

707.000  
Web Science and Web Technology  
„Network Evolution and Processes“

**Markus Strohmaier**

Univ. Ass. / Assistant Professor  
Knowledge Management Institute  
Graz University of Technology, Austria

e-mail: [markus.strohmaier@tugraz.at](mailto:markus.strohmaier@tugraz.at)  
web: <http://www.kmi.tugraz.at/staff/markus>

## Overview

### Agenda

- Network Creation and Evolution
  - Random Networks, Configuration Model, Barabasi and Albert
- Network Processes
  - The SIR Model
- Pajek
  - A social network analysis tool

# Administrative Issues

- **ATTENTION:** Next week's lecture
  - Nov 5, Mo, 18:00-19:30

# Preliminary Course Schedule I/II

Week	Date	Title, Links	Comments and Links
Week 1	2.10.2007	<b>Introduction and Motivation: Web &amp; Science</b> (slides, home assignment 1)	In this class, we will discuss the course organization and provide a basic motivation for and introduction to the course. <b>Readings:</b> Web science: a provocative invitation to computer science, B. Shneiderman, Communications of the ACM 50 25-27 (2007) [ <a href="#">Web link</a> ] <b>Readings:</b> Chapter 1 & 2, A Framework for Web Science, T. Berners-Lee and W. Hall and J. A. Hendler and K. O'Hara and N. Shadbolt and D. J. Wetzner Foundations and Trends® in Web Science 1 (2006) [ <a href="#">Web link</a> ]
Week 2	9.10.2007	<b>The Small World Problem</b> home assignment 1 due (slides)	We will discuss several examples and research efforts related to the small world problem and set the ground for our discussion of network theory and social network analysis. <b>Readings:</b> An Experimental Study of the Small World Problem, J. Travers and S. Milgram Sociometry 32 425-443 (1969) [ <a href="#">Protected Access</a> ]
Week 3	16.10.2007	<b>Network Theory and Terminology</b> (slides, home assignment 2)	In this class, we will discuss network theory fundamentals, including concepts such as diameter, distance, clustering coefficient and others. We will also discuss different types of networks, such as scale-free networks, random networks etc. <b>Readings:</b> Graph structure in the Web, A. Broder and R. Kumar and F. Maghoul and P. Raghavan and S. Rajagopalan and R. Stata and A. Tomkins and J. Wiener Computer Networks 33 309-320 (2000) [ <a href="#">Web link</a> ]
Week 4	23.10.2007	<b>Social Network Analysis</b> home assignment 2 due (slides, home assignment)	What information can you get out of social graphs? We will discuss some basic principles of social network analysis.
Week 5	30.10.2007	<b>Network Evolution</b> home assignment 3 due (slides, home assignment 4)	In this class, we will discuss the nature of network evolution and some selected network processes.
Week 6	6.11.2007	<b>Link Analysis</b> home assignment 4 due (slides, home assignment 5)	What are ways of searching in graphs? In this class, we will discuss basics of link analysis, including Google's PageRank algorithm as an example. <b>Readings:</b> The PageRank Citation Ranking: Bringing Order to the Web, L. Page and S. Brin and R. Motwani and T. Winograd (1998) [ <a href="#">Protected Access</a> ]

Knowledge Management Institute		TU Graz	
<h2>Preliminary Course Schedule II/II</h2>			
Week 7	13.11.2007	Web Mining and Information Retrieval home assignment 5 due (slides)	This class introduces basics of web mining and information retrieval including an introduction to the Vector Space Model, Latent Semantic Indexing, Associative Retrieval and Support Vector Machines.  Guest lecture: Michael Granitzer, Know-Center Graz
Week 8	20.11.2007	Webtechnologies I home assignment 6	This class focuses on a selected subset of web technologies that are of current interest  Read: TBA
Week 9	27.11.2007	Metadata, Tagging and Folksonomies (slides)	In this class, we will discuss metadata as well as current phenomena such as tagging and folksonomies.  Readings: P. Mika, Ontologies Are Us: A Unified Model of Social Networks and Semantics, International Semantic Web Conference, - 522-536, 2005. [Web link]
Week 10	11.12.2007	Trust and Reputation on the Web	TBA  Readings: New Scientist article "Wikipedia 2.0 - now with added trust" [protected access]
Week 11	8.1.2008	User Intentions and Intentional Structures on the Web home assignment 6 due (slides)	Search engines - such as Google - have been characterized as "Databases of intentions". This class will focus on different aspects of intentionality on the web, including goal mining, goal modeling and goal-oriented search.  Readings: M. Strohmaier, M. Lux, M. Granitzer, P. Scheir, S. Liaskos, E. Yu, How Do Users Express Goals on the Web? - An Exploration of Intentional Structures in Web Search, We Know'07 International Workshop on Collaborative Knowledge Management for Web Information Systems in conjunction with WISE'07, Nancy, France, 2007. [Web link]  Readings: Automatic identification of user goals in Web search, U. Lee and Z. Liu and J. Cho WWW '05: Proceedings of the 14th International World Wide Web Conference 391--400 (2005) [Web link]
Week 12	15.1.2008	Webtechnologies II (slides)	The semantic web represents a current research effort to increase the capability of machines to make sense of content on the web. In this class, Peter Scheir will give a guest lecture on the basic principles underlying the semantic web vision, including RDF, OWL and other standards.  Guest lecture: Peter Scheir, Knowledge Management Institute, Graz University of Technology
Week 13	22.1.2008	Final Exam	No aids are allowed at the final exam.

Knowledge Management Institute		TU Graz	
<h2>Background [Newman 2003]</h2>			
<ul style="list-style-type: none"> <li>• First example of a scale-free network (Price): <ul style="list-style-type: none"> <li>– Network of citations between scientific papers</li> <li>– Both in- and out-degrees had power-law distributions</li> </ul> </li> <li>• Answered the question: How do power law distributions emerge? <ul style="list-style-type: none"> <li>– “the rich get richer”</li> <li>– In other words: the amount you get goes up with the amount you already have</li> </ul> </li> <li>• The “Matthew affect” <ul style="list-style-type: none"> <li>– “For to every one that hath shall be given” (Matthew 25:29)</li> <li>– (in german ~ “wer hat dem wird gegeben”)</li> </ul> </li> <li>• Other labels <ul style="list-style-type: none"> <li>– Cumulative advantage <span style="color: red;">why?</span></li> </ul> </li> </ul>			
Markus Strohmaier		Preferential attachment 2007	
• Evident in scientific paper citations			
6			

## Two Assumptions [Leskovec 2006]

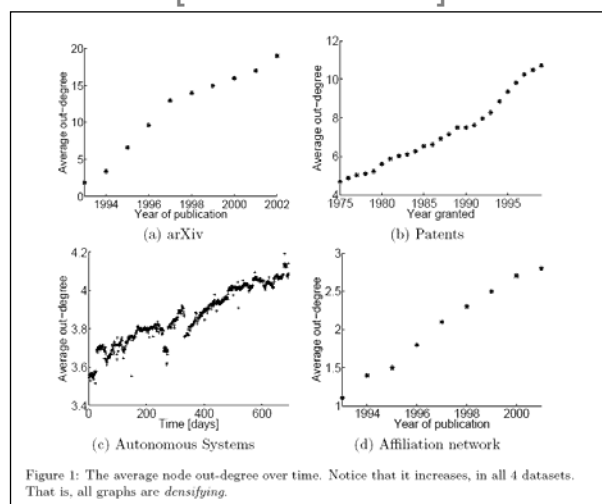
“Conventional Wisdom” that networks that evolve are characterized by

- Constant average degree
  - Edges grow linearly with edges
- Slowly growing diameter
  - Growing diameter with the addition of new nodes

Empirical observations show that

- Networks are becoming denser over time (densification power laws)
- Effective diameter is in many cases decreasing as networks grow (shrinking diameter)

## Empirical Observation: Densification [Leskovec 2006]



## Empirical Observation: Densification [Leskovec 2006]

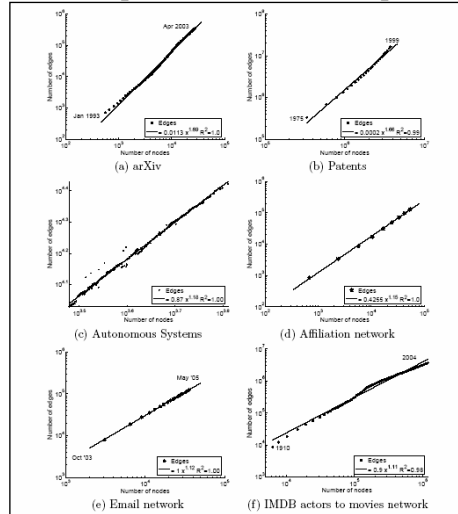


Figure 2: Number of edges  $e(t)$  versus number of nodes  $n(t)$ , in log-log scales, for several graphs. All 4 graphs obey the Densification Power Law, with a consistently good fit. Slopes:  $\alpha = 1.68, 1.66, 1.18, 1.12, 1.15, 1.12$  and  $1.11$  respectively.

## Empirical Observation: Effective Diameter [Leskovec 2006]

**Effective diameter**  
The minimum distance  $d$  such that at least 90% of the connected node pairs are at distance at most  $d$

Decreasing diameter over time

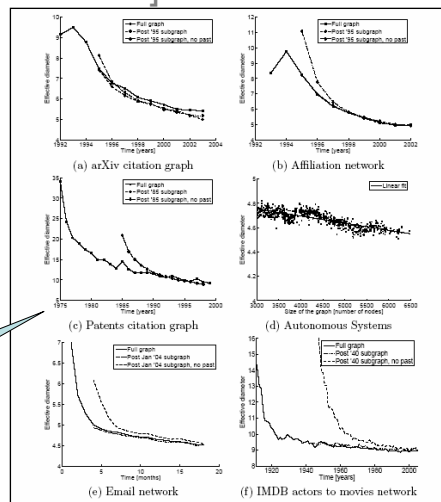


Figure 3: The effective diameter over time for 6 different datasets. Notice consistent decrease of diameter over time.

## Motivation [Leskovec 2006]

What underlying processes cause a graph to

- systematically densify?
- Experience a decrease in effective diameter even as its size increases?

But first, let's take a step back

## Graph Generators [Leskovec 2006]

Why are we interested in simulating graph evolution?

*“What if we could develop algorithms that are capable of constructing networks that exhibit similar characteristics as observed in “real-world” networks?”*

We could do interesting things, such as:

- **Extrapolations**
  - predicting future network development
- **Sampling**
  - Drawing a sample and generalizing to the entire population
- **Abnormality detection**
  - Identifying deviations from “normal” network behaviour
- **Simulation**
  - Exploring “what if” scenarios, e.g. deletion of hubs, network resilience

# Simple Graph Generators [Newman 2003]

Can we develop an algorithm that constructs random graphs?

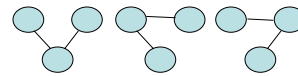
**Algorithm:**  
Take some number  $n$  of vertices and connect each pair (or not) with probability  $p$  (or  $1-p$ ). Done!

The Erdos-Renyi / Poisson random Graph

$G(n,m)$  the set of all graphs having  $n$  vertices and  $m$  edges, each possible graph appearing with equal probability

For example:  $G(3,2)$  is the set of all three graphs having 3 vertices and 2 edges, each graph has probability  $1/3$

->Does not mimic reality

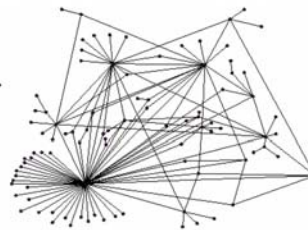


## Poisson vs. Scale-free network

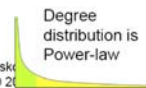
Faloutsos / Leskovec  
ECML/PKDD 2007



Poisson network  
(Erdos-Renyi random graph)



Scale-free (power-law) network



Function is scale free if:  
 $f(ax) = c f(x)$

## Random Graphs

[Faloutsos / Leskovec ECML/PKDD 2007]

- Pros:
  - Simple and tractable model
  - Phase transitions
  - Giant component
- Cons:
  - Degree distribution
  - No community structure
  - No degree correlations
- Extensions:
  - Configuration model
    - Random graphs with arbitrary degree sequence

## The Configuration Model

Consider the model defined in the following way.

We specify a degree distribution  $p_k$ , such that  $p_k$  is the fraction of vertices in the network having degree  $k$ .

We choose a degree sequence, which is a set of  $n$  values of the degrees  $k_i$  of vertices  $i = 1 \dots n$ , from this distribution. We can think of this as giving each vertex  $i$  in our graph  $k_i$  “stubs” or “spokes” sticking out of it, which are the ends of edges-to-be.

[Newman 2003]

## The Configuration Model

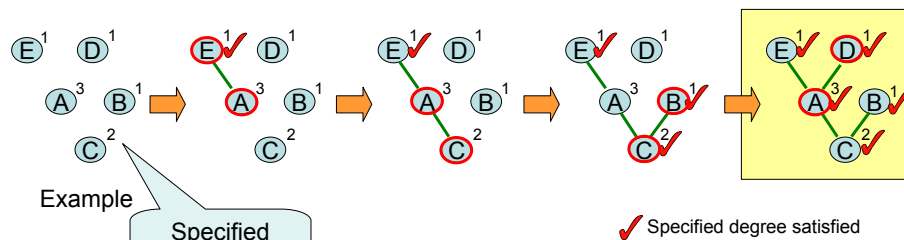
Then we choose pairs of stubs at random from the network and connect them together. It is straightforward to demonstrate that this process generates every possible topology of a graph with the given degree sequence with equal probability.

The configuration model is defined as the ensemble of graphs so produced, with each having equal weight.

[Newman 2003]

## The Configuration Model: Example

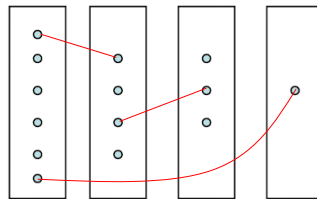
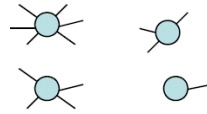
1. Define a degree distribution (e.g. 3,2,1,1,1)
2. Specify degrees for each node, based on the degree distribution (e.g. A->3, B->2, C->1, D->1, E->1)
3. Insert an edge between two arbitrary nodes in your node set that have not satisfied their specified degree yet.
4. Repeat step 3 until all node degrees are satisfied.



## The Configuration Model: Example II

Another perspective:

Configuration model



Example

Faloutsos / Leskovec  
ECML/PKDD 2007

## Generating Scale Free Networks

[Barabasi and Albert 1999]

To incorporate the growing character of the network, starting with a small number ( $m_0$ ) of vertices, at every time step we add a new vertex with  $m(\leq m_0)$  edges that link the new vertex to  $m$  different vertices already present in the system.

To incorporate preferential attachment, we assume that the probability  $\Pi$  that a new vertex will be connected to vertex  $i$  depends on the connectivity  $k_i$  of that vertex, so that

Probability of a new vertex attaching to a vertex  $i$  with degree  $k$

$$\Pi(k_i) = k_i / \sum_j k_j$$

Degree of vertex  $i$

The sum of all vertices' degrees

In other words: the probability is the degree of vertex  $i$  divided by the sum of all nodes' degrees

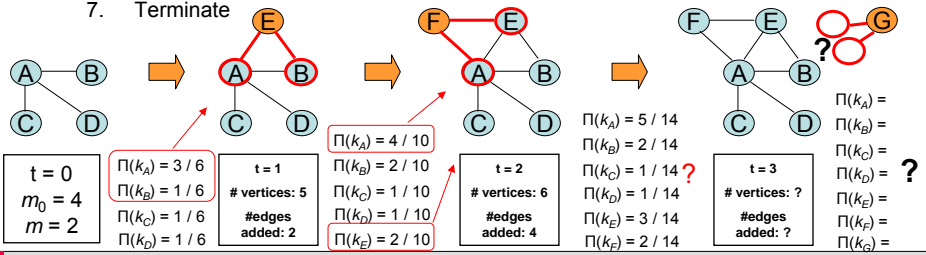
After  $t$  time steps, the model leads to a random network with  $t+m_0$  vertices and  $mt$  edges.

This network evolves into a scale-invariant state following a power law (satisfies the two conditions: Growth and Preferential Attachment).

## Generating Scale Free Networks [Barabasi and Albert 1999]

Example:

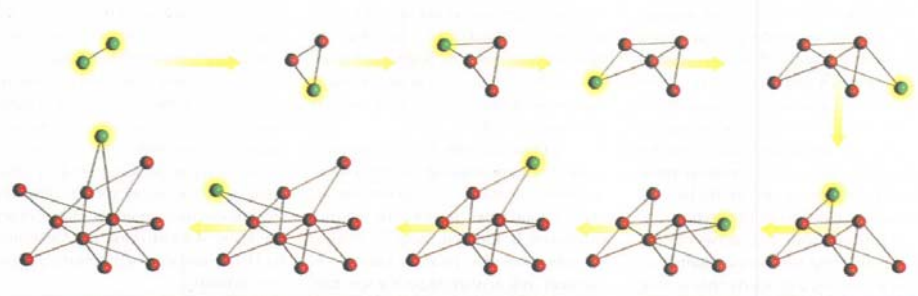
1. Specify a starting network with a given number of vertices  $m_0$  and an initial set of edges (e.g.: #edges = 3); initialize  $t=0$
2. Define the number of vertices a new node is required to link to (e.g.  $m=2$ )
3. Calculate the probabilities  $\Pi$  that a new vertex will be connected to vertex  $i$  by calculating  $\Pi(k_i) = k_i / \sum_j k_j$
4. Add the new vertex. Add edges according to the calculated probabilities and  $m$
5. Set  $t = t+1$
6. While  $t \leq 3$  Goto Step 3.
7. Terminate



## Generating Scale Free Networks [Barabasi and Albert 2003]

### BIRTH OF A SCALE-FREE NETWORK

A SCALE-FREE NETWORK grows incrementally from two to 11 nodes in this example. When deciding where to establish a link, a new node (green) prefers to attach to an existing node (red) that already has many other connections. These two basic mechanisms—growth and preferential attachment—will eventually lead to the system's being dominated by hubs, nodes having an enormous number of links.



## Generating Scale Free Networks

[Barabasi and Albert 1999]

Because of preferential attachment, a vertex that acquires more connections than another one will increase its connectivity at a higher rate; thus, an **initial difference** in the connectivity between two vertices **will increase further** as the network grows.

Thus **older** (with smaller  $t_i$ ) **vertices increase their connectivity at the expense of the younger** (with larger  $t_i$ ) ones, leading over time to some vertices that are highly connected, a “**rich-get-richer**” phenomenon that can be easily detected in real networks.

But, [Faloutsos / Leskovec ECML/PKDD 2007]

- all nodes have equal (constant) outdegree (in a directed network)
- one needs a complete knowledge of the network

## Edge copying model

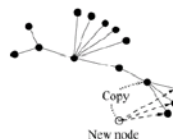
[Faloutsos / Leskovec ECML/PKDD 2007]



CMU SCS

### Edge copying model

- **But**, preferential attachment does not have communities
- Copying model [Kleinberg et al, 99]:
  - Add a node and choose  $k$  the number of edges to add
  - With prob.  $\beta$  select  $k$  random vertices and link to them
  - With prob.  $1-\beta$  edges are copied from a randomly chosen node
- Generates power-law degree distributions with exponent  $1/(1-\beta)$
- Generates communities



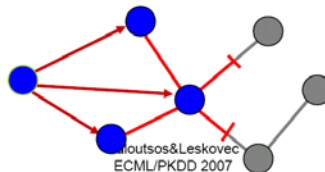
## Forest Fire Model

[Faloutsos / Leskovec ECML/PKDD 2007]



### Forest Fire Model

- **But**, we do not want to have explicit communities
- Want to model graphs that density and have shrinking diameters
- Intuition:
  - How do we meet friends at a party?
  - How do we identify references when writing papers?



## Forest Fire Model

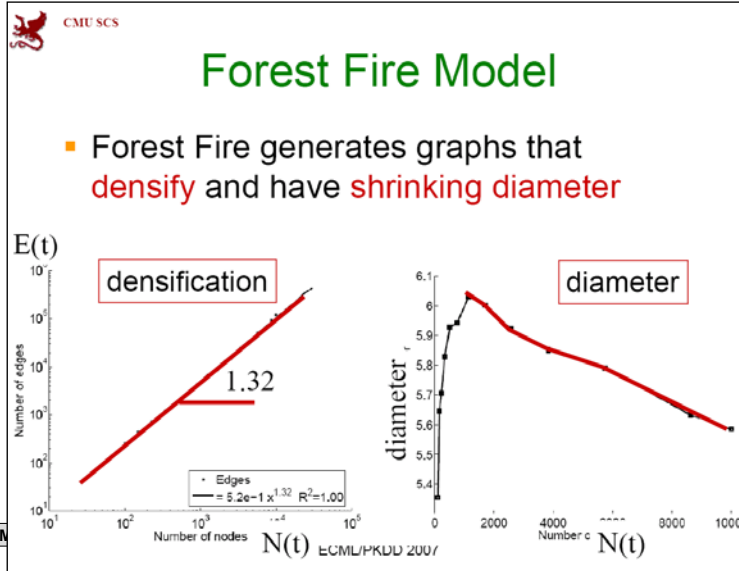
[Faloutsos / Leskovec ECML/PKDD 2007]



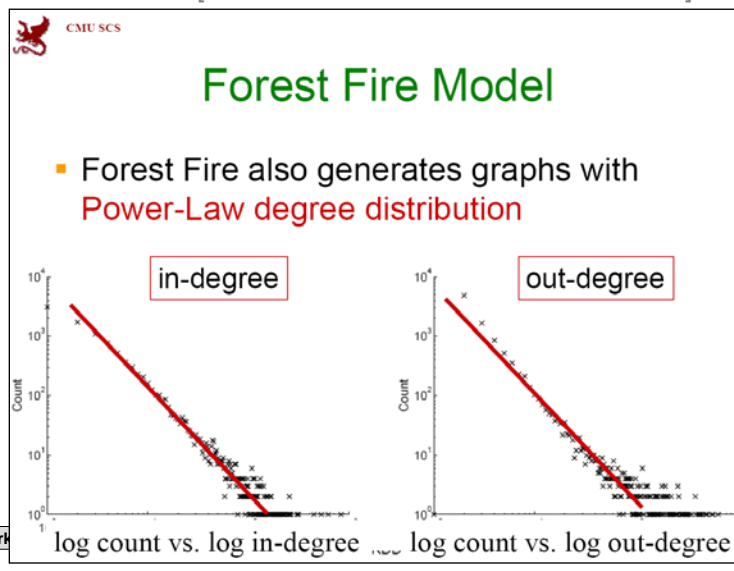
### Forest Fire Model

- The Forest Fire model [KDD05] has 2 parameters:
  - $p$  ... forward burning probability
  - $r$  ... backward burning probability
- The model:
  - Each turn a new node  $v$  arrives
  - Uniformly at random chooses an "ambassador"  $w$
  - Flip two geometric coins to determine the number in- and out-links of  $w$  to follow (burn)
  - Fire spreads recursively until it dies
  - Node  $v$  links to all burned nodes

Forest Fire Model  
[Faloutsos / Leskovec ECML/PKDD 2007]



Forest Fire Model  
[Faloutsos / Leskovec ECML/PKDD 2007]



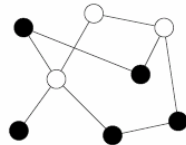
## Percolation Theory [Newman 2003]

A percolation process is one in which vertices or edges on a graph are randomly designated either “occupied” or “unoccupied”.

One of the main motivations for the percolation model when it was first proposed in the 1950s was the modeling of the spread of disease.

## Percolation Theory [Newman 2003]

*Why are we interested in percolation theory in the context of web science?*



site percolation



bond percolation

FIG. 13 Site and bond percolation on a network. In site percolation, vertices (“sites” in the physics parlance) are either occupied (solid circles) or unoccupied (open circles) and studies focus on the shape and size of the contiguous clusters of occupied sites, of which there are three in this small example. In bond percolation, it is the edges (“bonds” in physics) that are occupied or not (black or gray lines) and the vertices that are connected together by occupied edges that form the clusters of interest.

## Two Fundamental Network Process Distinctions [Newman 2003]

Can you name examples  
of these processes on the  
web?

### Epidemic processes

- such as influenza, which sweeps through the population rapidly and infects a significant fraction of individuals in a short outbreak (cf. the SIR model)

### Endemic processes

- such as measles, which persists within the population at a level roughly constant over time. The disease can persist indefinitely, circulating around the population and never dying out (cf. the SIS model)

## The SIR Model [Watts 2004]

The SIR model of network epidemics

<b>S</b>	<b>Susceptible</b> Vulnerable to infection, but not yet been infected
<b>I</b>	<b>Infected</b> infected and infectious (can infect others)
<b>R</b>	<b>Removed</b> either recovered or ceased to pose a threat

### Rules:

- New infections can only occur when an infected individual (an infective) comes into direct contact with a susceptible.
- The susceptible can become infected, with probability  $p$  depending on infectiousness of the disease and the characteristics of the susceptible
- Who comes into contact with whom will depend on the populations' network structure.

## The SIR Model [Watts 2004]

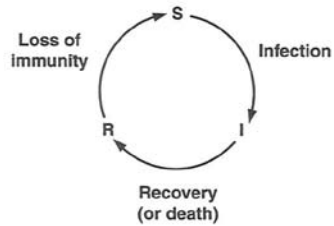


Figure 6.1. The three states of the SIR model. Each member of the population can be susceptible, infected, or removed. Susceptible individuals can become infected by interacting with infectives. Infectives can either recover or die, thus ceasing to take part in the dynamics. If they recover, they might become susceptible again through loss of immunity.

## The SIR Model [Watts 2004]

In its simplest version,

- based on purely random interactions
- Rate of infection depends only on the relative population sizes

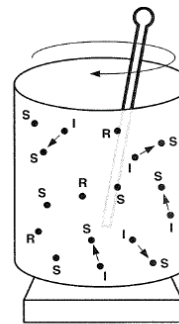


Figure 6.2. In the classical version of the SIR model, interactions are assumed to be purely random. One way to think of random interactions is as individuals being mixed together in a large vat. The main consequence of the random mixing assumption is that interaction probabilities depend only on the relative population sizes, a feature that greatly simplifies analysis.

## The SIR Model [Watts 2004]

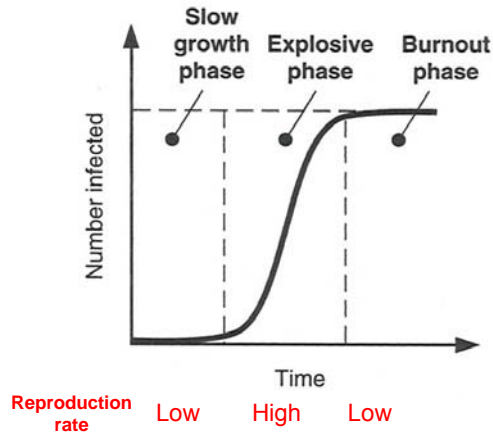


Figure 6.4. Logistic growth, displaying the slow-growth phase, explosive phase, and burnout phase.

In terms of the SIR model, stopping an epidemic is roughly equivalent to preventing it from reaching the explosive growth phase.

This implies focusing **not** on the size of the initial outbreak but on its rate of growth.

## The SIR Model [Watts 2004]

Each infection requires the participation of both an infected and a susceptible individual.

The rate at which new infections can be generated depends on the size of both populations.

**Reproduction rate:** the average number of new infectives generated by each currently infected.

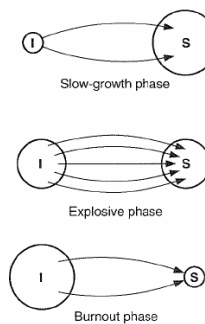


Figure 6.3. In logistic growth, the rate of new infections depends on the size of the susceptible and infected populations. When either population is small (top and bottom diagrams), new infections are rare. But when both populations are intermediate in size (middle diagram), infection rates are maximized.

## The SIR Model [Watts 2004]

Condition for epidemics: reproduction rate  $> 1$  (threshold)

Note: That's the same threshold at which a giant component occurs in networks

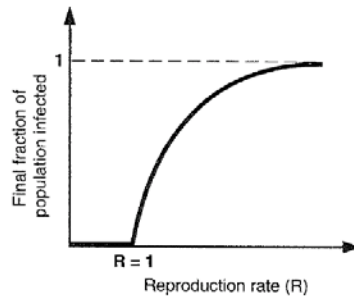


Figure 6.5. Phase transition in the SIR model. When the reproduction rate ( $R$ ) of the disease exceeds one (the epidemic threshold), an epidemic occurs.

**SIR simulation:** e.g.

[http://www.uni-tuebingen.de/modeling/Mod\\_Pub\\_Software\\_SIR\\_en.html](http://www.uni-tuebingen.de/modeling/Mod_Pub_Software_SIR_en.html)

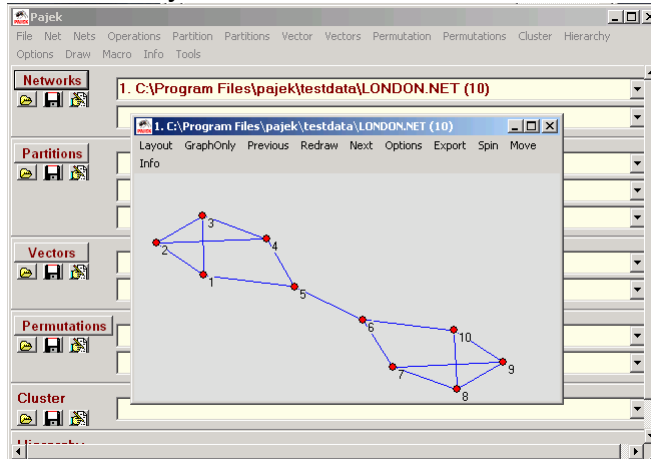
## Applications of Graph Generators and Growth Models [Leskovec 2006]

Recapitulation:

- „What if“ scenarios
- Forecasting future parameters of computer and social networks
- Anomaly detection
- Graph sampling algorithms
- Realistic graph generators

# Pajek

## A Social Network Analysis Tool



# Pajek

## Examples: The Pajek .net format

One mode network

```

home assignment2.n...
File Edit Format View Help
*Vertices 7
1 "A"
2 "B"
3 "C"
4 "D"
5 "E"
6 "F"
7 "G"
*Edges
1 4 1
1 3 1
3 4 1
2 3 1
2 6 1
6 7 1
5 7 1
3 5 1
    
```

Two mode network

```

home assignment3.net - Not...
File Edit Format View Help
*Vertices
1 "A" 10 5
2 "B"
3 "C"
4 "D"
5 "E"
6 "I"
7 "II"
8 "III"
9 "IV"
10 "V"
*Edges
1 6 1
1 7 1
1 8 1
2 7 1
2 8 1
3 9 1
4 10 1
5 8 1
5 10 1
    
```

## Pajek

Manuals, Documentation and Tutorials:

- “Official” website: <http://vlado.fmf.uni-lj.si/pub/networks/pajek/>
- “Official” manual: <http://vlado.fmf.uni-lj.si/pub/networks/pajek/doc/pajekman.pdf>
- A tutorial: <http://vw.indiana.edu/tutorials/pajek/>
- Some notes: <http://www.itee.uq.edu.au/~hallinan/ACCSWinterSchool/PajekTutorial.doc>

Any questions?

**See you next week!**