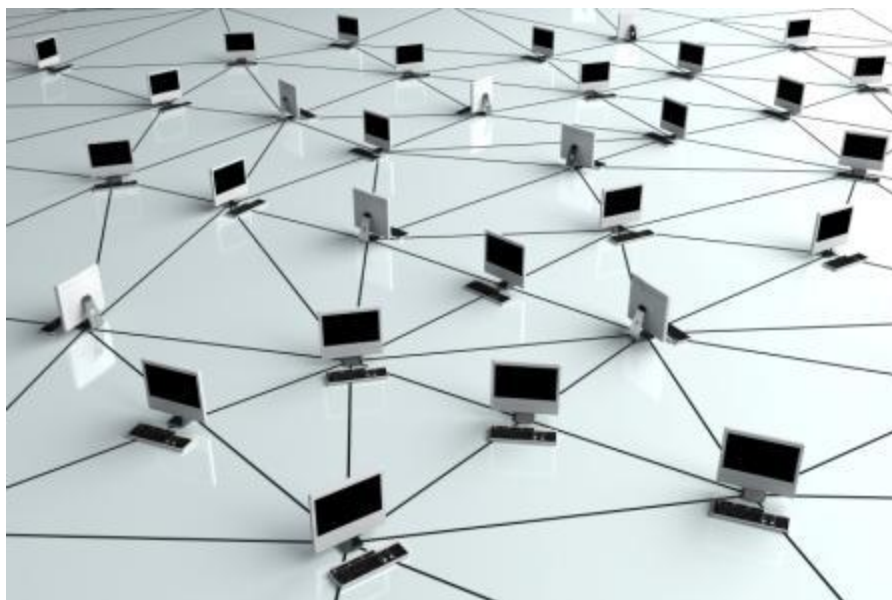




# IT Research

ICoSCIS



European Territorial Cooperation Programme  
**Greece-Bulgaria 2007-2013**  
INVESTING IN OUR FUTURE

## Blagoevgrad 2013

European Territorial Cooperation Programme  
**Greece - Bulgaria 2007 - 2013**

This Project is co-funded by the European Union (ERDF)  
and National Funds of Greece and Bulgaria



# What is IT research in the context of Complex Systems?

It deals mostly with:

- Communication Networks
- IT related infrastructure
- Virus spreading
- Targeted attacks
- Network infrastructure robustness
- etc.

# Communication networks

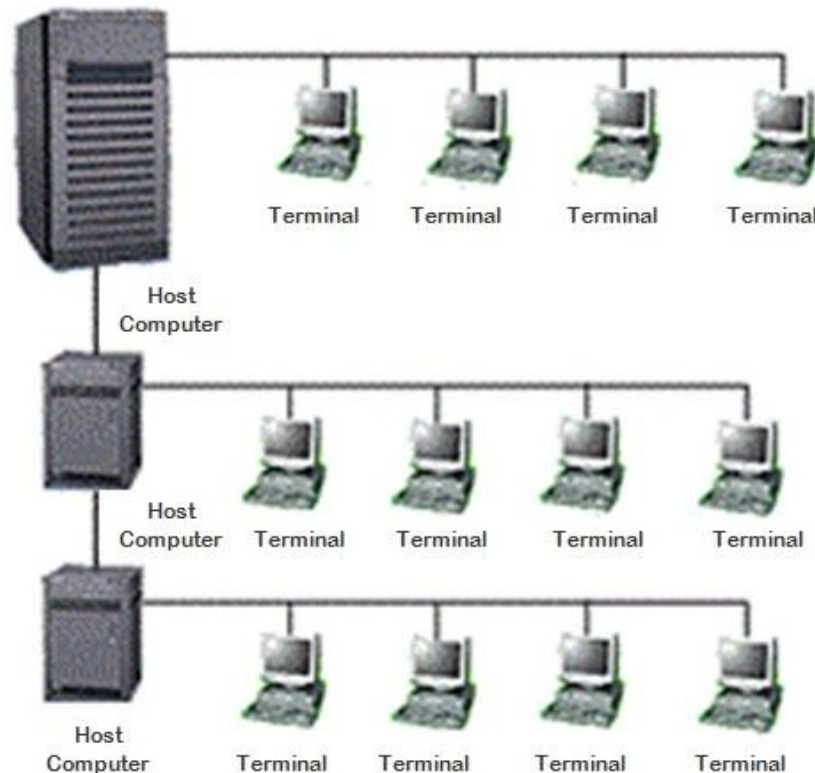
- A telecommunications network is a collection of terminal nodes, links and any intermediate nodes which are connected so as to enable telecommunication between the terminals.

# Communication networks

Example of telecommunications networks

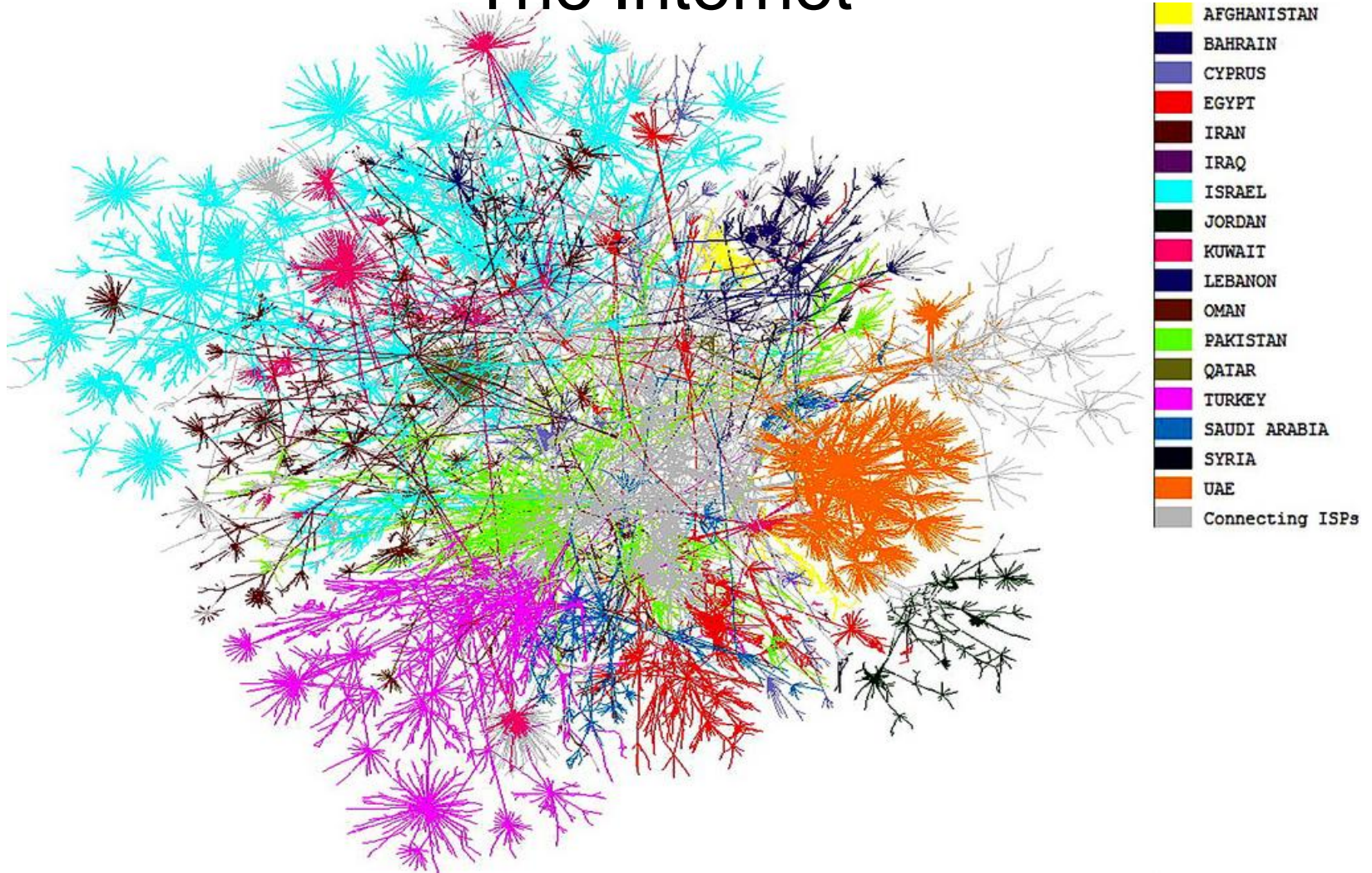
## Computer networks

Distributed Processing



# Communication networks

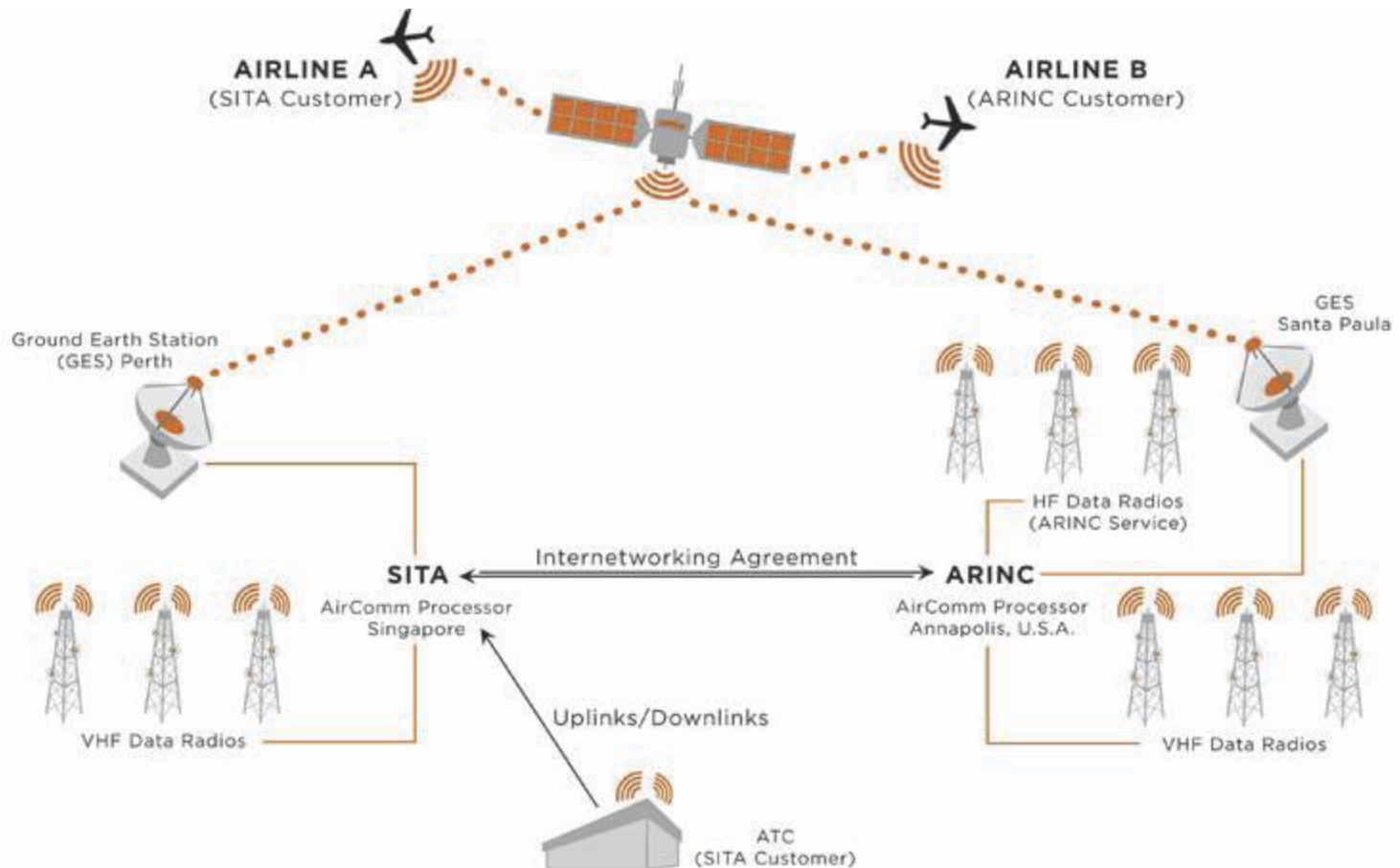
## The Internet





# Communication networks

## Aeronautical networks



# Communication networks

## Communication of communication networks



# Communication network components

- Terminals. Any input or output device used to transmit or receive data. For example, a telephone is a terminal.
- Channels. Copper wires, fiber-optic and coaxial cables (structured cabling) and wireless radio frequencies.



# Communication network components

- Nodes. They have three or more channels, that switch information between links to direct it towards its terminal node.
- Processors. These convert data from digital to analog and back.
- Software. Present on all networked computers and responsible for controlling network activities and functionality.

# Communication network example

The TCP/IP data network

- TCP/IP are the fundamental protocols that provide the control and routing of messages across the data network. There are many different network structures that TCP/IP can be used across to efficiently route messages.

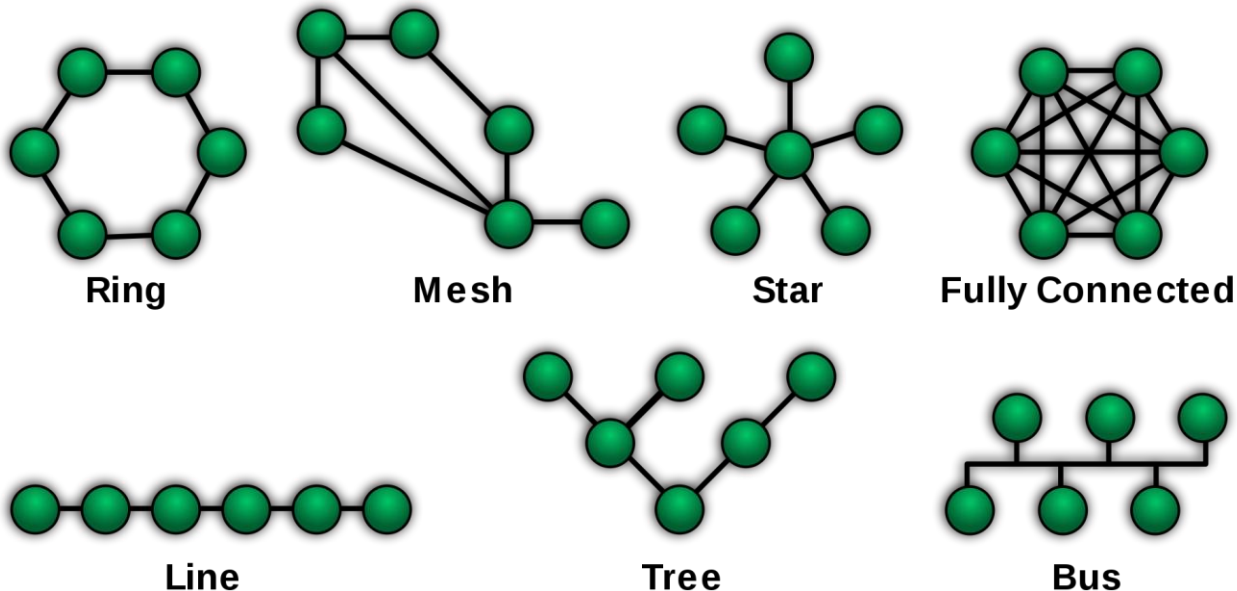
# Communication network example

Example structures used in TCP/IP:

- wide area networks (WAN)
- metropolitan area networks (MAN)
- local area networks (LAN)
- campus area networks (CAN)
- virtual private networks (VPN)

# Topology effect

Different topologies have different properties and target different services



# Network topology

Common layouts are:

- Bus network: all nodes are connected to a common medium. (used in the original Ethernet - 10BASE5 and 10BASE2).
- Star network: all nodes are connected to a special central node. This is the typical layout found in Wireless LAN, where each client connects to the central access point.

# Network topology

Common layouts are:

- Ring network: each node is connected to its left and right neighbor node. All nodes are connected and each node can reach every other node by traversing nodes left- or rightwards. The Fiber Distributed Data Interface (FDDI) made use of such a topology.



# Network topology

Common layouts are:

- Mesh network: each node is connected to an arbitrary number of neighbors in such a way that there is at least one traversal from any node to any other.
- Fully connected network: each node is connected to every other node in the network.

# Virus spreading models in communication systems

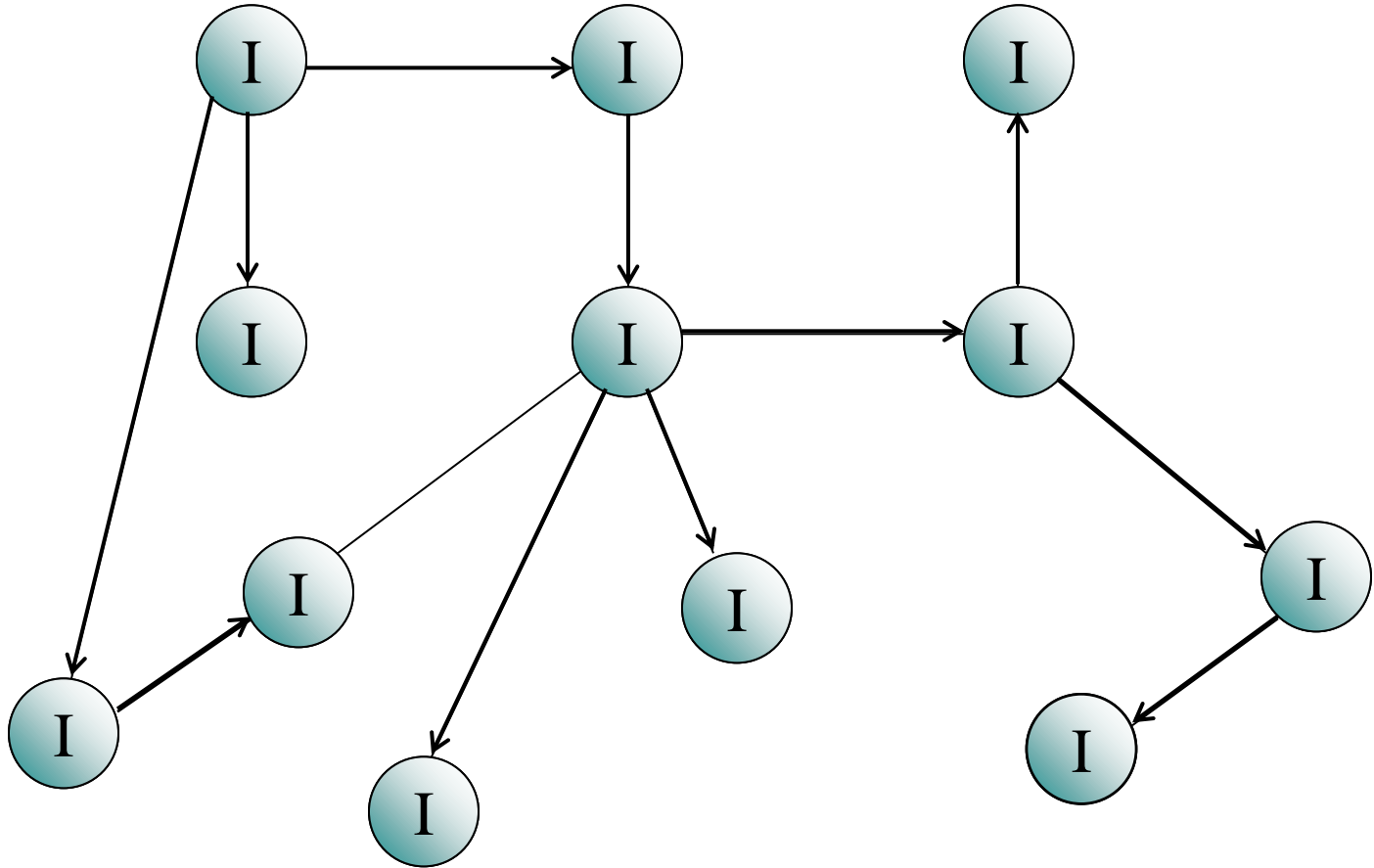
- Most virus spreading models are based on the original SI model.
- The SI model was introduced in the 1920s to explain how human viruses and diseases are transmitted.
- Since in many aspects computers are like humans when it comes to viruses these models are used in computer science too.

# Virus spreading models in communication systems

- We simply think of a computer as a node (human) and the link is the wired/wireless connection (human contact).
- In this simple model all nodes are in a susceptible (meaning that they can be infected) or in an infected state.
- Many variants exist.
- An infection transmission rate can be applied.

CASE 2

SI



**S** Susceptible

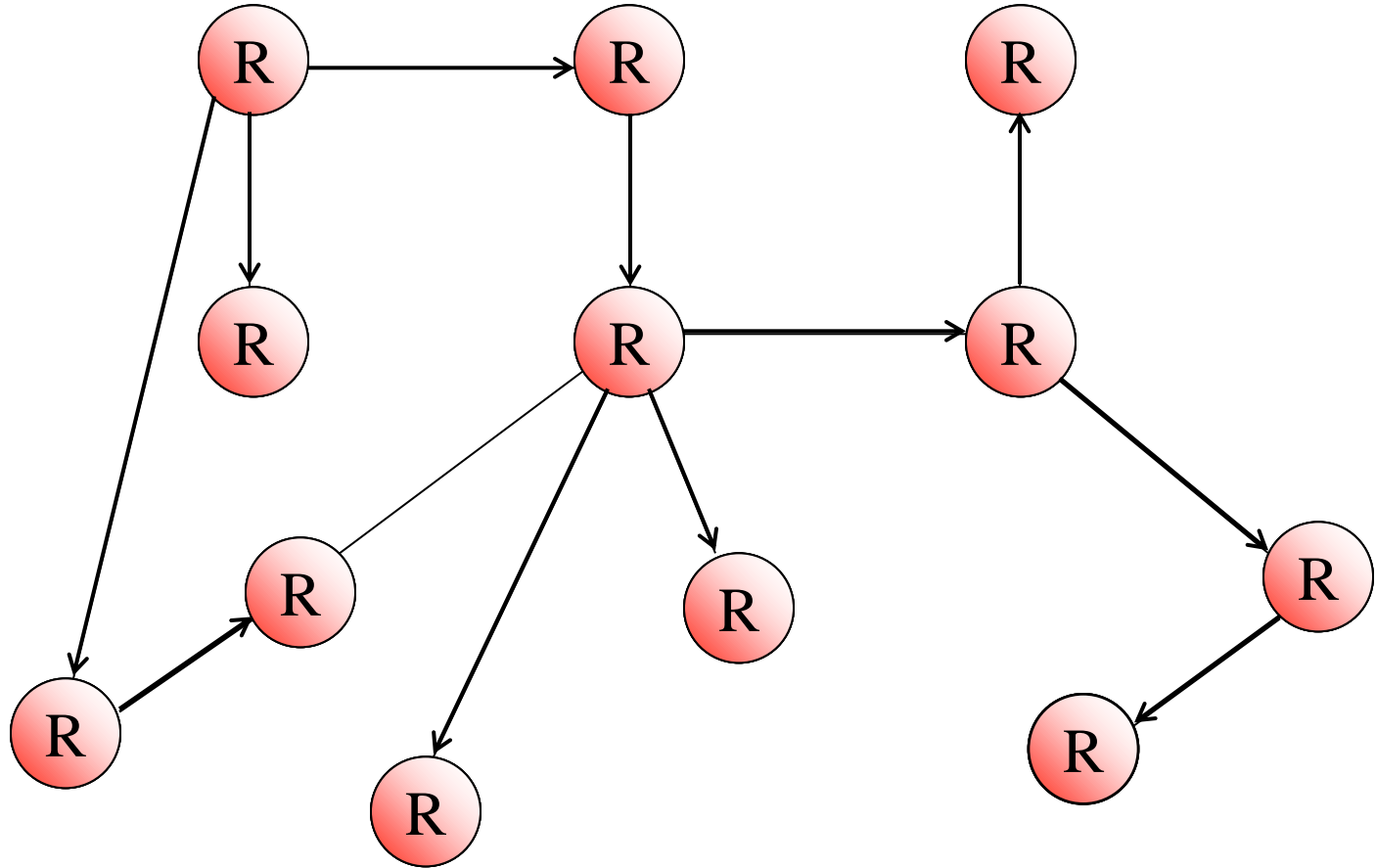
**I** Infected

# SI model variants - SIR

- Simplest of all variants is the SIR model.
- Every node is susceptible, then it becomes infected and after that it recovers (or it is removed from the network).
- A recovery rate can be applied.

CASE 2

# SIR



**S** Susceptible

**I** Infected

**R** Recovered (or Removed)

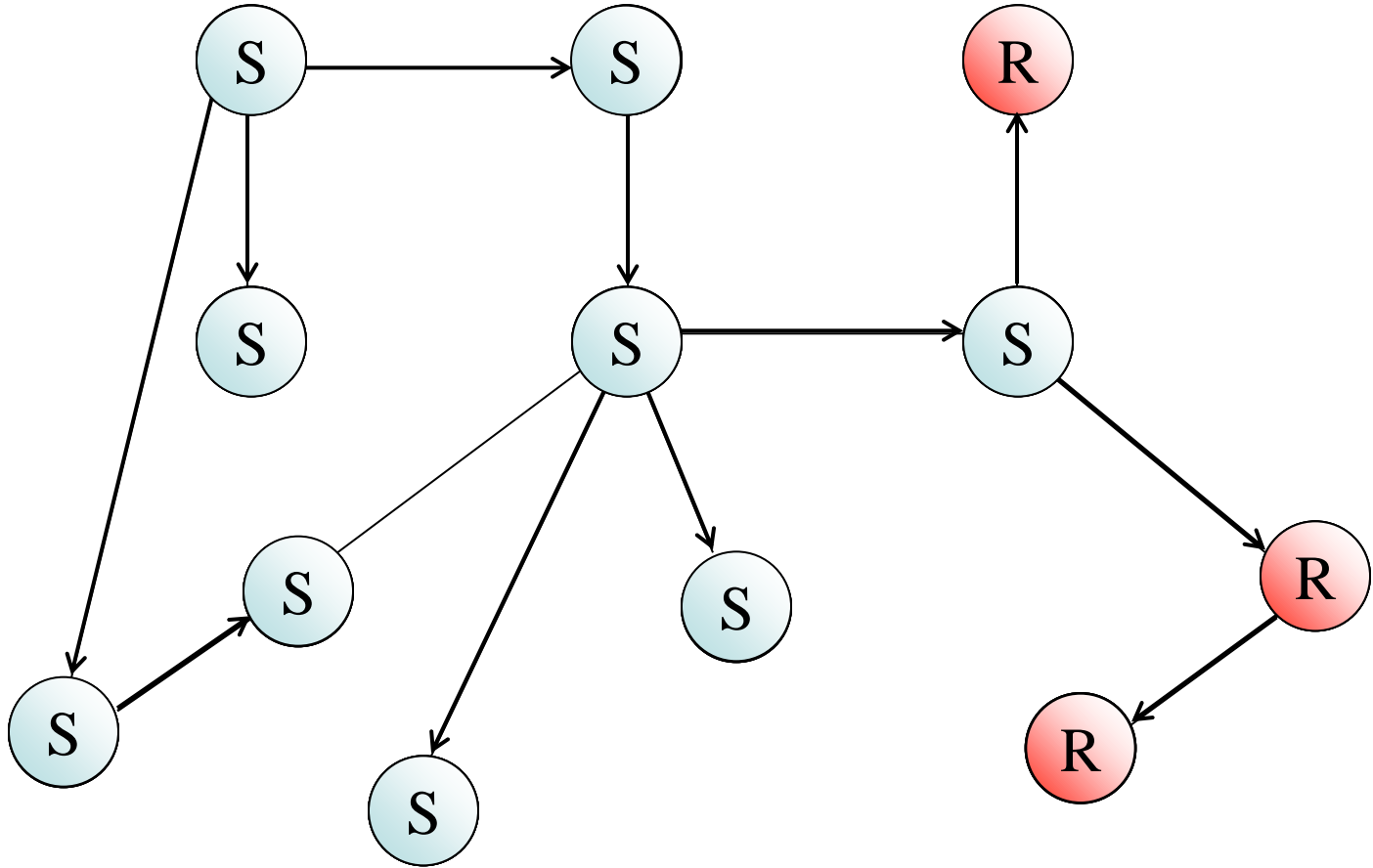


# SI model variants - SIRS

- Another simple variant is the SIRS model.
- Every node is susceptible, then it becomes infected and after that it recovers (or it is removed from the network). At the end of this process it can become susceptible (to infection) again.
- A susceptibility rate can be applied.

CASE 2

# SIRS



**S** Susceptible

**I** Infected

**R** Recovered (or Removed)

# SI model variants - SEIS

- Another variant is the SEIS model.
- Every node is susceptible (S), then it is exposed (E) to an infection for some time and after that it can become infected (I) (or not). At the end of this process it can become susceptible (S) (to infection) again.
- An exposure time can be applied.

# SI model variants - other

- Combinations of these variants also exist:
- SEIR
- SEIRS
- SIS – BD (where BD are the birth and death rates)
- SIRS – BD
- etc

# Some math about it

- We can formulate a simple generic model with two, three or more compartments.
- i.e. in the SIR model suppose we have a population of  $N$  nodes, if  $S_{(t)}$  is the susceptible population,  $I_{(t)}$  is the infected population and  $R_{(t)}$  is the recovered (or removed) population, then:

$$N = S_{(t)} + I_{(t)} + R_{(t)}$$

# Some math about it

- This equation is time dependent.
- By applying a rate  $\beta$  for disease (virus) contraction, a rate  $\gamma$  for the recovery, and by doing a simple differentiation:
- $dS/dt = -\beta SI$
- $dI/dt = \beta SI - \gamma I$
- $dR/dt = \gamma I$



# Some math about it

- But networks are evolving. Birth and death is a natural process for nodes (new ones added, old ones removed).
- Thus, if we assume  $\mu$  the birth and death rate (assume equal rates) we have:
- $dS/dt = -\beta SI + \mu(N - S)$
- $dI/dt = \beta SI - \gamma I - \mu I$
- $dR/dt = \gamma I - \mu R$

# Some math about it

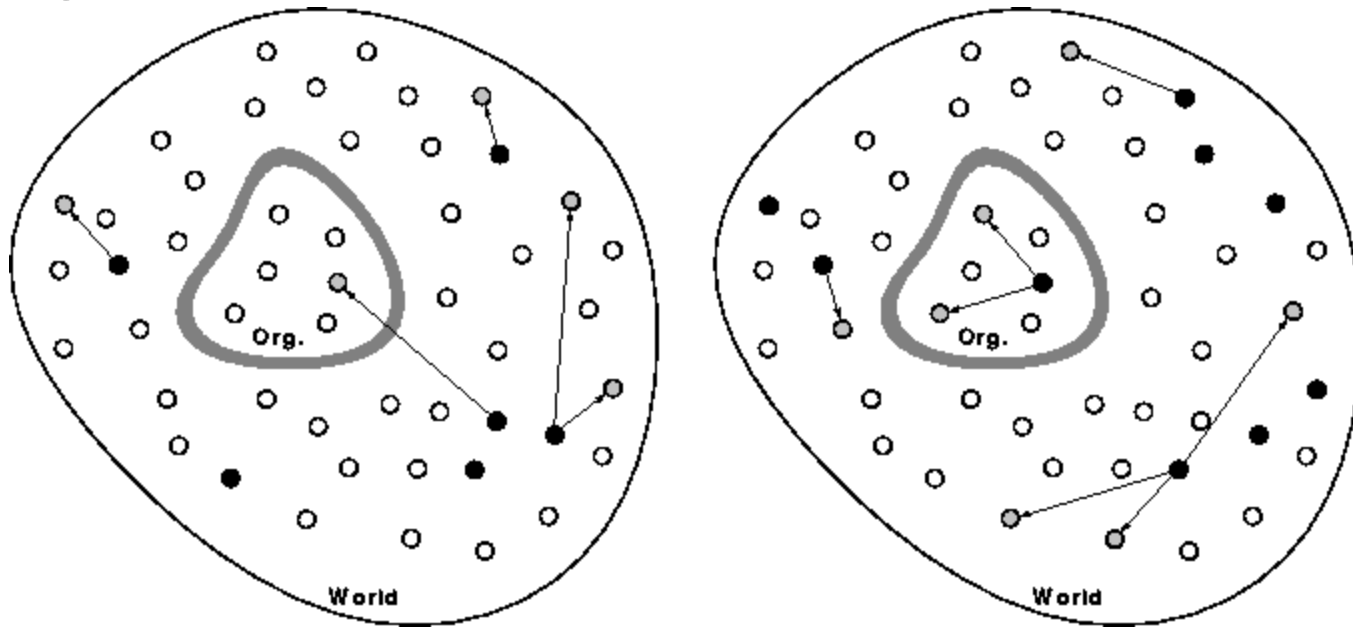
- Similar equations can be produced for every variant of the model.
- Example: the SEIS model, where  $E_{(t)}$  is the additional compartment for the exposure period and  $\varepsilon$  is the rate associated with it:
- $dS/dt = B - \beta SI - \mu S + \gamma I$
- $dE/dt = \beta SI - (\varepsilon + \mu)E$
- $dI/dt = \varepsilon E - (\gamma + \mu)I$

# Differences with humans

- Immunity in computers can be gained by updates. You don't have to get a virus to become immune.
- Not all computers are susceptible to all kinds of viruses. Different OSes have different problems (Linux, Windows, etc).
- Recovery happens in very different timeframes (whenever an admin sees the problem).

# Differences with humans

- Some parts of the population may be secure (not easily penetrated). i.e closed organizations.



# Example time

[www.gleamviz.org](http://www.gleamviz.org)

# Email based worms

- Zou et al. describe a model of e-mail worm propagation, where the e-mail service is an undirected graph between people (i.e. if user A has user B's e-mail address in the address book, B has probably A's address too).
- In order to build a graph, each node degree is distributed on a power-law probability function.



# Email based worms

- It is unclear if this distribution reflects also the true distribution of contacts (i.e. not considering e-mail lists) among Internet users.
- Furthermore, the model assumes that each user “opens” an incoming virus attachment with a fixed probability  $P_i$ , a function of the user but constant in time.

# Email based worms

- This does not describe very well the typical behavior of users. Indeed, most experienced users avoid virus attachments altogether, while unexperienced users open them every time, thus making this approximation misleading.

# Targeted attacks

- In communications networks it is not uncommon to have a ‘targeted attack’.
- In such a case a specific computer, a range of computers, or a range of users having a common property are targeted.
- The end goal is to either disable the target(s) or steal information from them.

# Information theft

- In case the goal is to steal information a virus or trojan horse is used and (from a physical point of view) the system can be treated as before (a simple spreading model).
- A difference can be that the virus is selective on the links it will follow. It will for example attack nodes that belong to a certain country.

# Information theft

- Such attacks have been happening for a long time in many levels:
- Countries use cyberattacks to gather information on various issues. Examples of this are the attacks of Iran to Israel and vice versa, as well as the recent attacks of Chinese hackers in many US based institutional services and intelligence agencies.

# Information theft

- Companies use such attacks to gain access to confidential information of other companies (industrial espionage).
- Individuals (hackers) target groups of people in order to retrieve personal information (i.e. bank accounts, credit card information, personal data, etc.).

# Disabling a target - DoS

- In case the goal is to disable the target what usually happens is a DoS – Denial of Service attack.
- This is an attempt to make a machine or network resource unavailable to its intended users.

# Disabling a target - DoS

- Although the means, motives, and targets of a DoS attack may vary, it is generally an effort to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet.
- The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management.



# Disabling a target - DoS

- DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root nameservers.
- This technique has now seen extensive use in certain games, used by server owners, or disgruntled competitors on games such as Minecraft.

# Disabling a target - DDoS

- Modern computers that are usually targeted have great resources at their disposal.
- This means that resources are also needed to attack them.
- That is why we have distributed denial-of-service attacks (DDoS attack)!
- Many computers attack just one!

# Disabling a target - DDoS

- Denial of service took on new visibility and importance in February 2000, when denial of service attacks took down the websites of Yahoo, Buy.com, eBay, CNN, Amazon.com, ZDNet.com, E\*Trade, and Excite.



# Disabling a target - DDoS

- These attacks were so large that they suggested multiple origin points harnessed into a DDoS attack.
- The attacks were ultimately traced to Michael Calce, aka "Mafiaboy", a 15 year-old from Montreal. He was identified because he bragged about the attacks in an IRC channel. He served eight months of "open custody" for his crimes.

# Disabling a target - DDoS

- The size of the largest reported DDoS attacks has increased steadily, from 400 megabits (Mbps\*) per second in 2002 to 49 gigabits per second (Gbps\*) in 2009.
- For comparison, Harvard College connected its tens of thousands of users on its network to the Internet through a 2 Gbps link in 2009.

# Disabling a target - DDoS

- On an average day in 2010 there were about 1300 DDoS attacks!
- About 93% of these attacks were stopped within an hour.

# Disabling a target - DDoS

- On an average day in 2010 there were about 1300 DDoS attacks!
- About 93% of these attacks were stopped within an hour. 18% within the first 15 minutes!

# Network robustness

- In graph theory, the "robustness" of a graph, or class of graphs, measures its resilience (in terms of connectivity) to the removal of edges or vertices.
- Robustness can be formalized in a many ways, depending on how connectivity is measured, whether nodes or links are removed, and how they are chosen to be removed.



# Network robustness

- For the purposes of measuring the robustness of a network, the size of the graph's largest component is frequently used to measure connectivity.
- Of particular interest is whether the graph has a giant component (this is a component that contains a majority of the vertices of the graph).

# Network robustness

- The giant component is also sometimes referred to as a "spanning cluster", which is related but not identical to the concept of the same name in percolation theory.
- The robustness of a graph greatly depends on the graph's structure. In particular, the vulnerability of a network to random failures has been connected to its degree distribution.

# Random node removal

- A random removal of nodes in a network does not lead to a collapse of the giant component.
- Instead the giant component reduces steadily in size up to almost 100% removal.

# Targeted node removal

- Targeted attacks are different!
- Targeted removal of nodes can lead to a breakdown much faster.
- Networks under a targeted attack can break down (have their giant component decrease very fast in size) with a removal of only about 7% of the highest degree nodes.

# Example of research in IT

- The Byzantine Empire had frequent problems of internal treachery.
- High level officials have not been as loyal as in other empires!
- In many cases they disobeyed direct orders.
- The problem is similar with that of computers being compromised.

# Byzantine Generals Problem - BGP

- A compromised computer is not trustworthy and can relay information based on what the one controlling it wants.
- A quick glance of this problem can explain why this is related to IT.
- In the generals problem a general can be thought of as a computer node.

# BGP Definition

- Each division of Byzantine army is directed by its own general.
- There are  $n$  Generals, some of which are traitors.
- All armies are camped outside an enemy castle, observing the enemy.
- They communicate with each other by messengers.

# BGP Definition

- Requirements:
- G1: All loyal generals decide upon the same plan of action
- G2: A small number of traitors cannot cause the loyal generals to adopt a bad plan
- Note: We do not have to identify the traitors.



# BGP - Naive Solution

- $i$ th general sends  $v(i)$  to all other generals
- To deal with two requirements:
  - All generals combine their information  $v(1), v(2), \dots, v(n)$  in the same way.
  - Majority ( $v(1), v(2), \dots, v(n)$ ), ignore minority traitors.

# BGP - Naive Solution

- Naive solution does not work:
  - Traitors may send different values to different generals.
  - Loyal generals might get conflicting values from traitors.
- Requirement: Any two loyal generals must use the same value of  $v(i)$  to decide on same plan of action.

# BGP – Reduced problem

- If we restrict ourselves to the problem of one general sending its order to others.
  - A commander must send an order to his  $n-1$  lieutenants.
- Interactive Consistency Conditions:
  - IC1: All loyal lieutenants obey the same order.
  - IC2: If the commanding general is loyal, then every loyal lieutenant obeys the order he sends.

# BGP – Reduced problem

- Note: If General is loyal, IC2  $\Rightarrow$  IC1.
- Relation to the original problem:
- Each general sends his value  $v(i)$  by using the above solution, with other generals acting as lieutenants.

# BGP – 3 general impossibility

- 3 generals, 1 traitor among them.
- Two messages: Attack or Retreat
- L1 sees (A,R). Find the traitor. C or L2?
  - Fig 1: L1 has to attack to satisfy IC2.
  - Fig 2: L1 A, L2 R. IC1 violated.

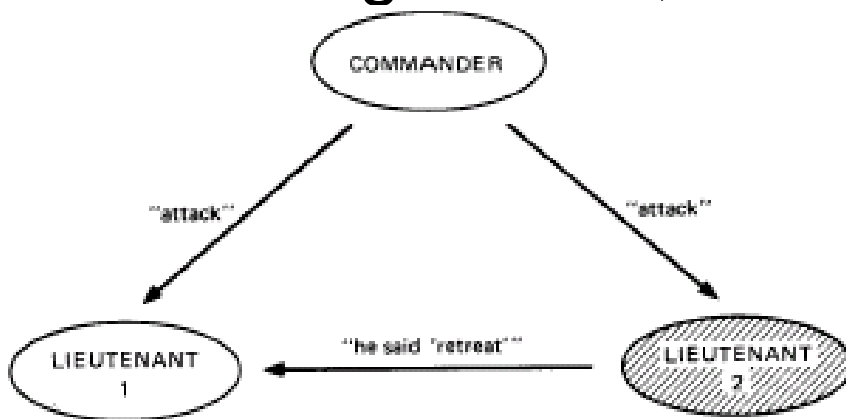


Fig. 1. Lieutenant 2 a traitor.

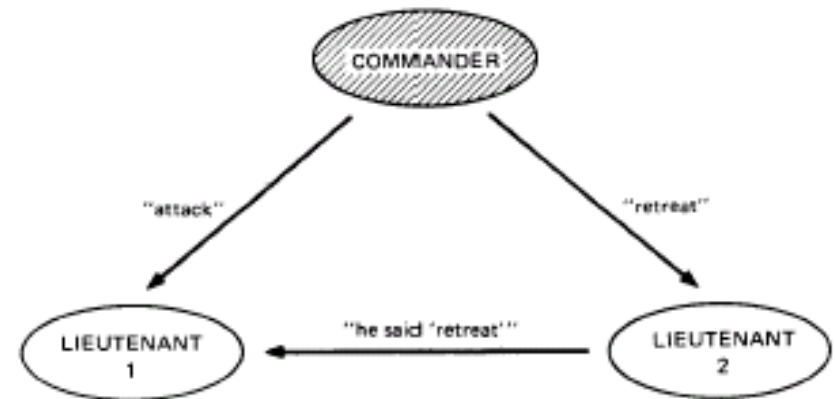


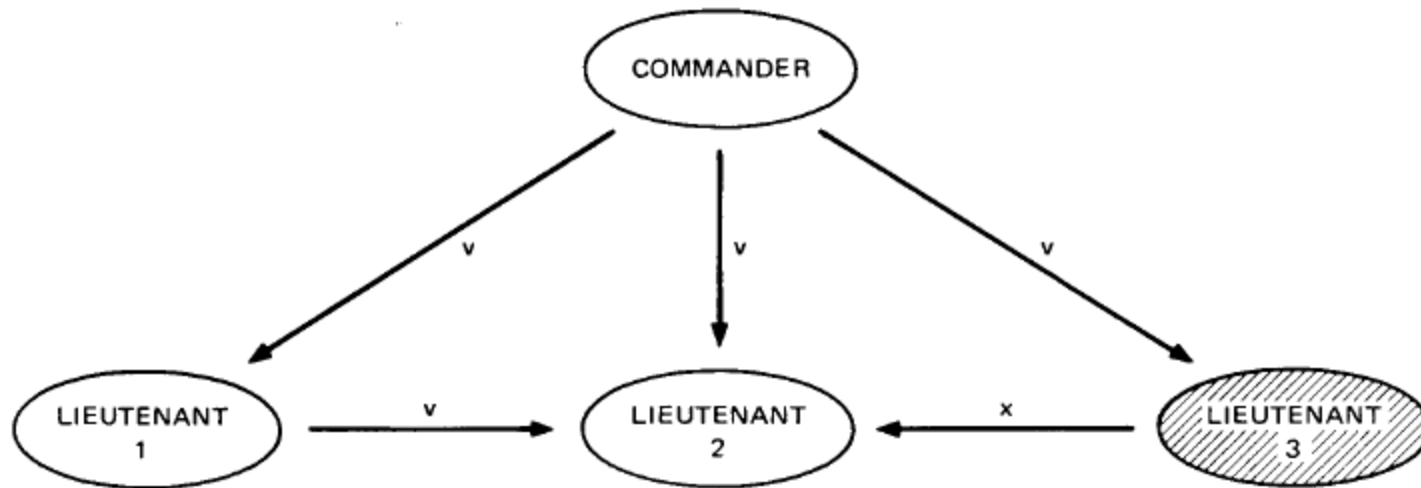
Fig. 2. The commander a traitor.

# BGP – more generals

- In general, no solutions with fewer than  $3m+1$  generals can cope with  $m$  traitors.
- What happens when the generals (along with the lieutenants) are four ( $m=1$ )?
- Restatement:
  - IC1: All loyal lieutenants obey the same command.
  - IC2: If the commander is loyal, then every loyal lieutenant obeys the command he sends.

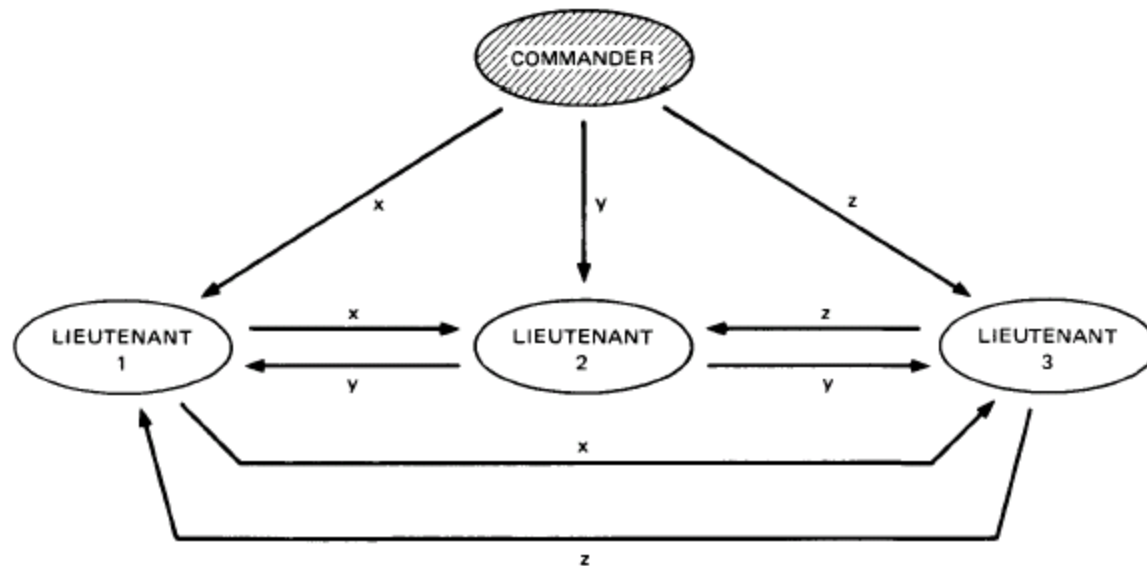
# BGP – more generals

- Algorithm OM(1): L3 is a traitor.
- L1 and L2 both receive  $v, v, x$ . (IC1 is met.)
- IC2 is met because L1 and L2 obeys C.



# BGP – more generals

- Algorithm OM(1): Commander is a traitor.
- All lieutenants receive  $x, y, z$ . (IC1 is met).
- IC2 is irrelevant since commander is a traitor.







# Example No 2

## Motivation

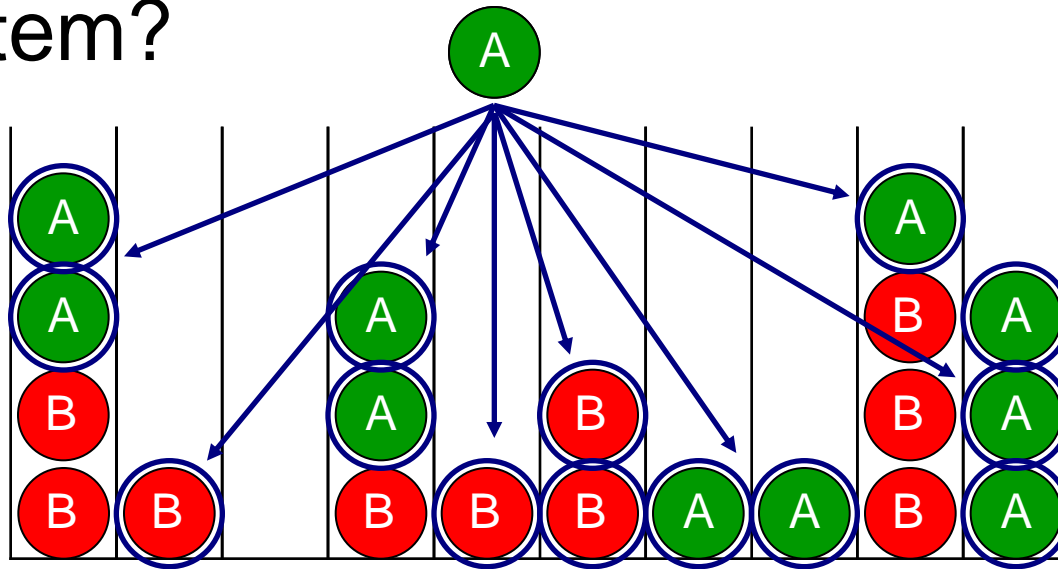
- Many communication networks use random walk to search other computers or spread information.
- Some data packets have higher priority than others.
- How does priority policy affect diffusion in the network?

# Example No 2

- Two species of particles, A and B.
- Symmetric random walk (NN).
- A has high priority, B has low priority.
- B can move freely only if all the A's in its site have already moved.  
- We choose randomly either a
  - a) Site (site protocol) or a
  - b) Particle (particle protocol)at every time step.

# Example – 1D lattice.

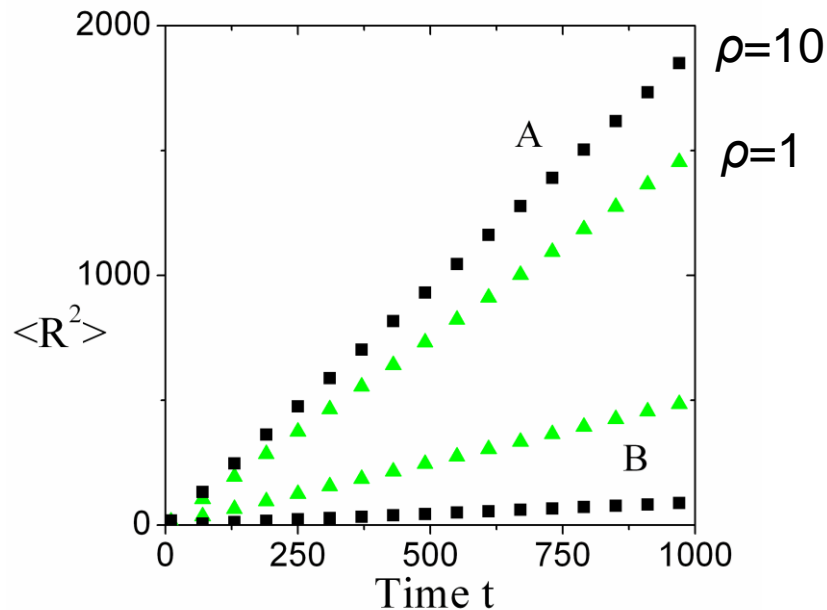
- Which particles are mobile in such a system?



- All A
- B in sites 2, 5 and 6.

# PDM – Lattices example

- Both types diffuse normally:  $\langle R^2 \rangle = Dt$ .



- But how is time shared between A and B?

# Priority diffusion – site protocol

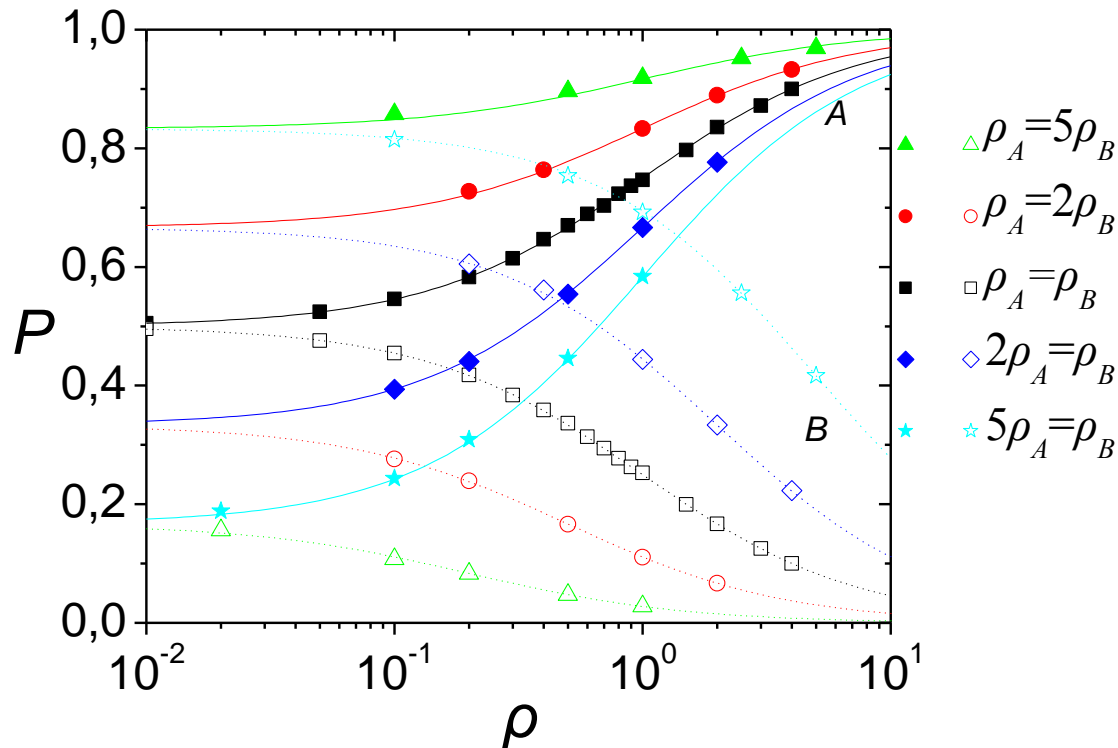
- Densities are  $\rho_A$  and  $\rho_B$ .
- Sites with any A:  $\frac{\rho_A}{1 + \rho_A}$
- Sites with no A *and* no B:  $\frac{1}{1 + \rho_A + \rho_B}$
- Time A and B are moving:

$$P_A = \frac{\rho_A(1 + \rho_A + \rho_B)}{(1 + \rho_A)(\rho_A + \rho_B)}; \quad P_B = \frac{\rho_B}{(1 + \rho_A)(\rho_A + \rho_B)}$$

# PDM – site protocol

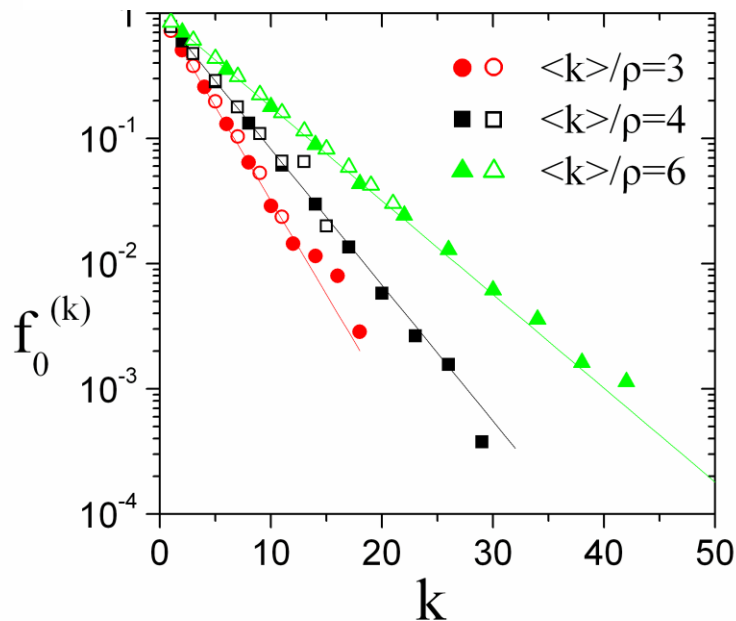
- Time A and B are moving:

$$P_A = \frac{\rho_A(1 + \rho_A + \rho_B)}{(1 + \rho_A)(\rho_A + \rho_B)}; \quad P_B = \frac{\rho_B}{(1 + \rho_A)(\rho_A + \rho_B)}$$



# PDM in Complex Networks

- What about a network with PDM?
- Consider one type only.
- Fraction of empty nodes:  $f_0^{(data,k)} = \exp(-\rho k / \langle k \rangle)$



SF  
networks

# PDM in networks – discussion

- A's move freely, and tend to aggregate at the hubs (form large queues).
- B's at the hubs have very low probability to escape (the queue of A has to be zeroed).
- In lattices, hubs do not exist so B's can move.
- In networks hubs exist. B's also tend to aggregate at these hubs and therefore become practically immobile there.



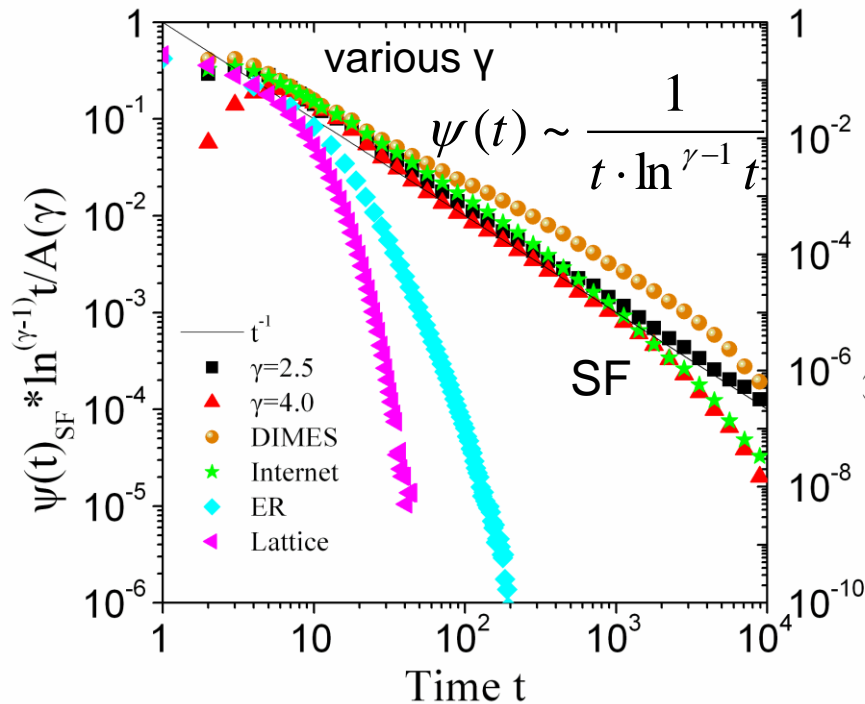
# PDM in networks – results

- B can move if a node is empty of A, which happens with probability  $p = \exp(-\rho_A k / \langle k \rangle)$
- In an average sense, in every time step a node can become empty with probability  $p$ .
- This leads to exponential distribution of B's waiting times:

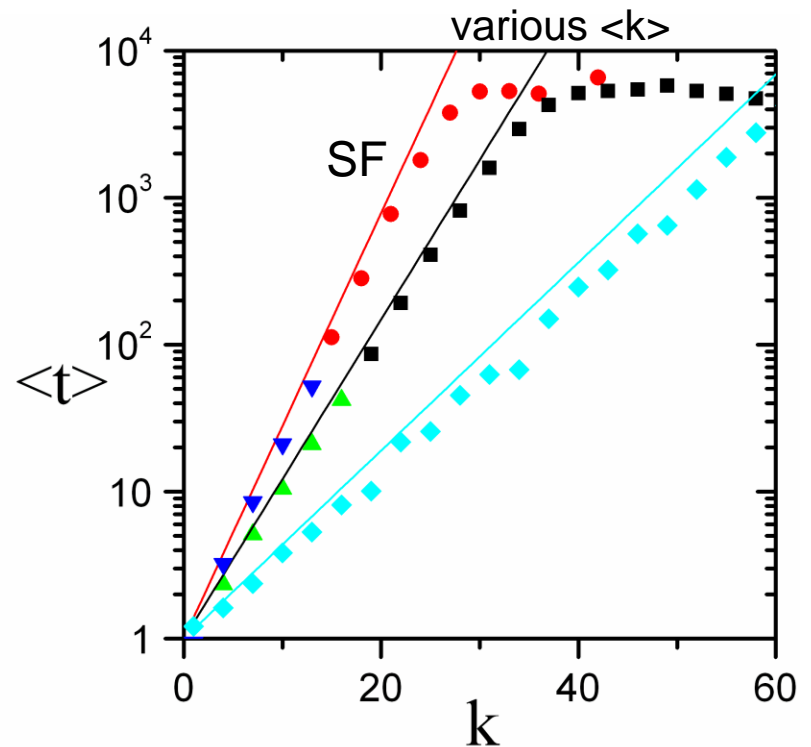
$$\psi(t | k) = e^{-t/\tau} / \tau, \text{ where } \tau \equiv \langle t(k) \rangle = \exp(\rho_A k / \langle k \rangle)$$

- For networks with  $P(k) \sim k^{-\gamma}$ ,  $\psi(t) \sim \frac{1}{t \cdot \ln^{\gamma-1} t}$

# PDM in networks - simulations

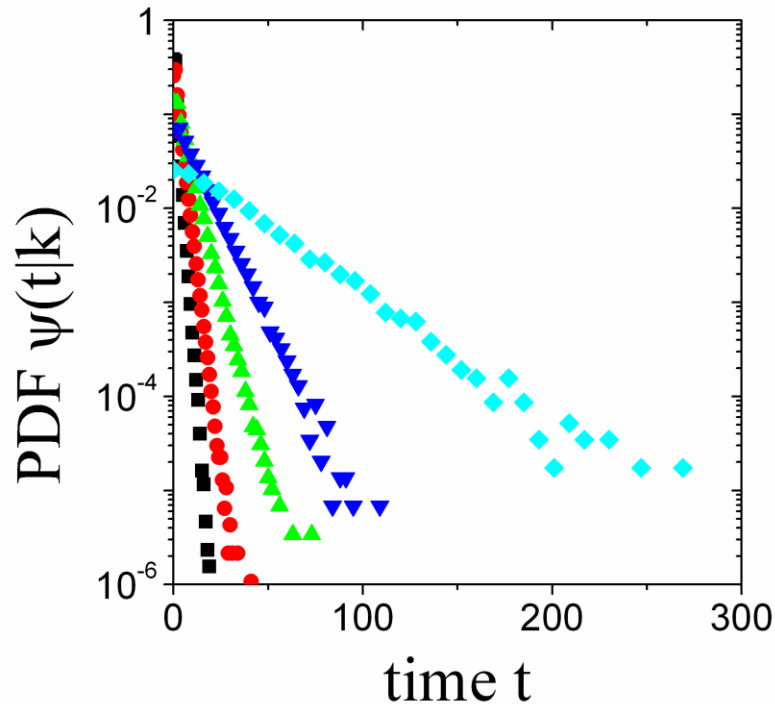


Distribution of waiting times (for B):  
narrow for lattices, broad for SF.

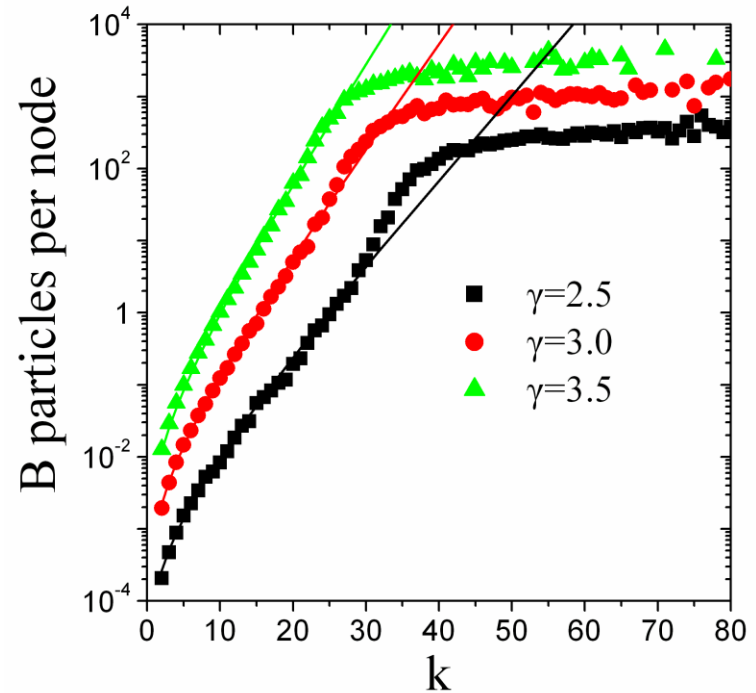


Waiting time for the B's grows  
exponentially with the degree

# PDM in networks - simulations



Exponential distribution of waiting times (of B) for different degrees.



Number of B vs. node degree. Theory (solid lines):  
 $k \exp(\rho_A k / \langle k \rangle)$

# Main results of the protocol before

- B particles are effectively trapped in a hub (waiting times larger by several orders of magnitude) → mixing of B
- A hub accepts exponentially higher volumes of particles than low degree nodes (both A and B, but only B get stuck there).

# Why care any more about it?

- In real world systems it is not always desirable to get large queues of B
- How can one reduce the delay of B to an acceptable value?
- How would this affect the diffusion mechanism within the system?
- Strategies to avoid halting of B are needed!

# Strategies to avoid mirroring of B

moveA subprotocol.

## Description

If a B is chosen to move, the movement goes to coexisting A, if any exist.

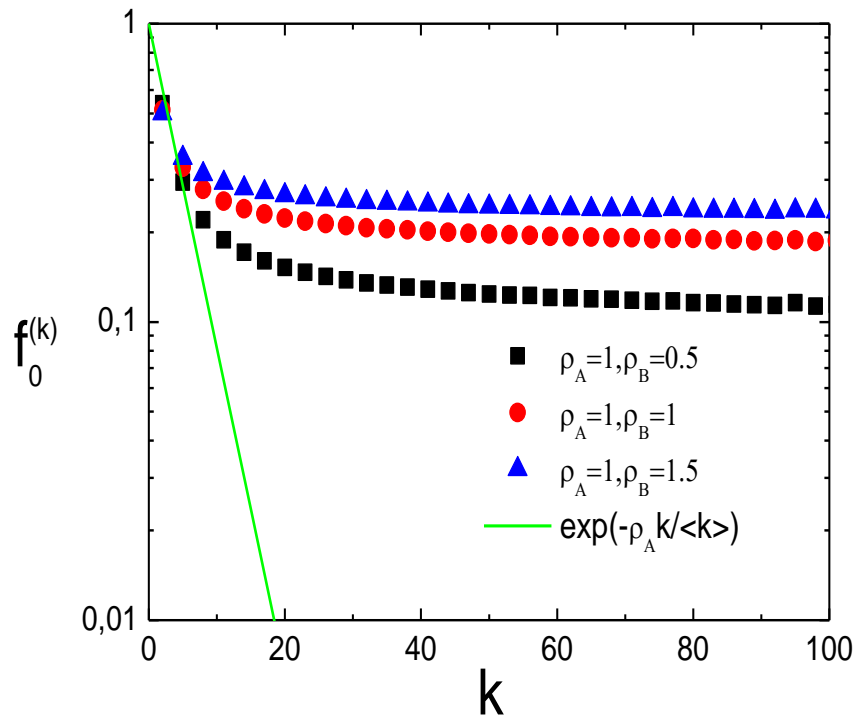
## Result

A's are expelled from hubs and B can also leave the hub.

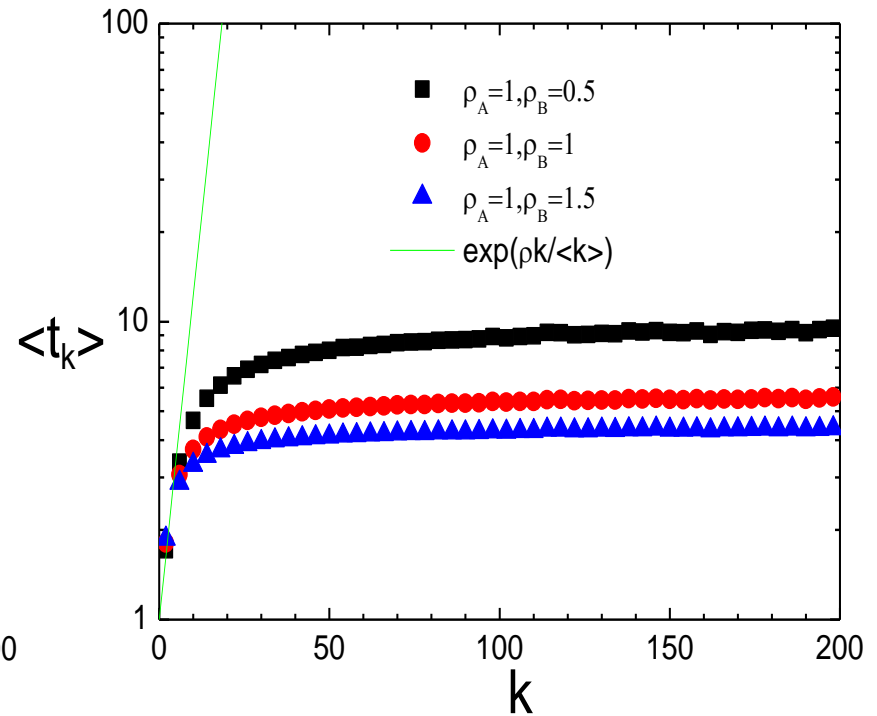
The average waiting time is almost independent of the degree (for high degree).

# Strategies to avoid miring of B

moveA subprotocol.



Nodes free of A vs the degree  $k$ .



Average waiting time at degree  $k$ .

# Strategies to avoid mirroring of B

Soft priorities.

## Description

Apply a probability for the B (even a small one) to move when an A is coexisting.

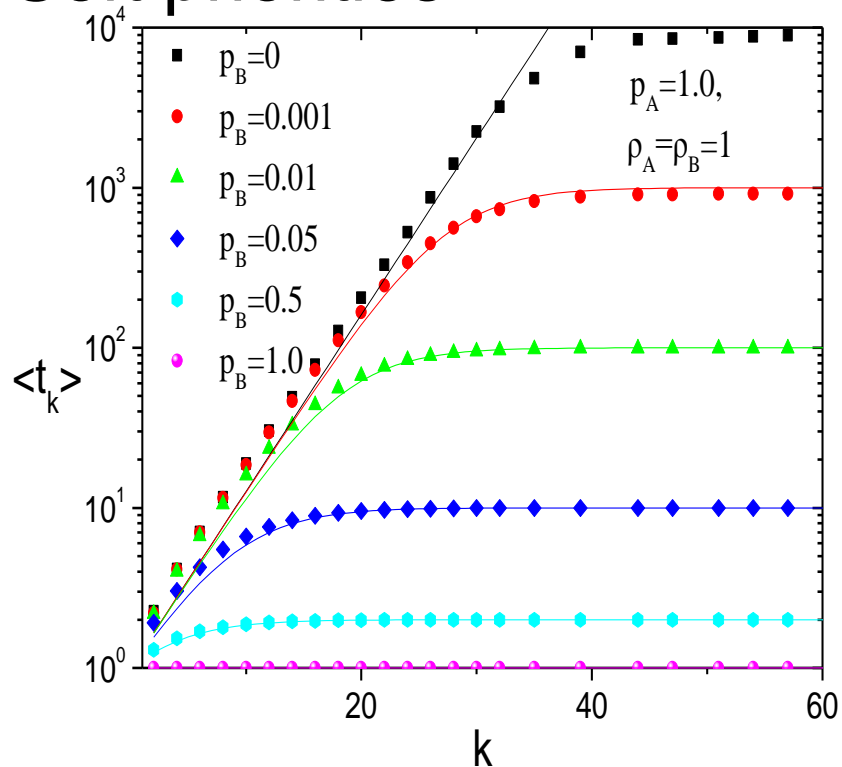
## Result

This makes the average waiting time of B in hubs finite inversely dependent on the probability applied.

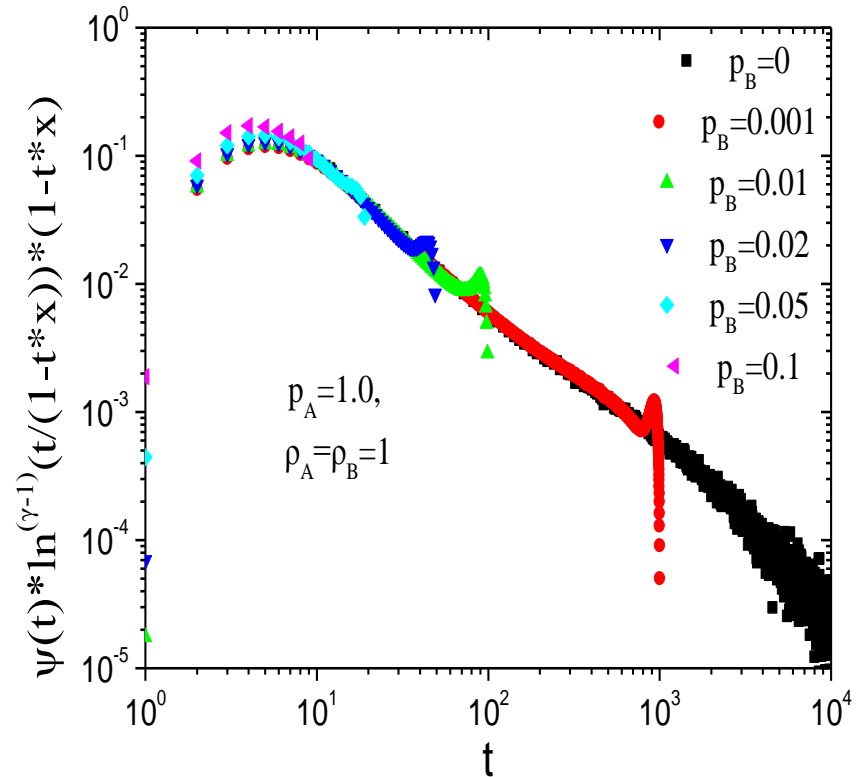


# Strategies to avoid mirroring of B

## Soft priorities.



Average waiting time at degree k.



Waiting time distribution.

# Strategies to avoid miring of B

Avoid hubs.

## Description

Particles tend to avoid moving through hubs whenever possible.

$$P_{\text{jump to } i\text{th neighbor}} = \frac{k_i^{\alpha}}{\sum_{j \text{ neighbor}} k_j^{\alpha}}$$

## Result

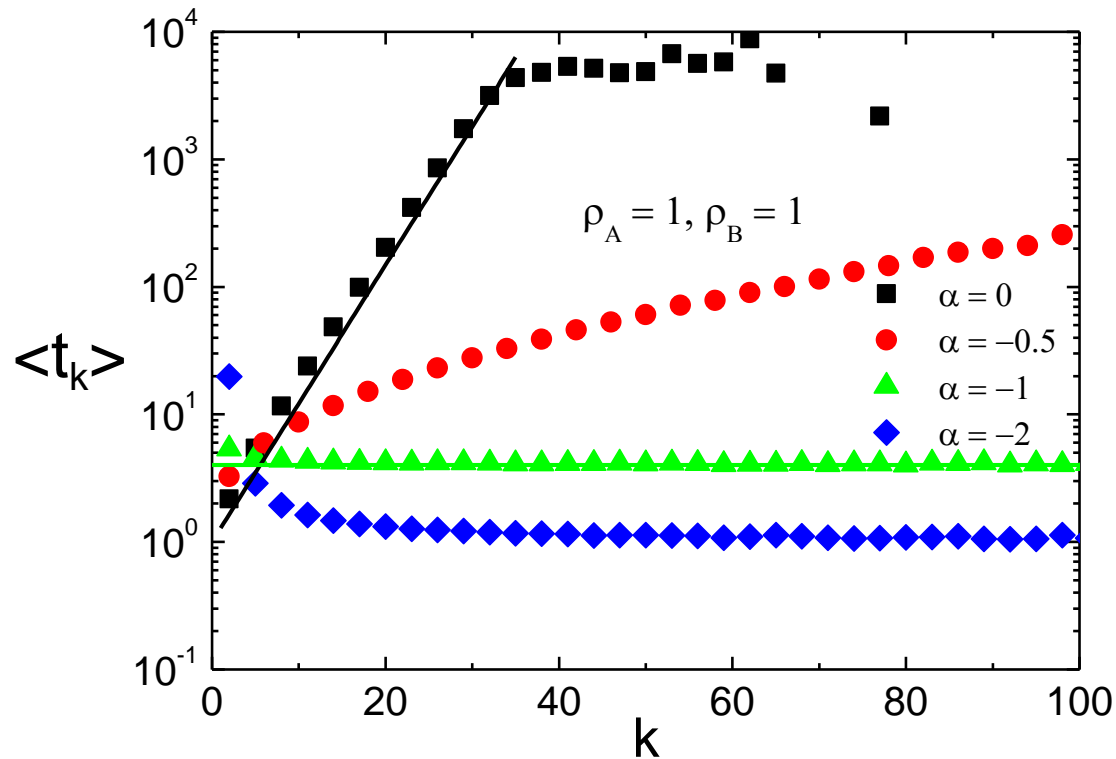
If  $\alpha = -1$ , revert to the lattice case.

If  $\alpha > -1$ , B are still trapped.

if  $\alpha < -1$ , B move through small degree nodes

# Strategies to avoid miring of B

Avoid hubs.



Average waiting time at degree  $k$ .

# Strategies to avoid mirroring of B

Limited node capacity.

## Description

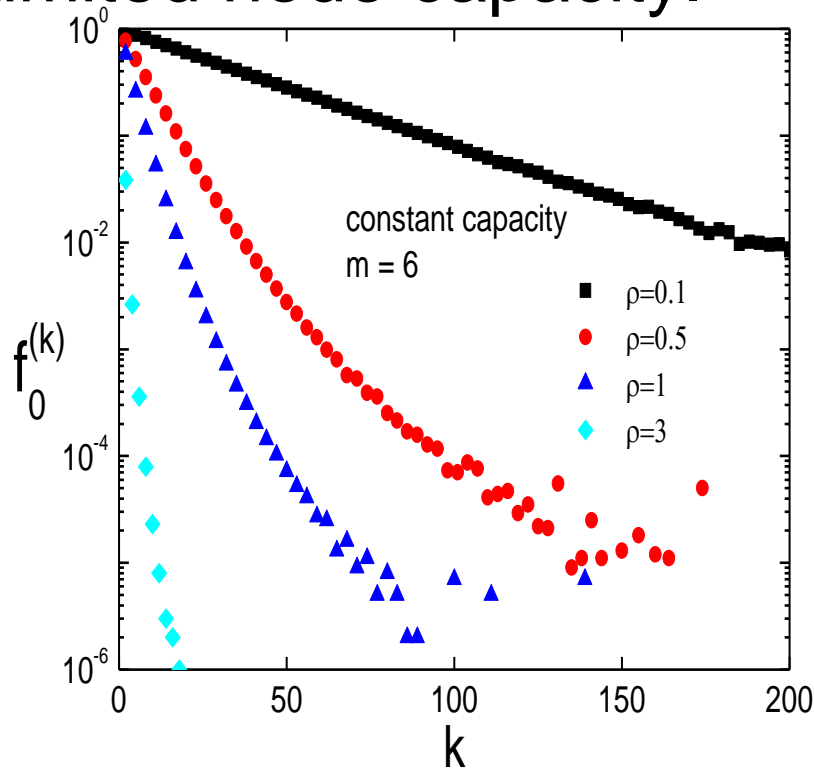
Apply a limit in the capacity of each node to accept particles (depending on the degree).

## Result

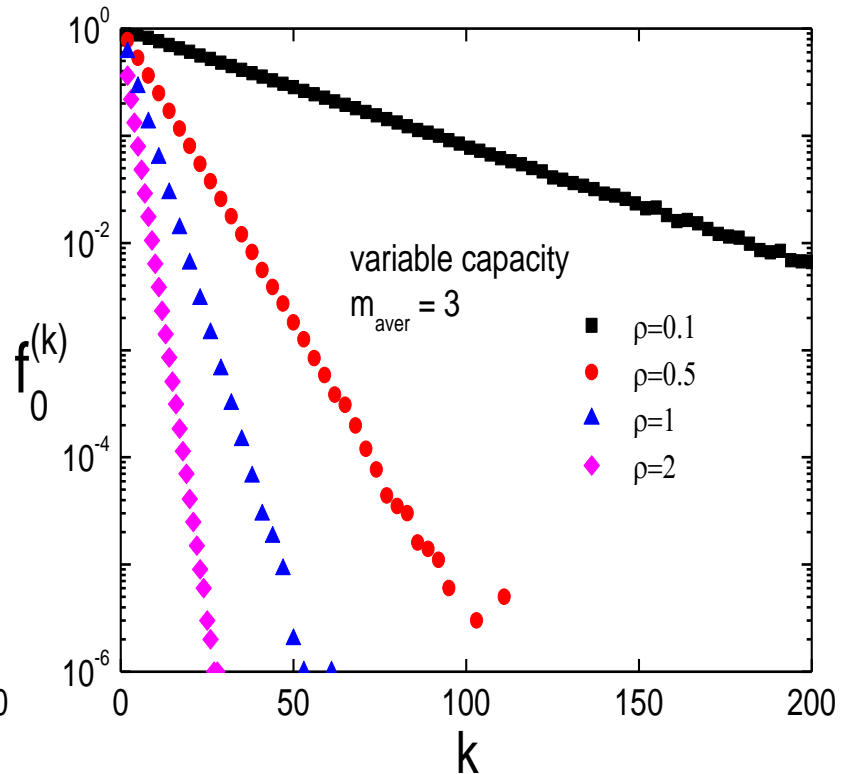
A move as usual, occupy hubs and block any incoming B, therefore Bs go through lower degree nodes and no halting is present.

# Strategies to avoid miring of B

Limited node capacity.



Nodes free of A vs the degree  $k$ .



Nodes free of A vs the degree  $k$ .

# Summary

- The probability for B to be in a node empty of A decreases exponentially with  $k$ , and leads to halting.
- Several strategies can be applied to avoid the aggregation of the particles at the hubs and allow for improved mobility.
- When priority constraints exist, network structure and protocols should be designed with care.