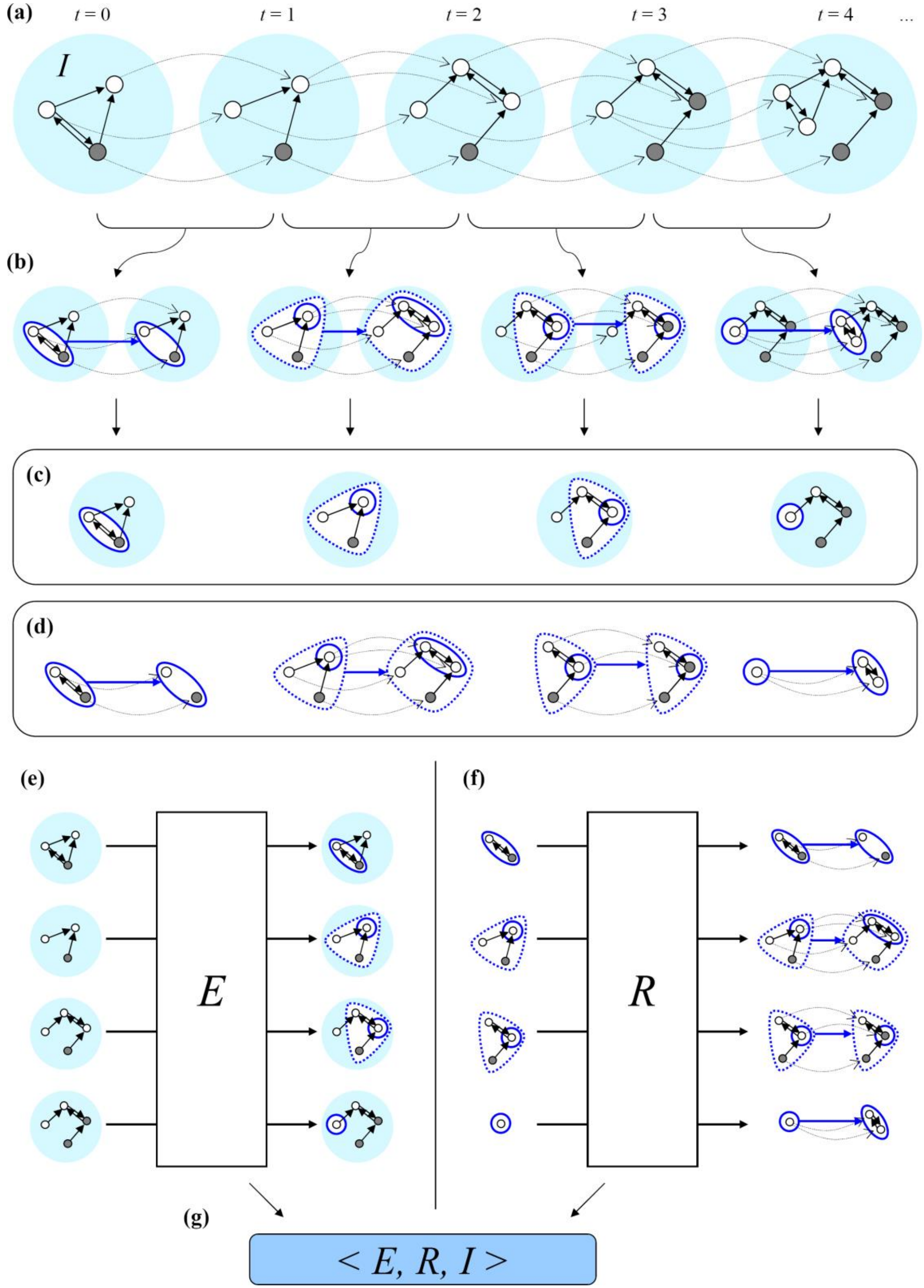
- Canadian Arctic SAR (Search And Rescue) operational network
  - Rewriting rules manually built directly from actual communication log of a December 2008 SAR incident

  - Developed a simulator for hypothetical SAR operational network development

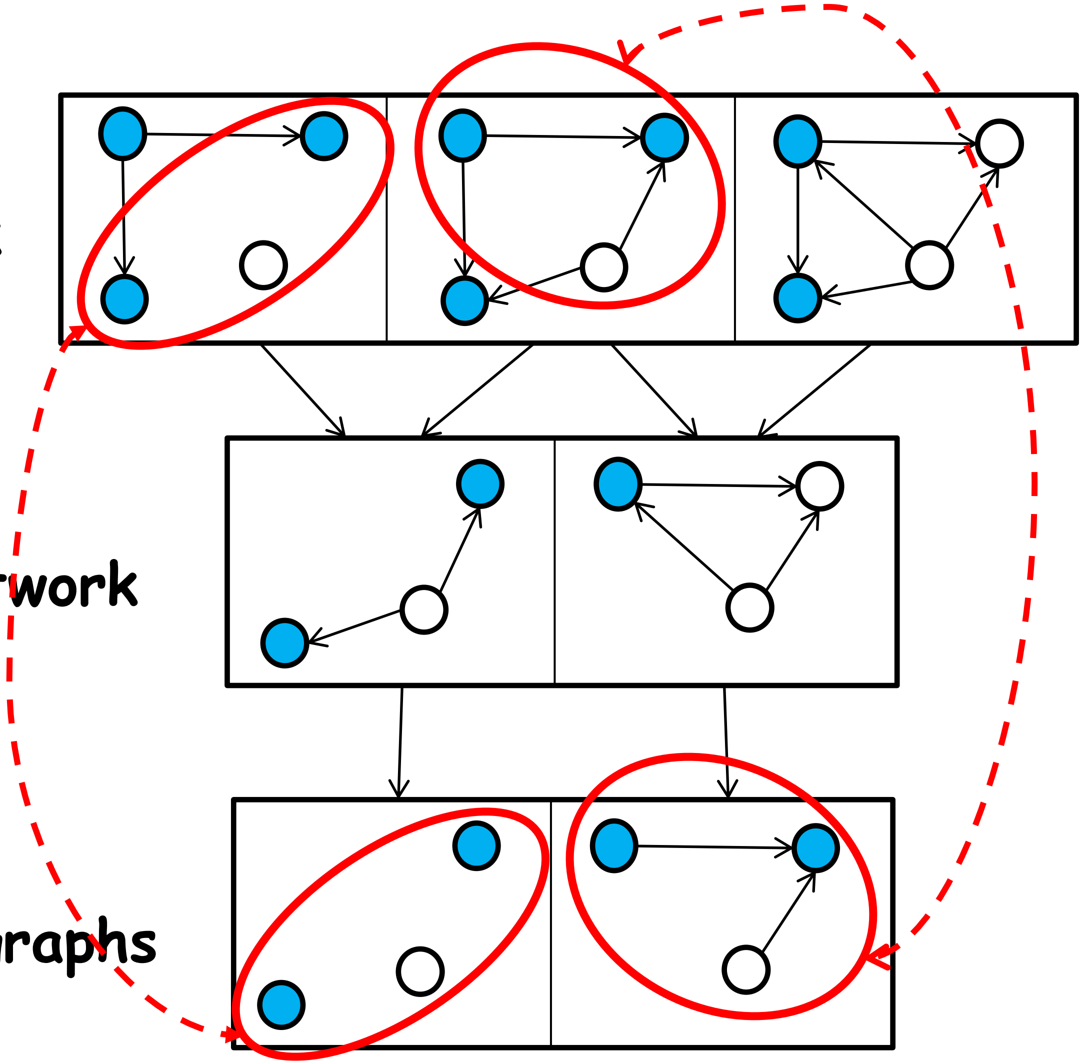

a
b
c

Automation of model discovery from data: PyGNA

- Adaptive network rule discovery and simulation implemented in Python
  - https://github.com/schmidtj/PyGNA
- Input: Time series of network snapshots
- Output: A GNA model that best describes given data


(a)
t = 0
t = 1
t = 2
t = 3
t = 4
...
I
(b)
(c)
(d)
(e)
(f)
E
R
(g)
< E, R, I >

Extracted subgraphs



Input Network

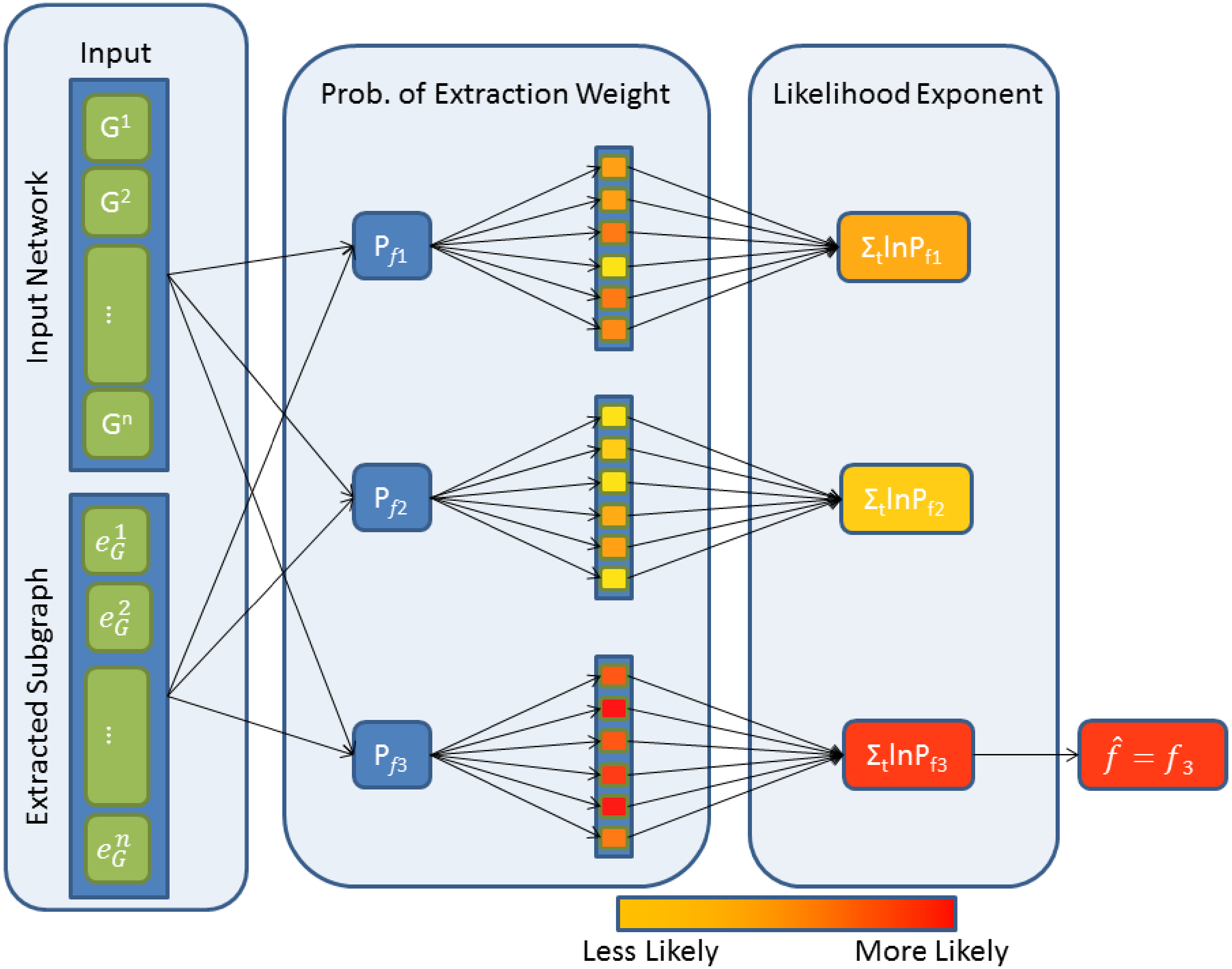
Compressed Network

Extracted Subgraphs

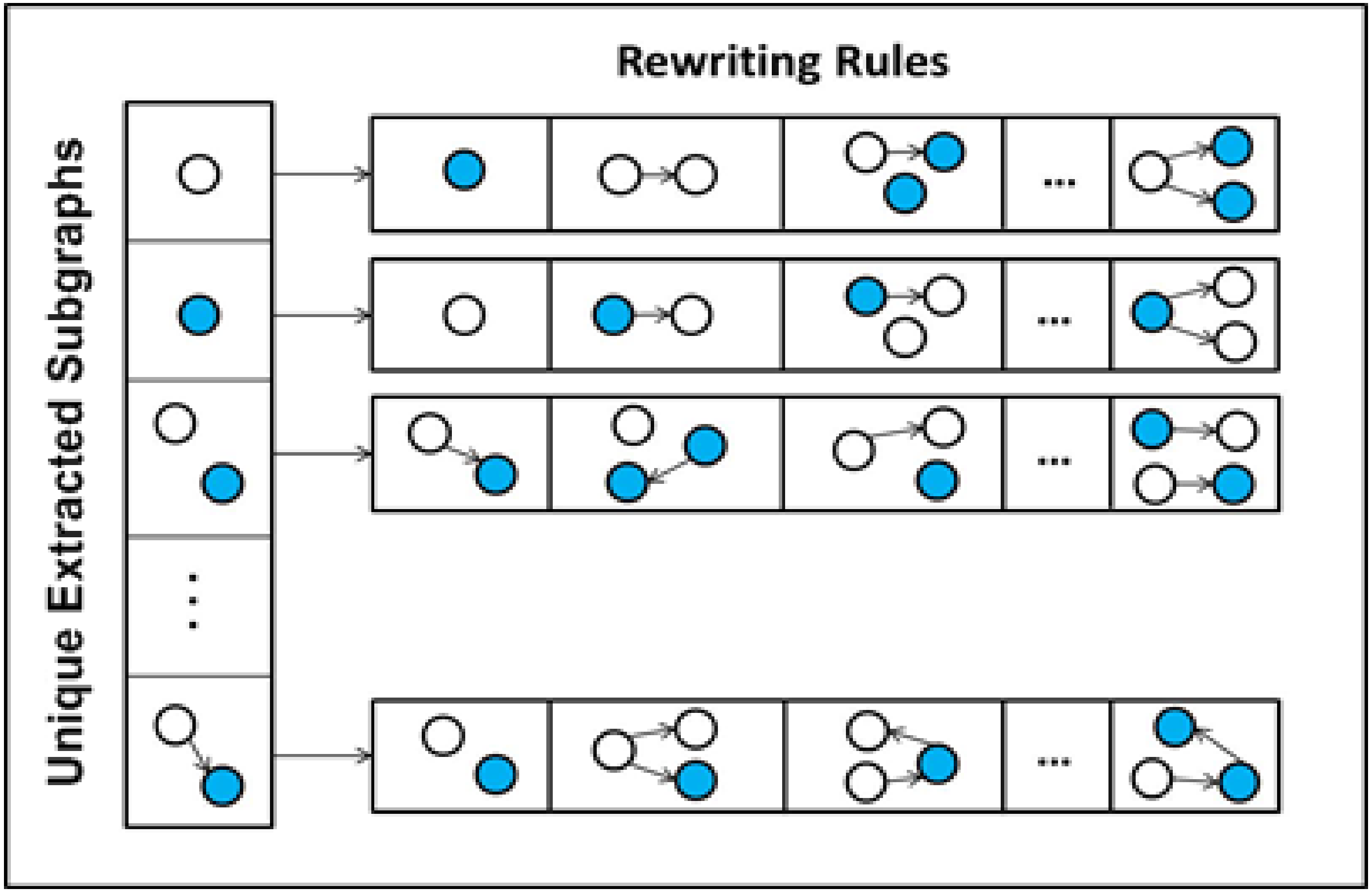
Extraction mechanism identification: “Where, when”

- Candidate models provided by user
  - Degree-based preferential selection
  - State-based preferential selection
  - Degree & State-based etc…
- Maximum likelihood method
  - Computes likelihood using each hypothetical model & accumulates log likelihood over time
  - Chooses the model with maximum likelihood

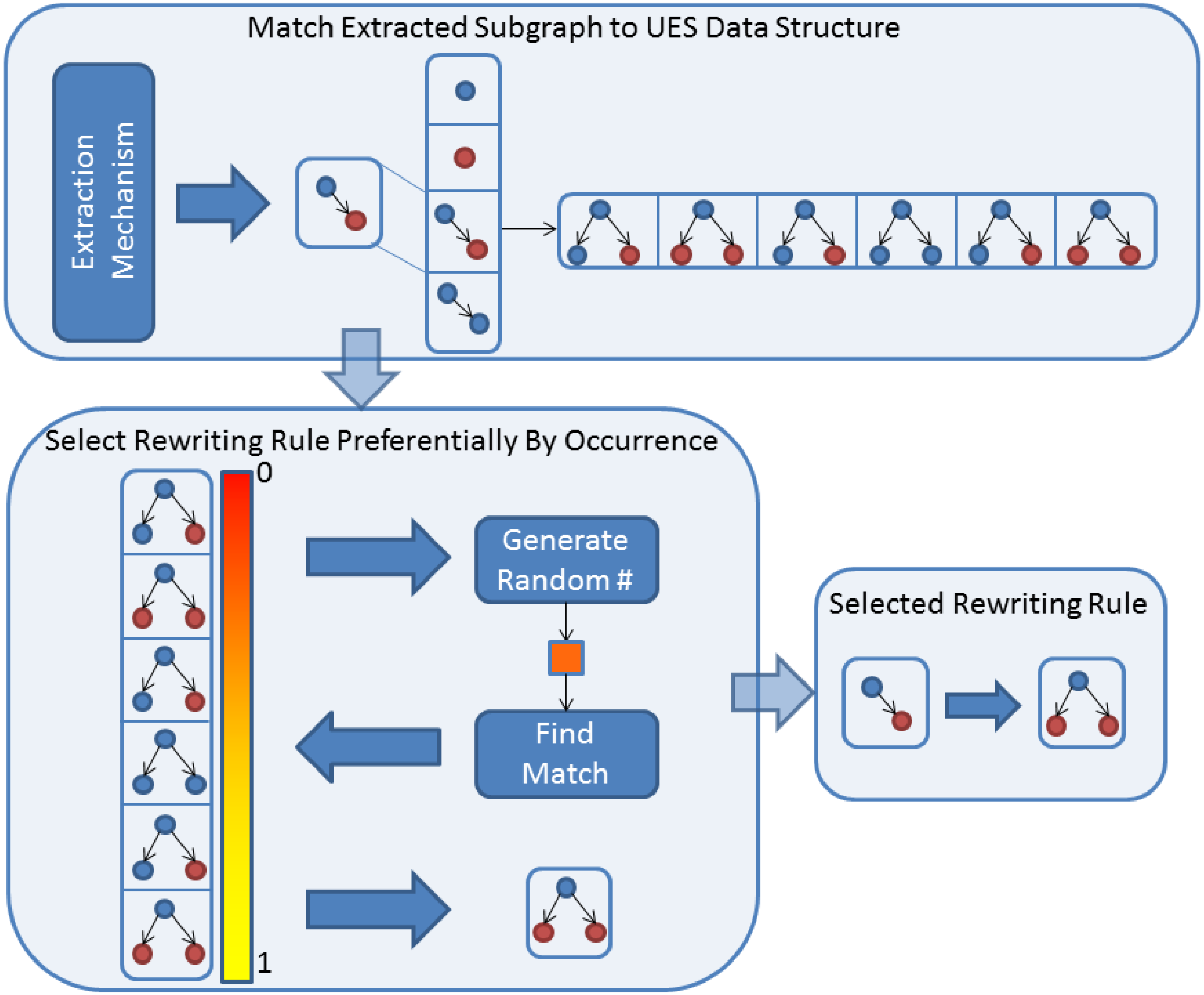
Algorithm


Input
Input Network
G^1
G^2
⋮
G^n
Extracted Subgraph
e_G^1
e_G^2
⋮
e_G^n
Prob. of Extraction Weight
P_{f1}
P_{f2}
P_{f3}
Likelihood Exponent
$\Sigma_t \ln P_{f1}$
$\Sigma_t \ln P_{f2}$
$\Sigma_t \ln P_{f3}$
$\hat{f} = f_3$
Less Likely
More Likely

Replacement mechanism identification: "What"

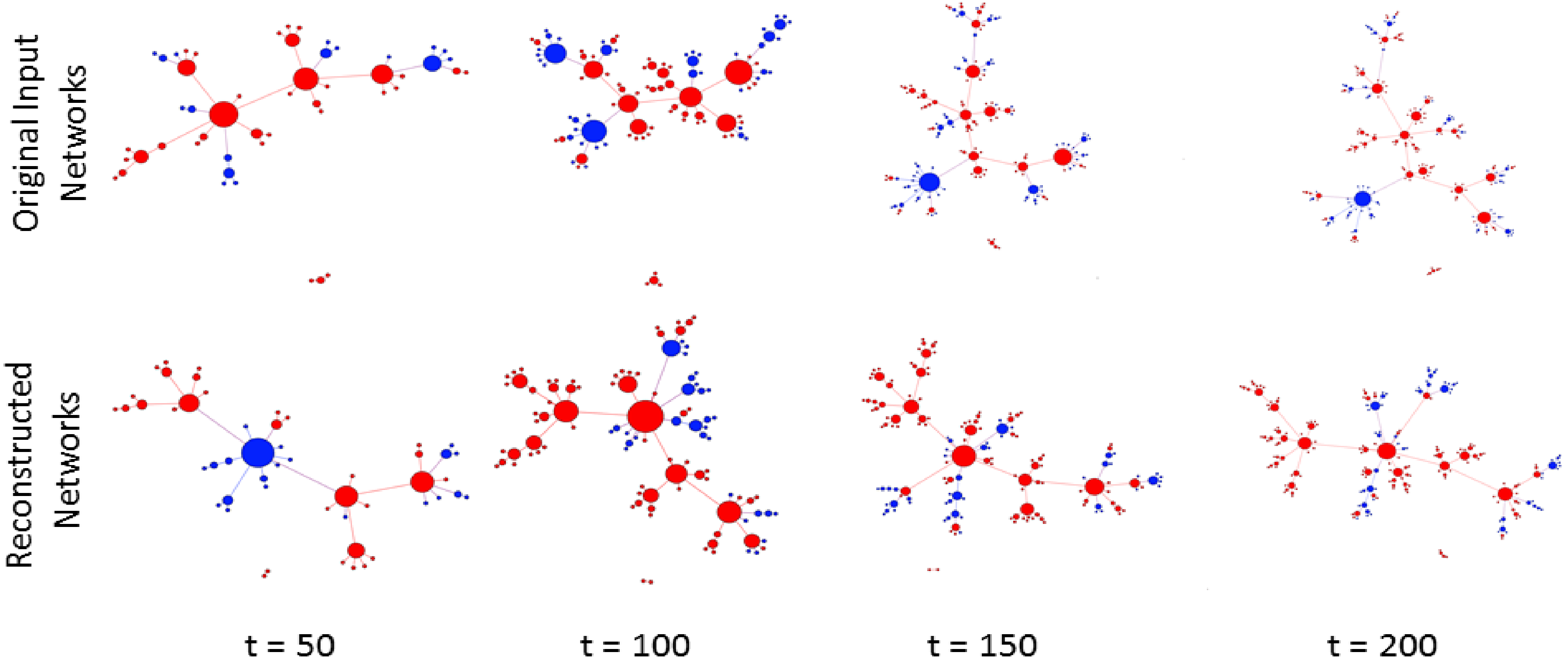


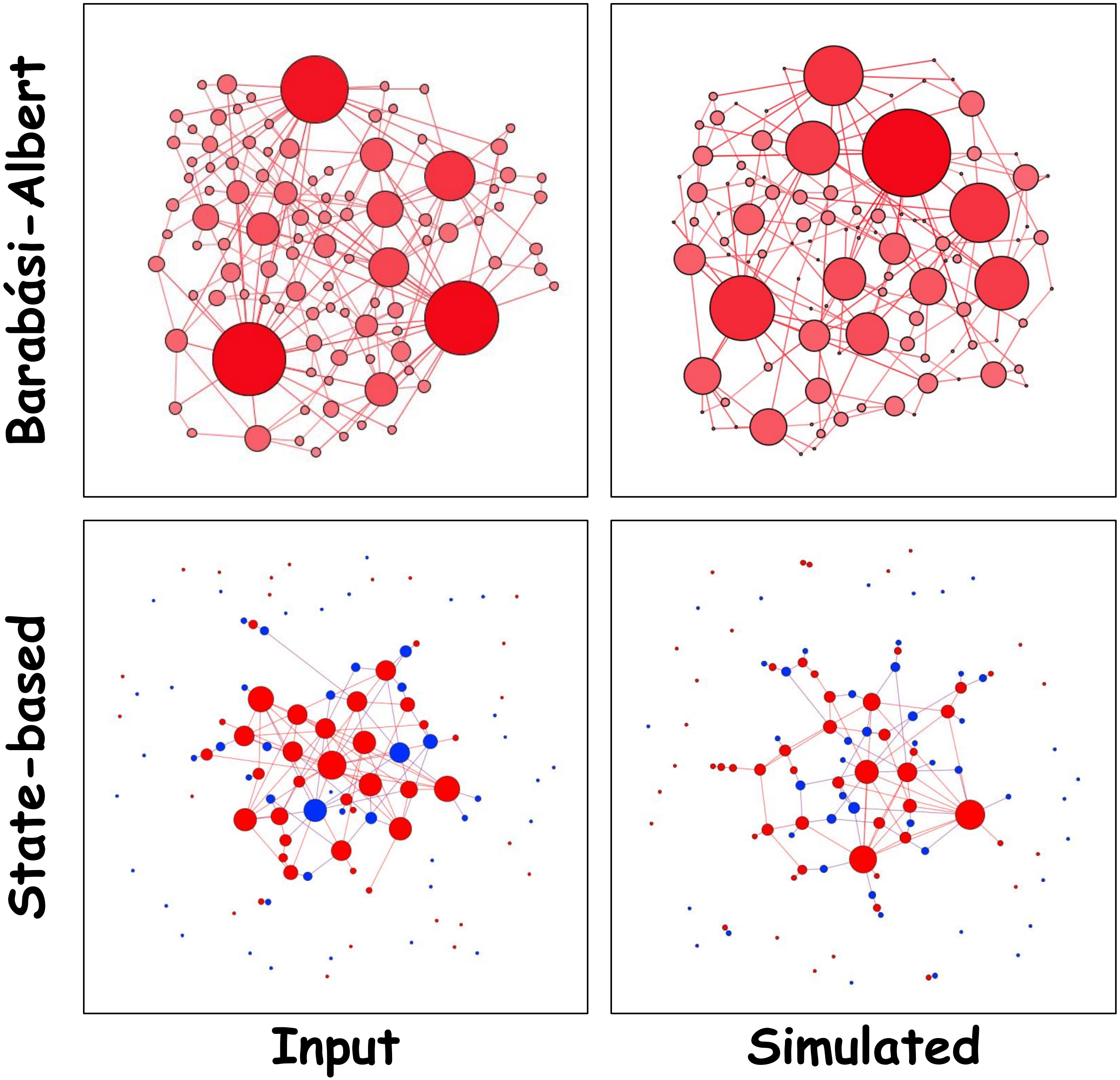
Algorithm

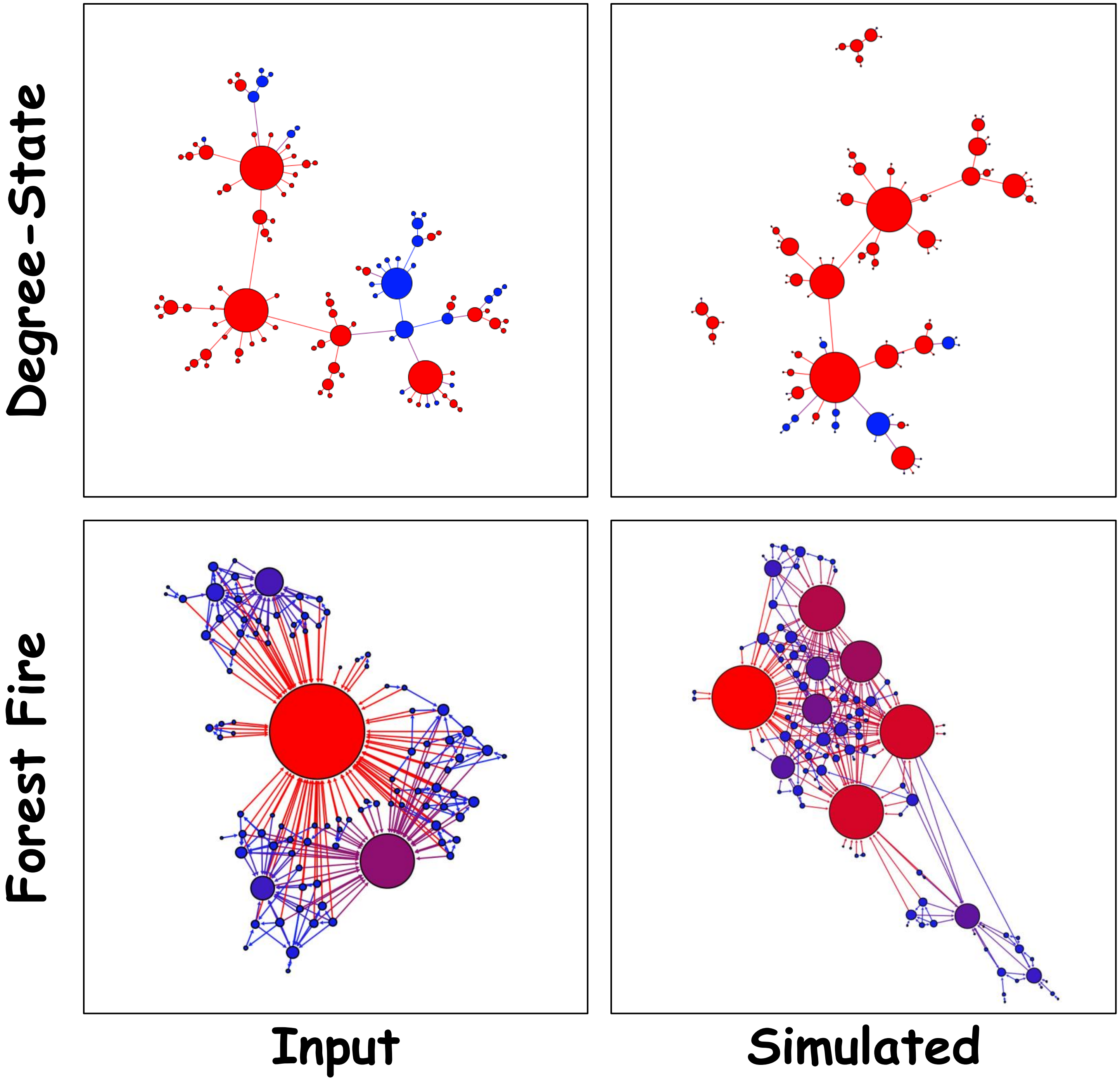

Match Extracted Subgraph to UES Data Structure
Extraction Mechanism
Select Rewriting Rule Preferentially By Occurrence
0
1
Generate Random #
Find Match
Selected Rewriting Rule

Results

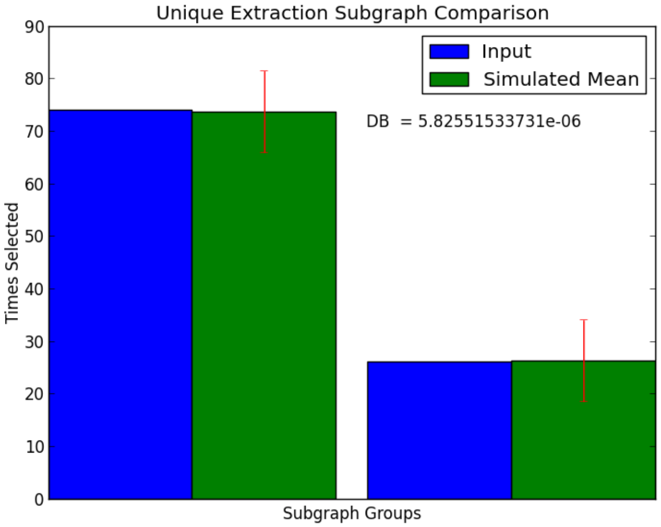
- **Example: "Degree-state" networks**



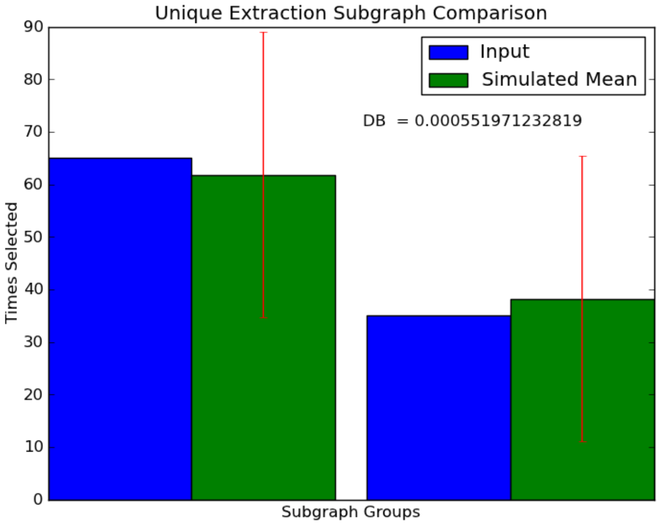

Barabási-Albert
State-based
Input
Simulated


Degree-State
Forest Fire
Input
Simulated

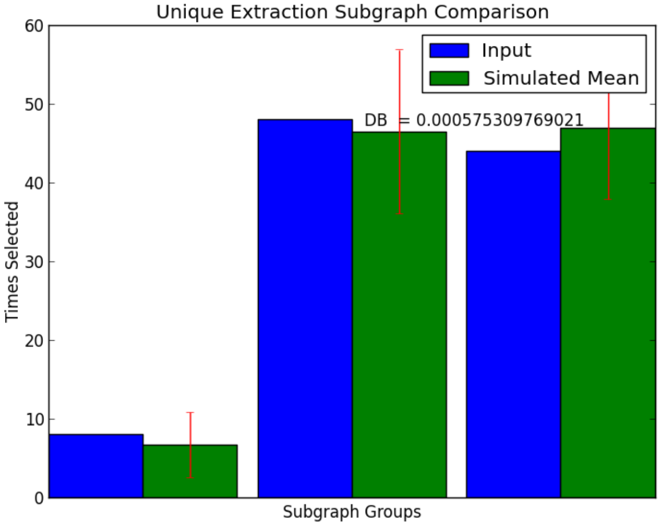
Barabási-Albert



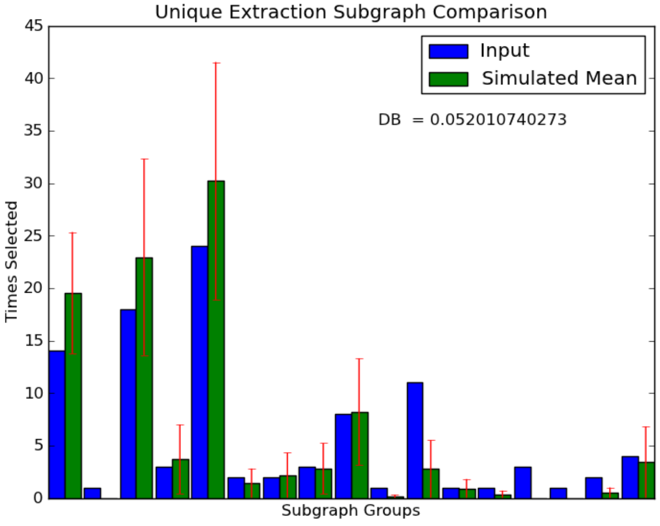
Degree-state



State-based



Forest Fire

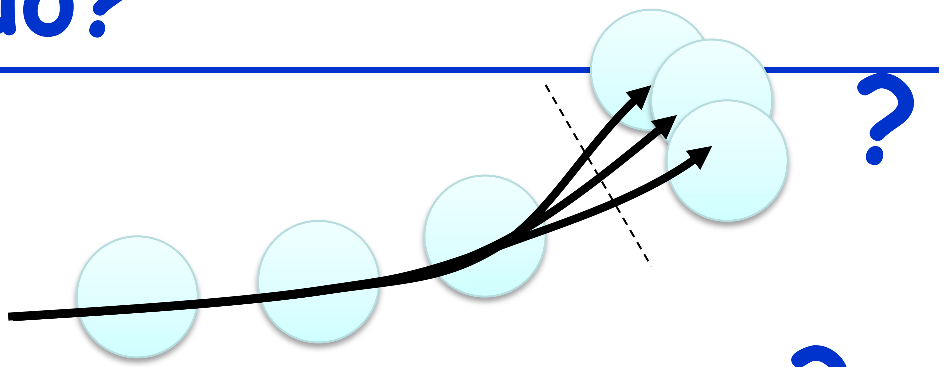


Comparison with other methods

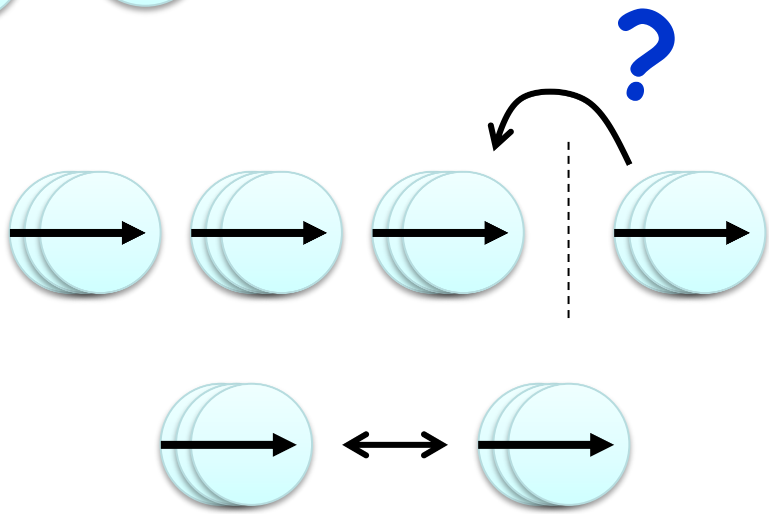
- PyGNA produces *generative models* using detailed state-topology information
  - Capable of generative simulation that is not available in statistical approaches (e.g., Rossi et al. 2013)
- PyGNA models extraction and replacement as *explicit functions*
  - More efficient and flexible than graph-grammars (e.g., Kurth et al. 2005)

What can we do?

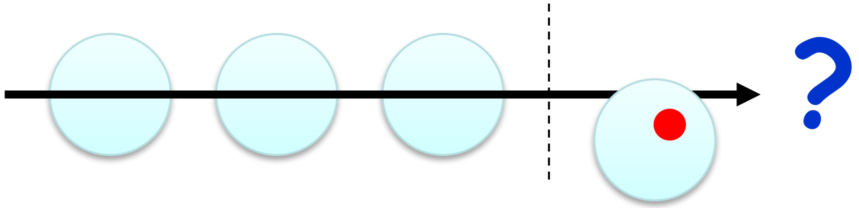
- Prediction



- Classification



- Anomaly detection



Summary

- State-topology coevolution of adaptive networks is a promising, unexplored area
  - Theory-driven approaches
    - Dynamical modeling, exhaustive rule search
    - Applications to social sciences etc.
  - Data-driven approaches
    - Application to operational network modeling
    - Automatic rule discovery from data

http://coco.binghamton.edu/NSF-CDI.html

Additional Topic:

Analysis of Adaptive Networks

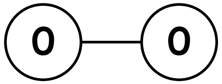
How to analyze adaptive network dynamics?

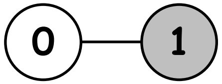
- Non-trivial coupling between network states and topologies are not easily handled in a simple analytical formula

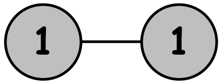
- But such couplings could be partially incorporated in analysis by considering densities of node "pairs"

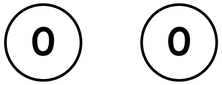
Pair approximation

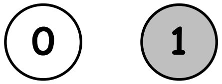
- Considers densities of every pair of nodes with states & connectivity (in addition to individual state densities)

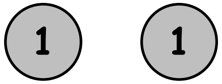
ρ_{00c} = density of 

ρ_{01c} = density of 

ρ_{11c} = density of 

ρ_{00n} = density of 

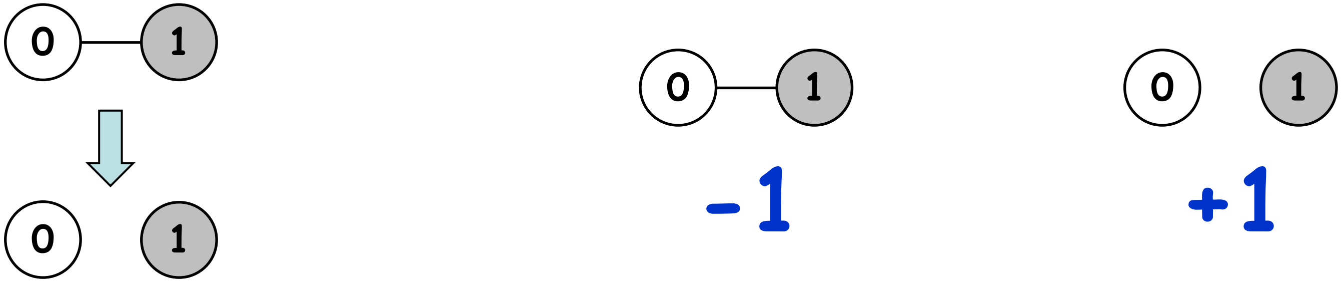
ρ_{01n} = density of 

ρ_{11n} = density of 

Describes how these densities change over time

Example: Adaptive voter model

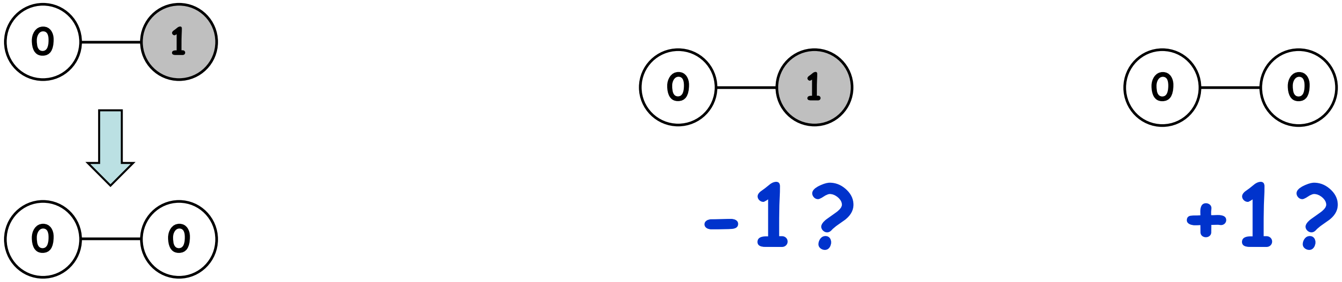
- Disconnect of a link:



-1

+1

- Change of an opinion:



-1 ?

+1 ?

(Any other densities affected too?)

Exercise

- Complete the number of changes in each pair density for the adaptive voter model on a random network
- Calculate how often each transition occurs
- Make a prediction using the pair-approximation-based model

Exercise

- Conduct pair approximation of the adaptive SIS model and study its dynamics

FYI: Moment closure

- Similar approximations are possible for densities of higher-order motifs
- Approximation techniques (including MFA, PA and higher-order ones) is called the "**moment closure method**"
  - Predicting the change of a "moment" (ρ_{00}) would require higher-order "moments" (ρ_{000}), but you "close" this open chain by assuming $\rho_{000} = \rho_{00}\,\rho_{00} / \rho_{0}$, etc.