IEIIT-CNR

# The PageRank Computation in Google, Randomized Algorithms and Consensus of Multi-Agent Systems

Roberto Tempo

IEIIT-CNR

Politecnico di Torino

tempo@polito.it

IEIIT-CNR

# This Lecture

- The objective of this lecture is to study connections between three apparently unrelated topics

- Search engines (PageRank computation in Google)
- Randomized algorithms (Las Vegas type)
- Consensus of multi-agent systems and sensor networks

IEIIT-CNR

# This Lecture

- The objective of this lecture is to study connections between three apparently unrelated topics

- Search engines (PageRank computation in Google)
- Randomized algorithms (Las Vegas type)
- Consensus of multi-agent systems and sensor networks

The PageRank problem is useful for developing novel ideas for systems and control

# This Lecture

- The objective of this lecture is to study connections between three apparently unrelated topics

- Search engines (PageRank computation in Google)
- Randomized algorithms (Las Vegas type)
- Consensus of multi-agent systems and sensor networks

- Technical tool: theory of positive matrices

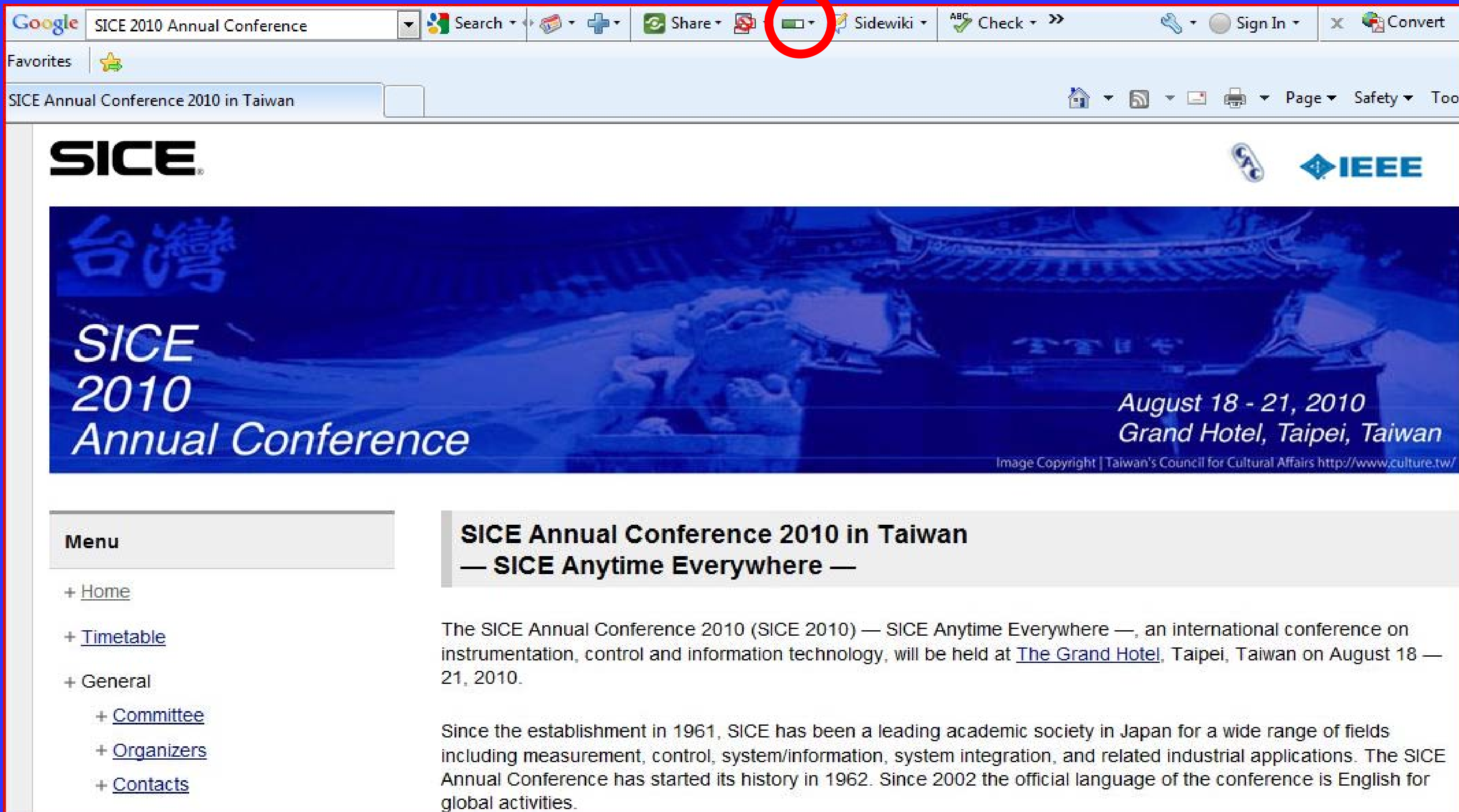
# The PageRank Problem in Google

IEIIT-CNR

# SICE 2010 Annual Conference

IEIIT-CNR

# SICE 2010 Annual Conference



# PageRank for SICE Annual Conference

IEIIT-CNR


*"PageRank is Google's view of the importance of this page (7/10)"*

PageRank is a numerical value in the interval [0,1] which indicates the importance of the page you are visiting

IEIIT-CNR

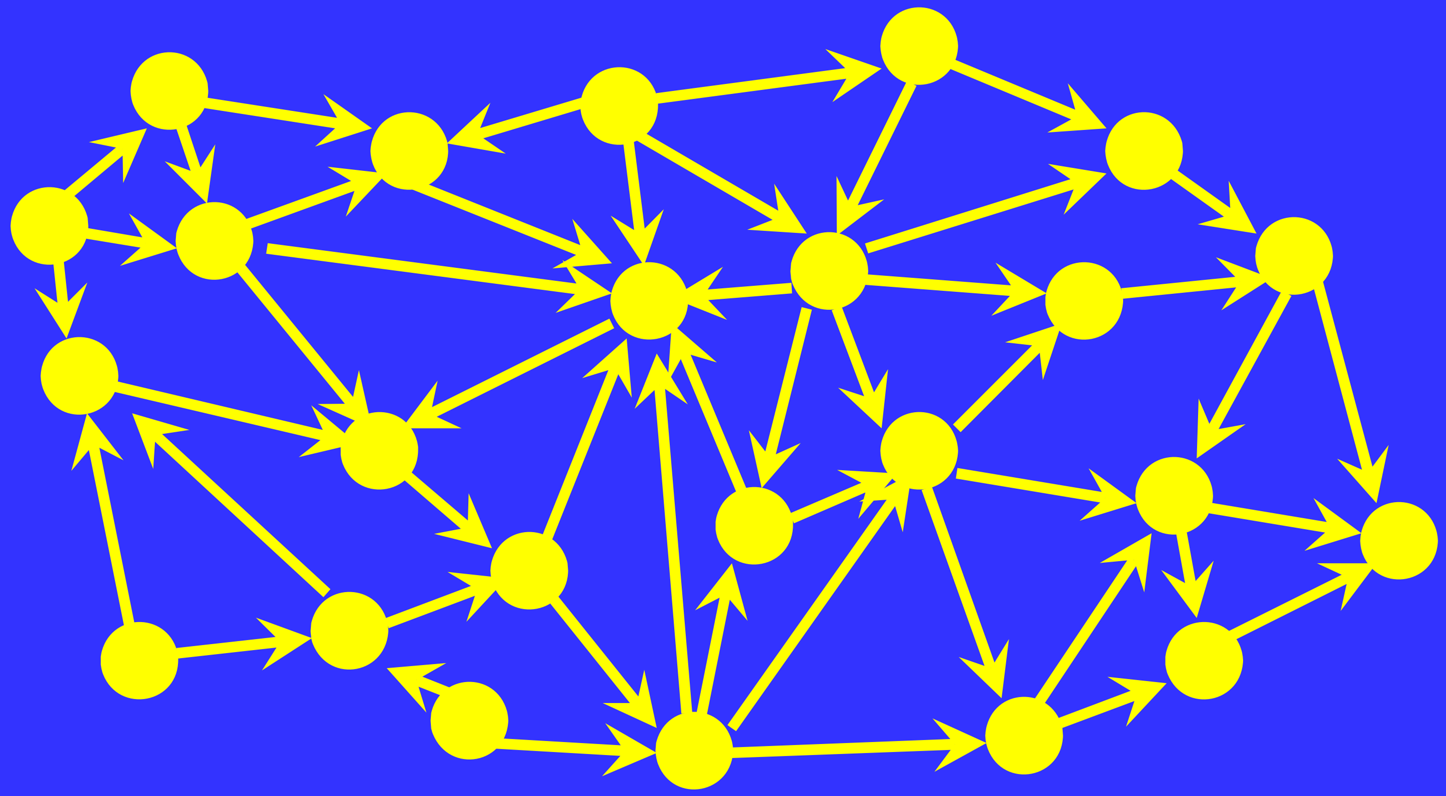
# Random Surfer Model

- Web surfer moves along randomly following the hyperlink structure
- When arriving at a page with several outgoing links, one is chosen at random, then the random surfer moves to a new page, and so on…

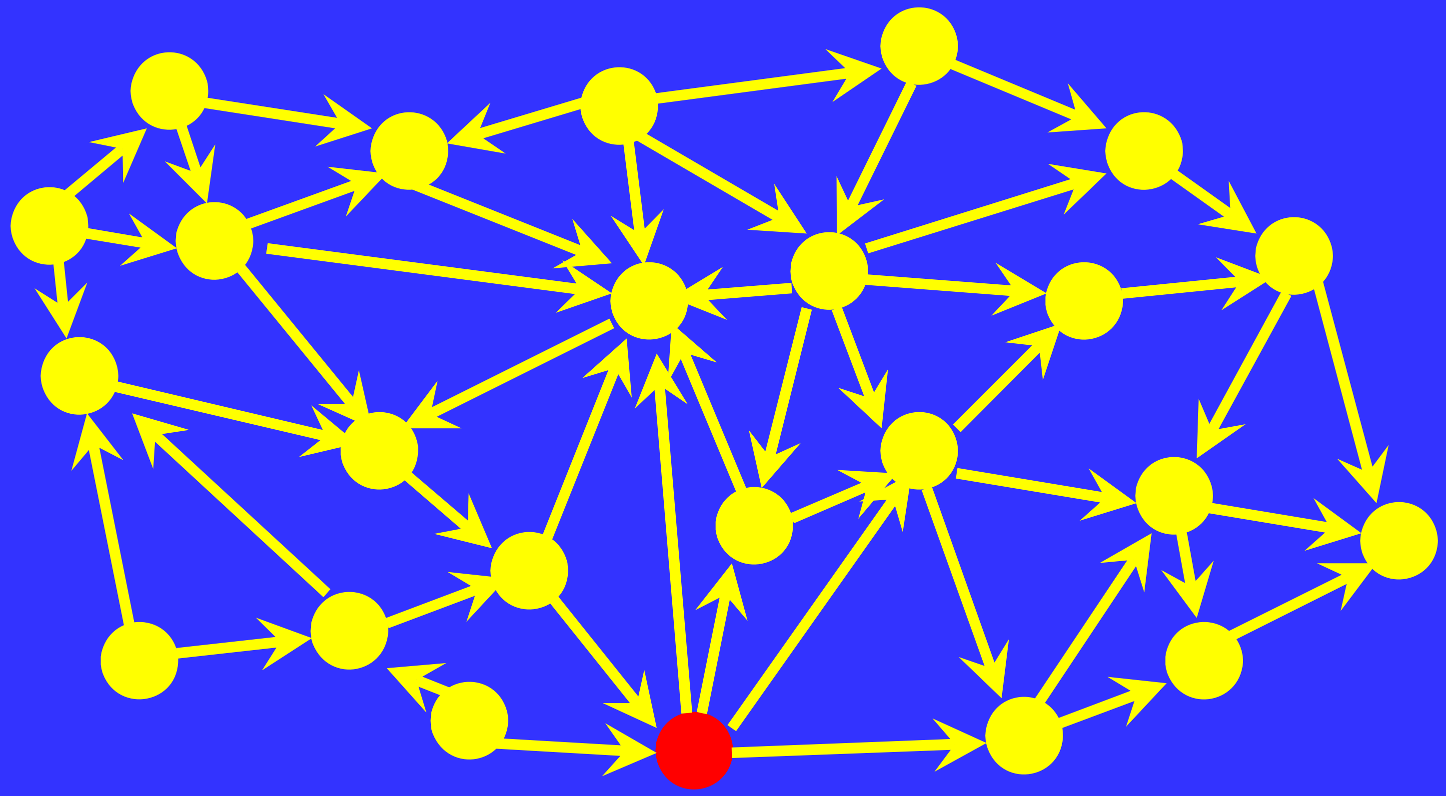
IEIIT-CNR

# Random Surfer Model

- Web representation with incoming and outgoing links



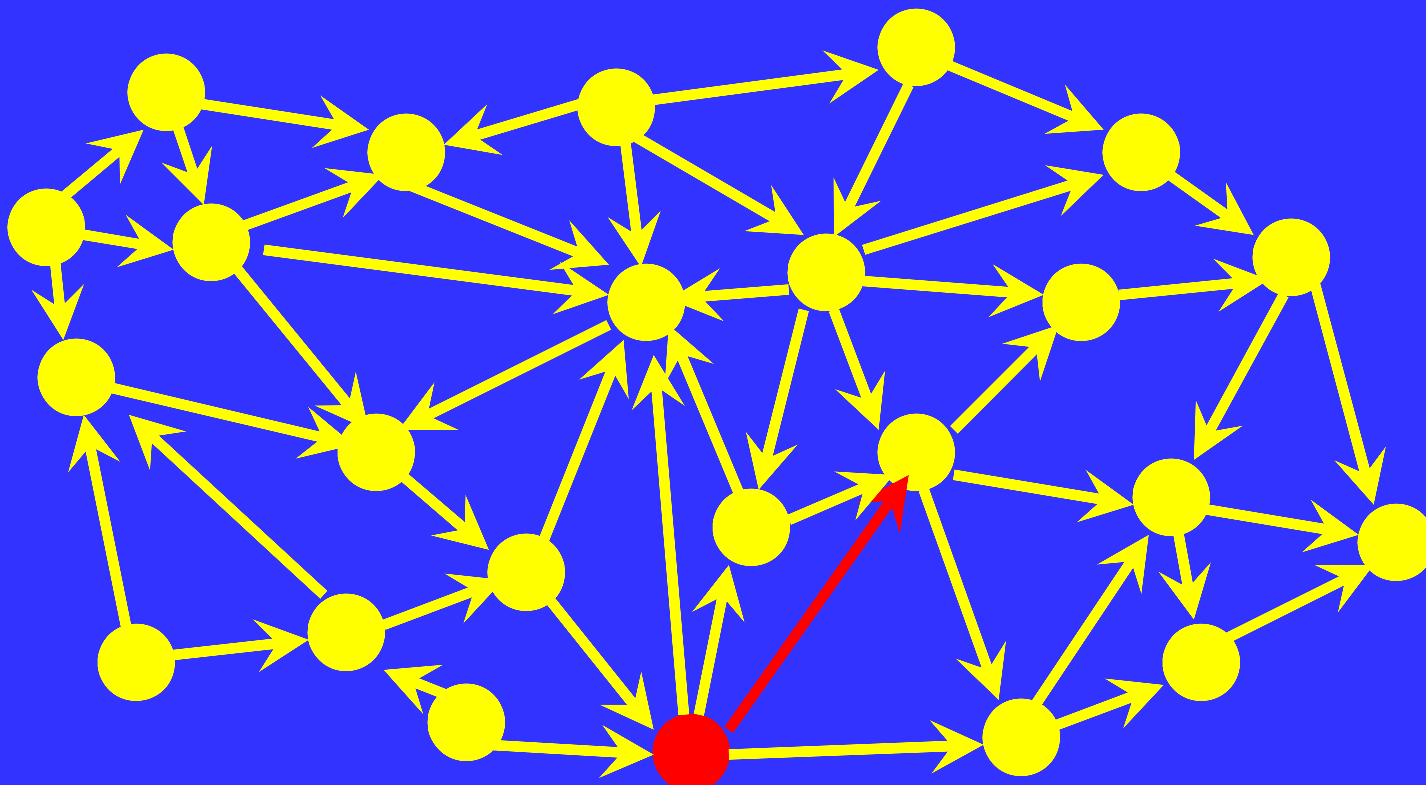
# Random Surfer Model



IEIIT-CNR

# Random Surfer Model

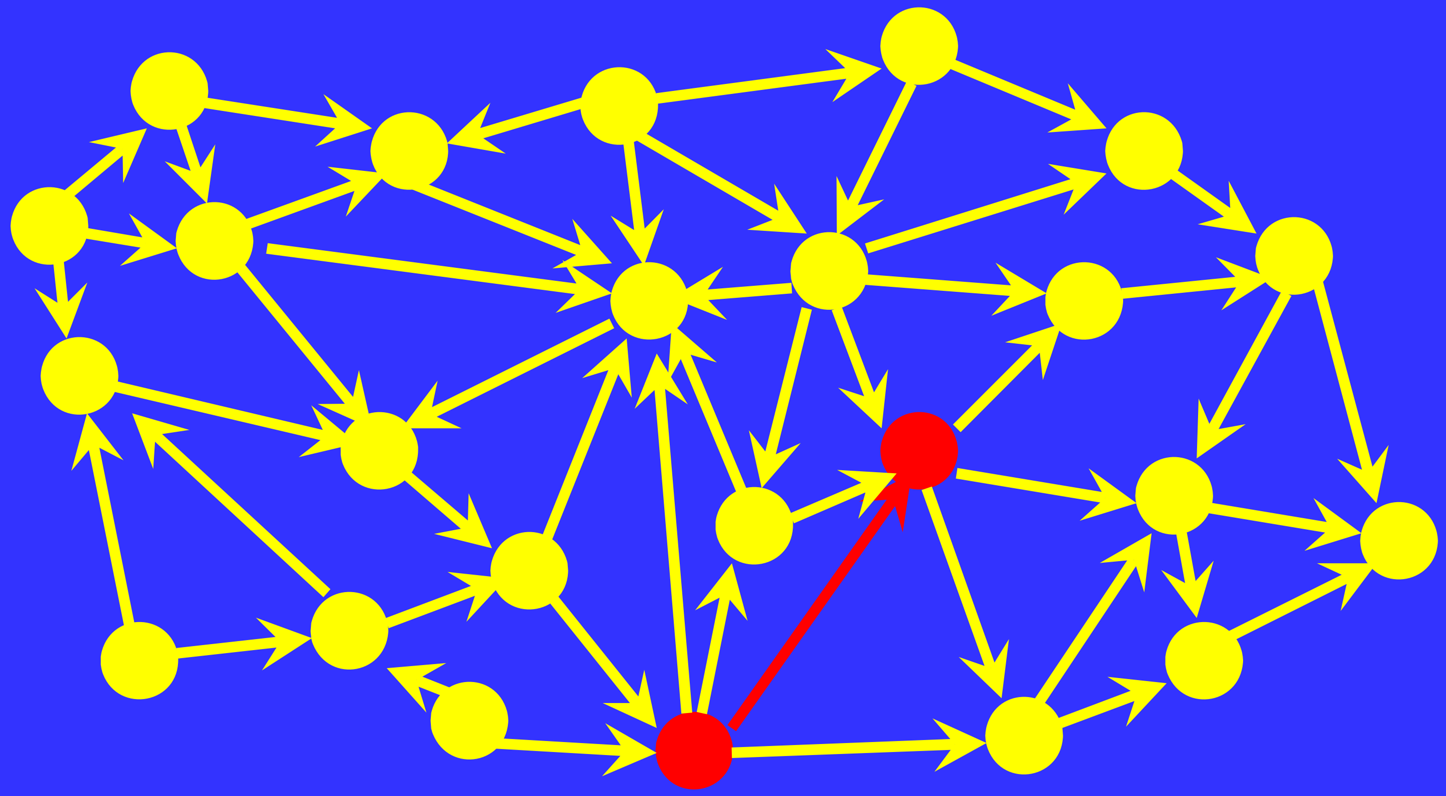
- Pick an outgoing link at random



IEIIT-CNR

# Random Surfer Model

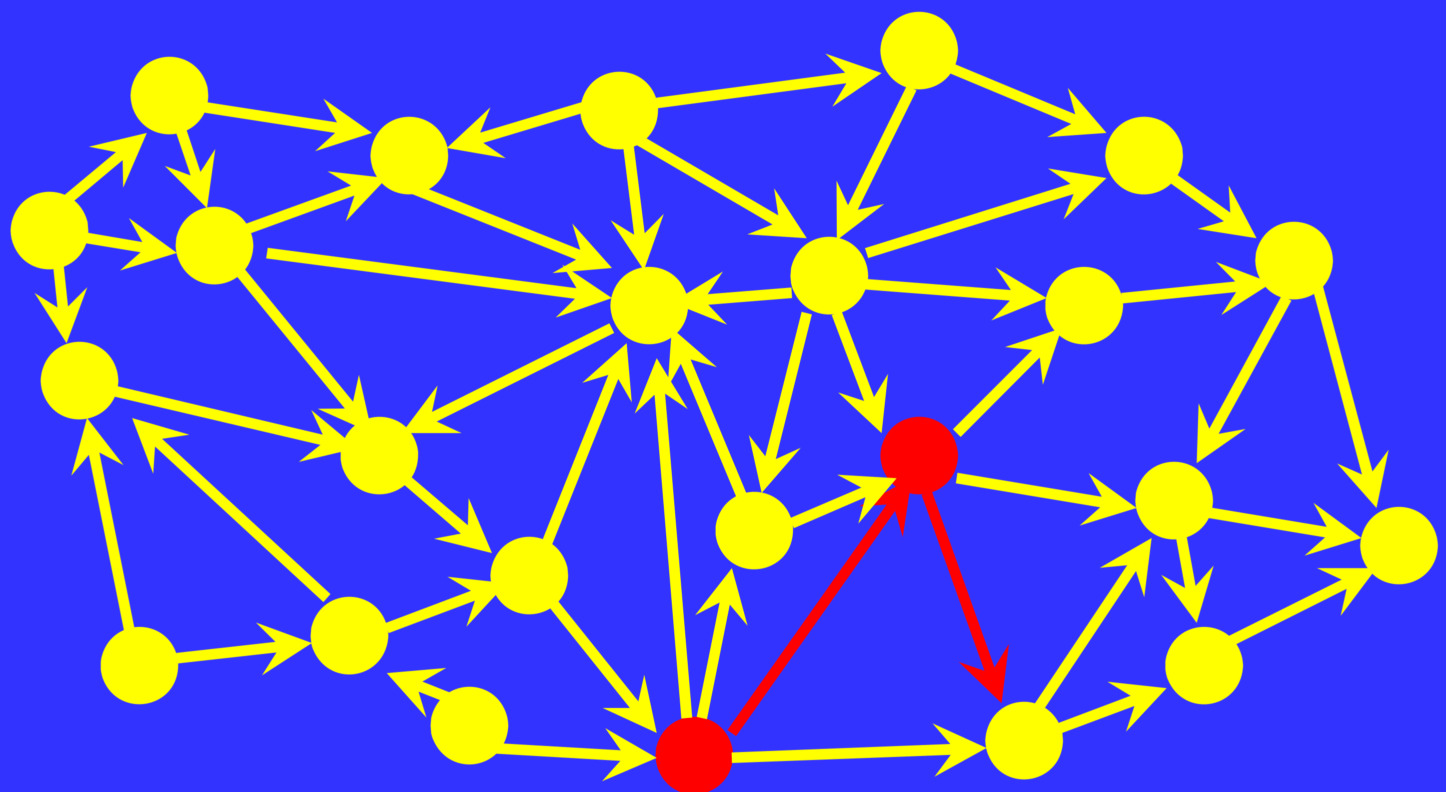
- Arriving at a new web page



IEIIT-CNR

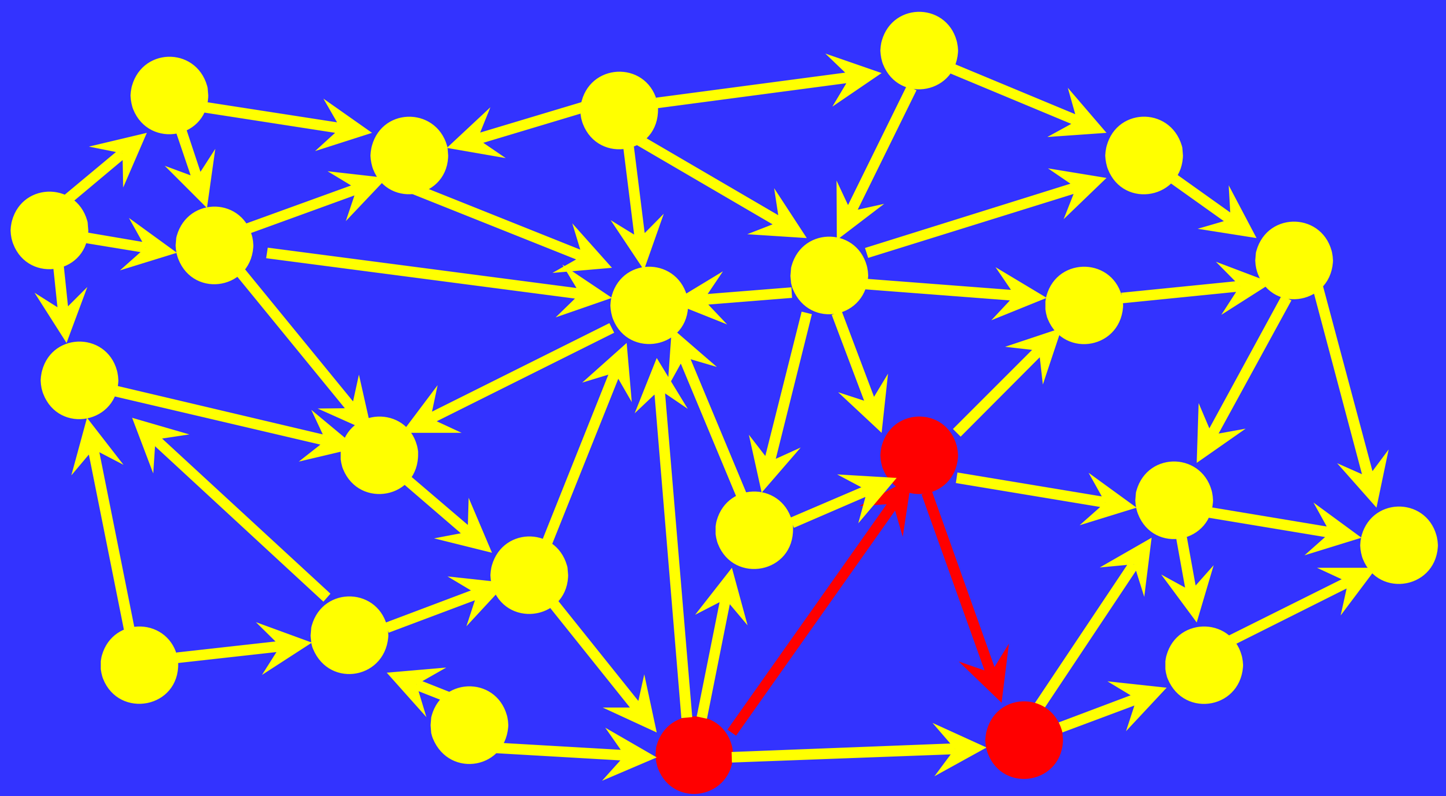
# Random Surfer Model

- Pick another outgoing link at random

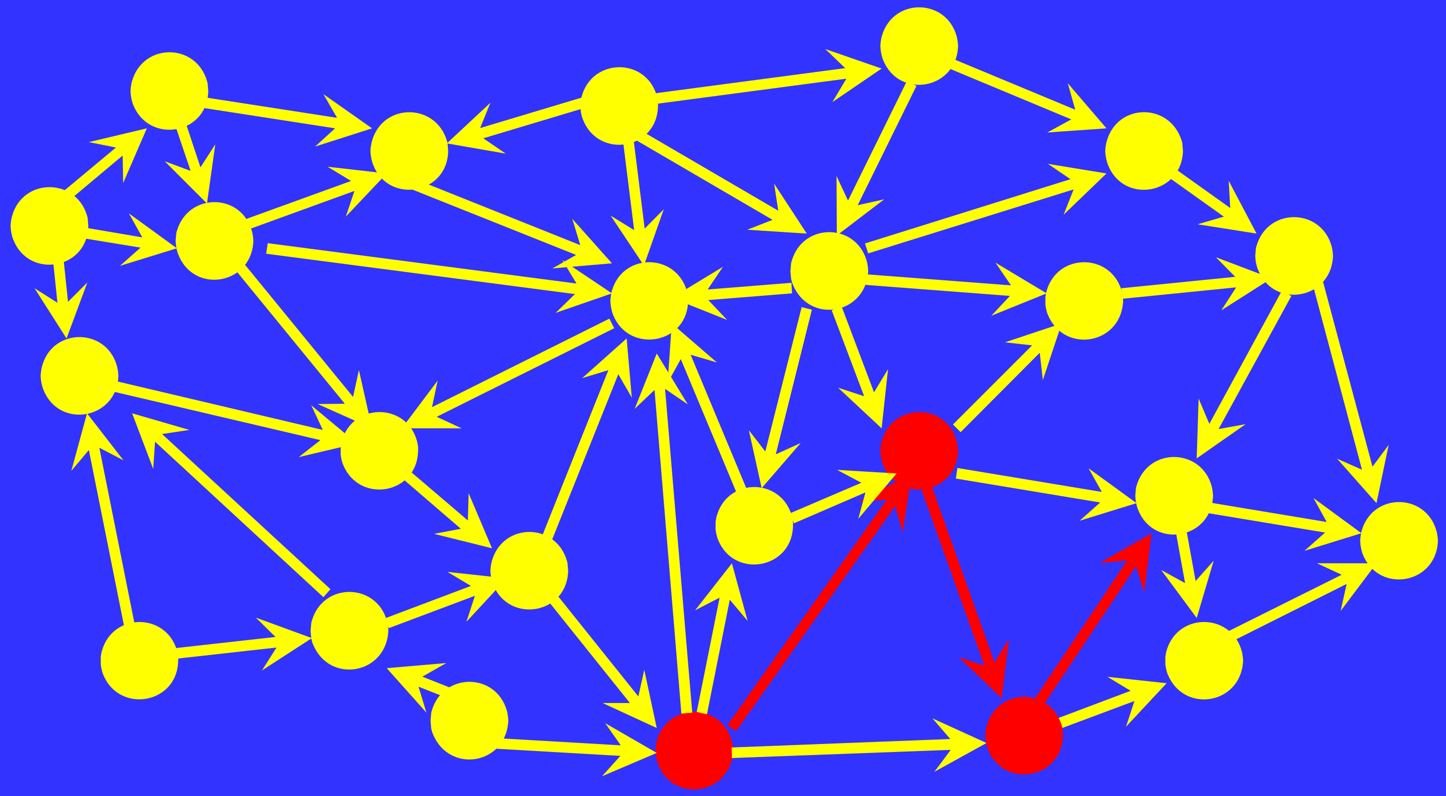


IEIIT-CNR

# Random Surfer Model

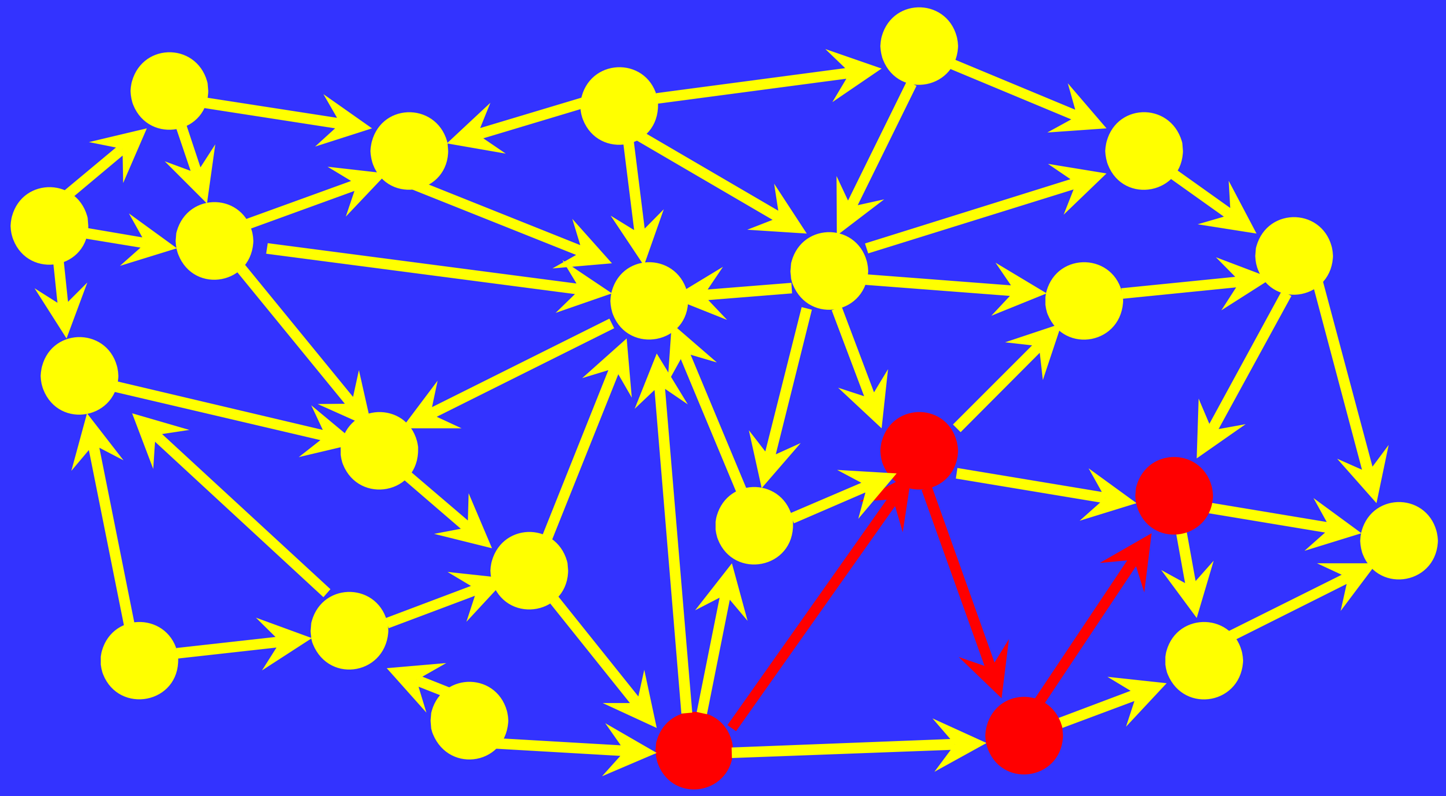


# Random Surfer Model



IEIIT-CNR

# Random Surfer Model



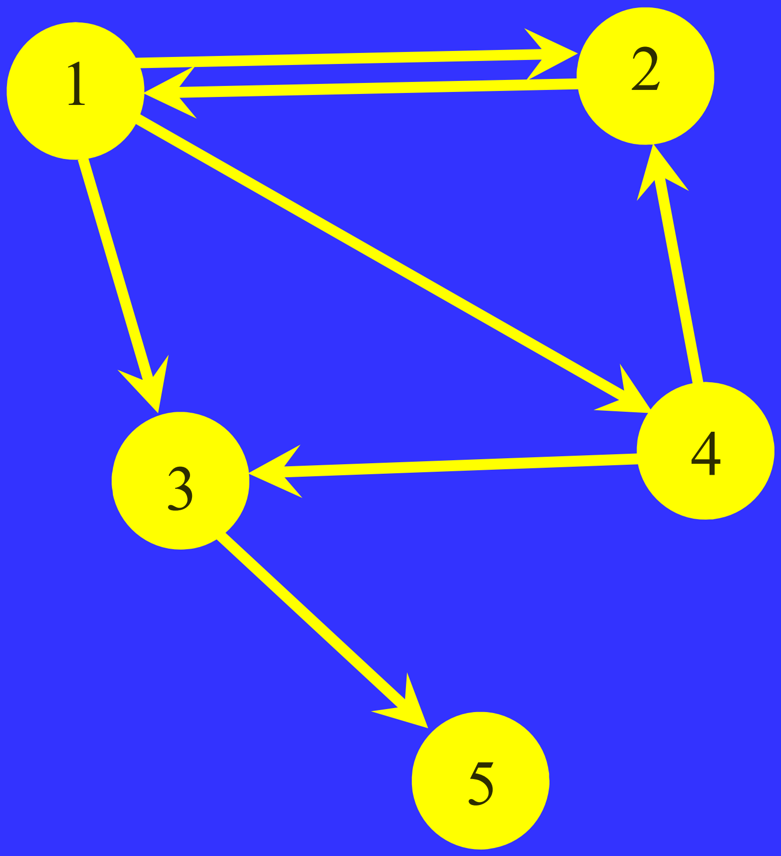
IEIIT-CNR

# Random Surfer Model

- If a page is “important” then it is visited more often...
- The time the random surfer spends on a page is a measure of the importance of the page
- If important pages point to your page, then your page becomes important (because it is often visited)
- Need to rank the pages in order of importance for facilitating the web search

IEIIT-CNR

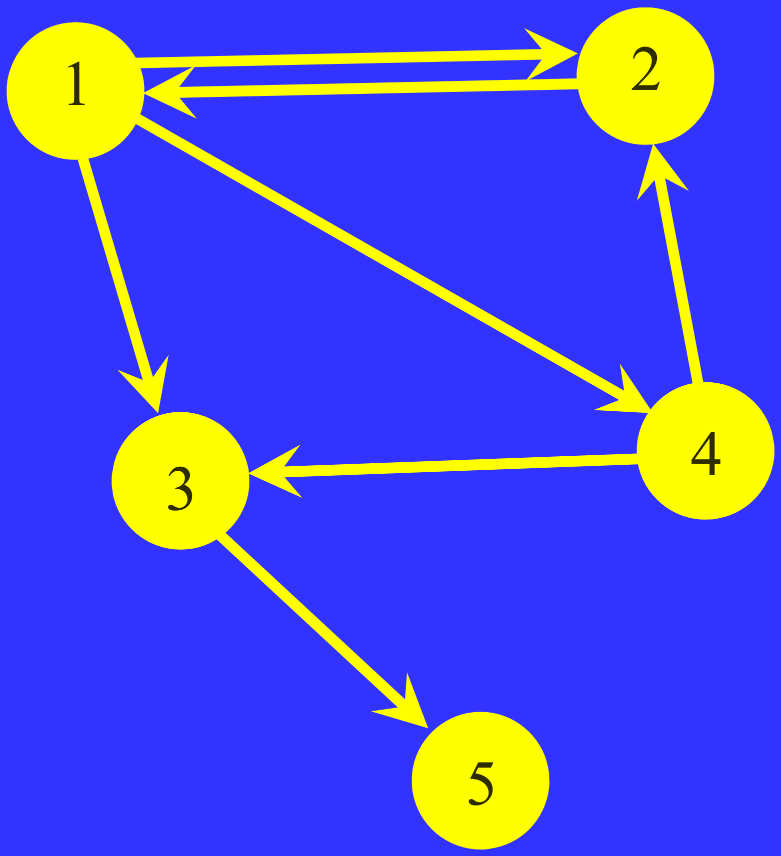
# Graph Representation




- Directed graph with nodes (pages) and links representing the web
- Graph is not necessarily strongly connected (from 5 you cannot reach other pages)
- Graph is constructed using *crawlers and spiders* moving continuously along the web

IEIIT-CNR

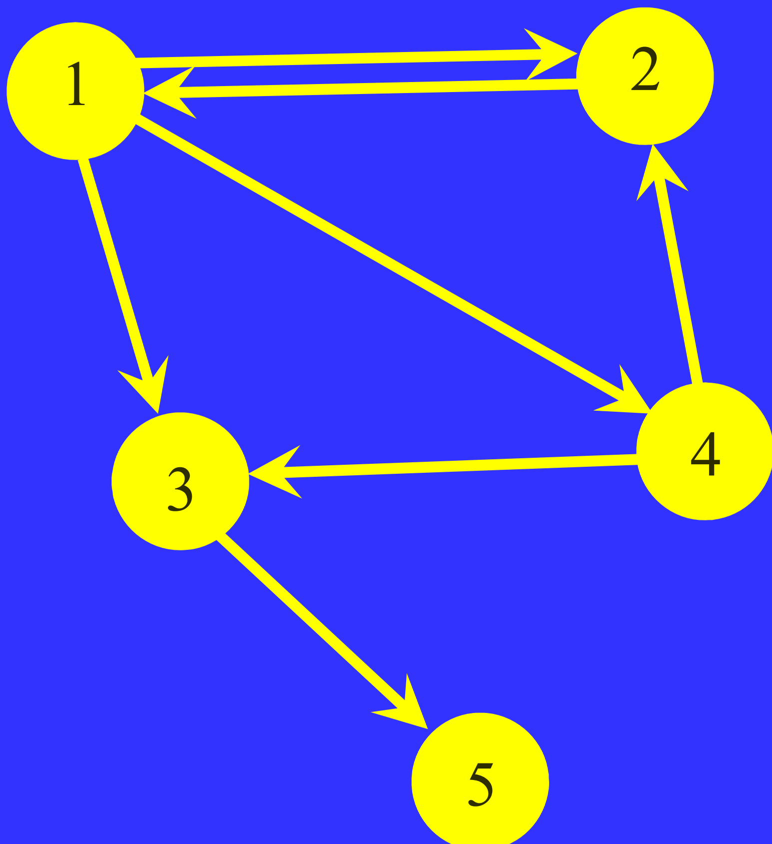
# Hyperlink Matrix




- For each node we count the number of outgoing links and normalize them to 1
- Hyperlink matrix is a nonnegative (column) substochastic matrix

IEIIT-CNR

# Hyperlink Matrix




$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

IEIIT-CNR

# PageRank: Bringing Order to the Web[1,2]

- Need to rank pages in order of importance
- The PageRank $x^*$ is defined as

$$x^* = Ax^* \quad \text{where} \quad x^* \in [0,1]^n \ \text{and} \ \sum_i x_i^* = 1$$

- $x^*$ is a nonnegative unit eigenvector corresponding to the eigenvalue 1 for the hyperlink matrix $A$

*The question is when $x^*$ exists and it is unique*

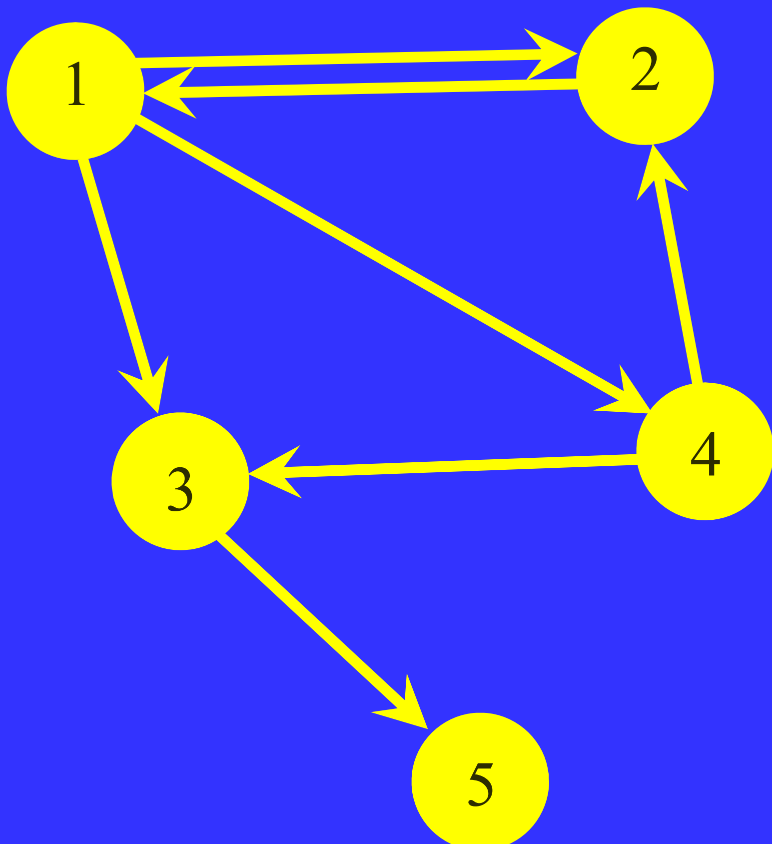
[1] S. Brin, L. Page (1998)
[2] S. Brin, L. Page, R. Motwani, T. Winograd (1999)

# Issue of Dangling Nodes

- **First issue:** We have *dangling nodes*
- Random surfer gets "stuck" when visiting a pdf file
- In this case the "back button" of the browser is used
- Mathematically, the hyperlink matrix is nonnegative and (column) substochastic

- **Easy fix:** Add artificial links to make the matrix stochastic

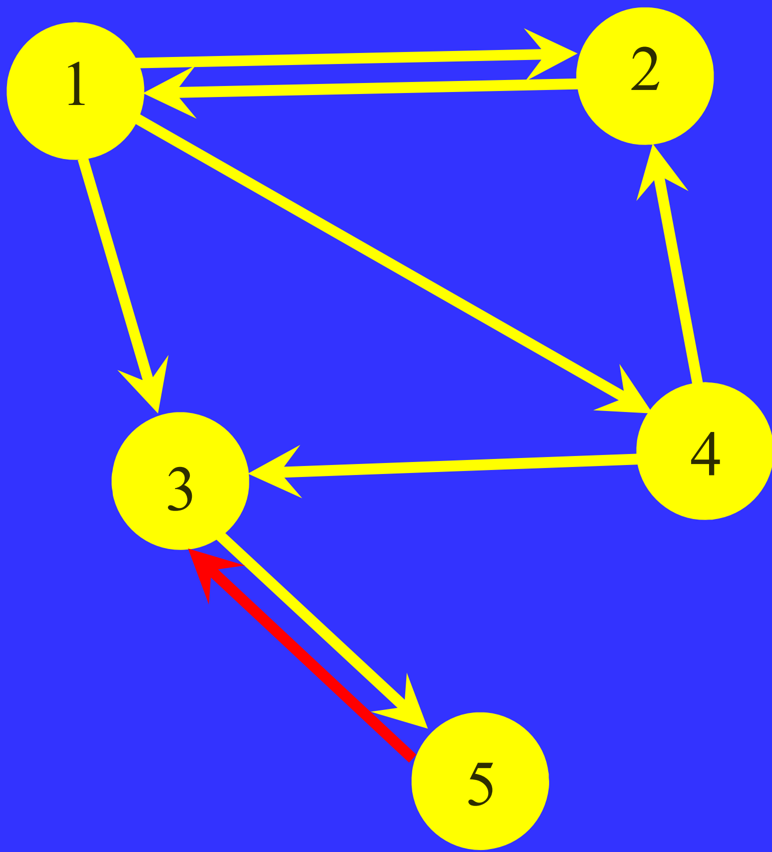
# Page 5 is a Dangling Node




$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

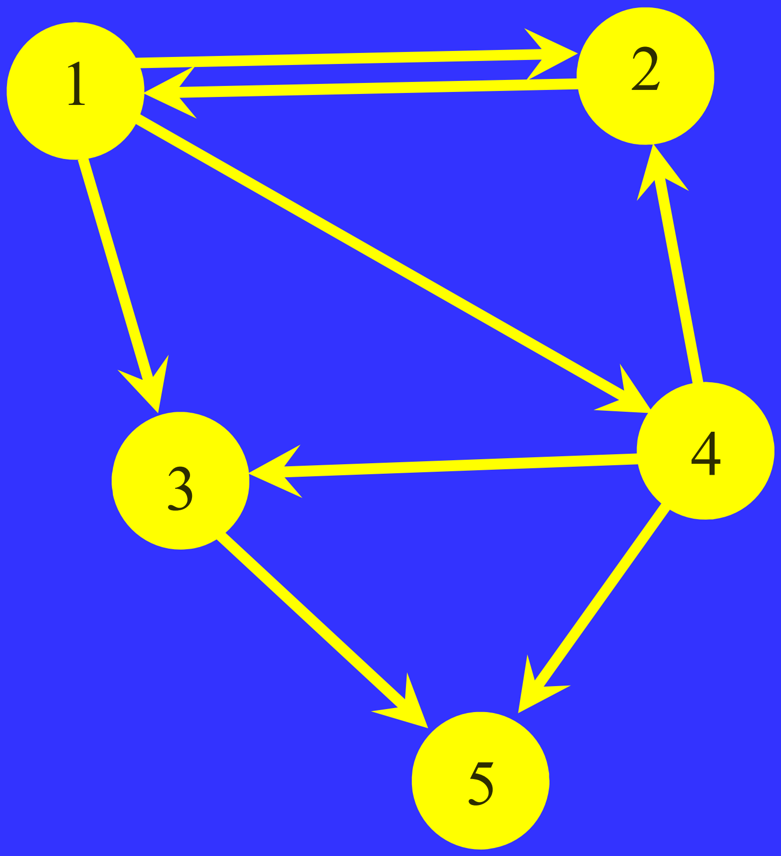
# Easy Fix

- We add an outgoing link to page 5




$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 0 & 0 & 1/2 & 1 \\ 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

# In General the Fix is not so Easy…

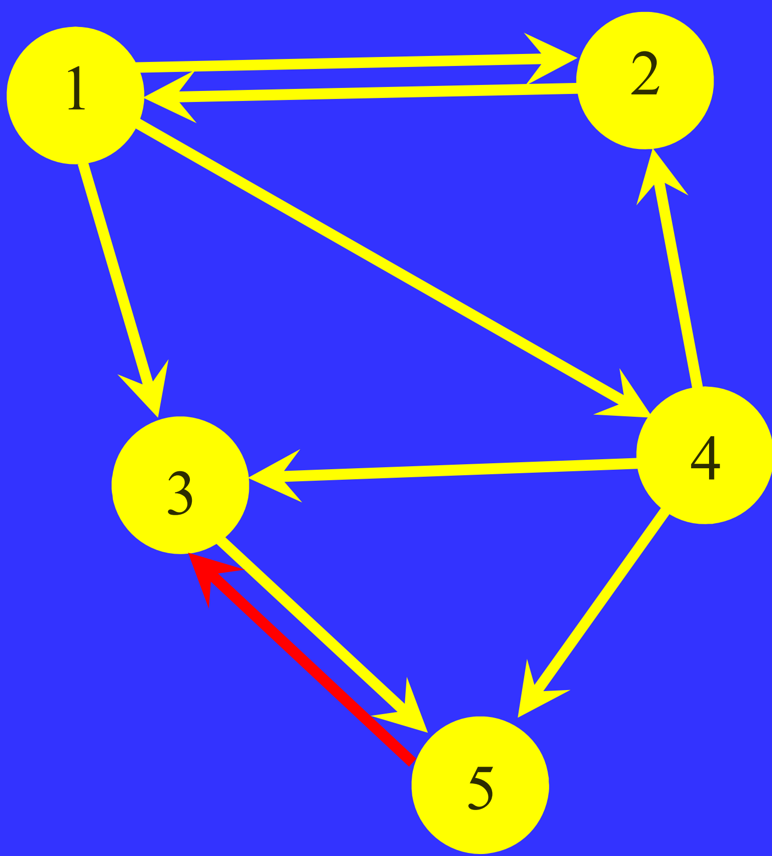



❖ Page 5 has two incoming links

IEIIT-CNR

# In General the Fix is not so Easy…

- Add an outgoing link from 5 to 3…

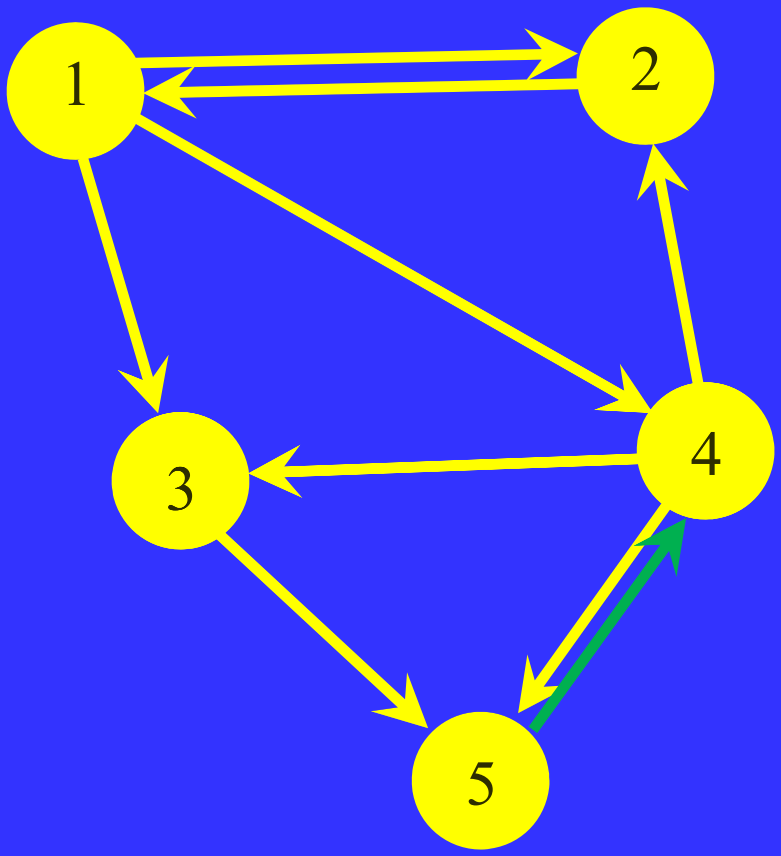



$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/3 & 0 \\ 1/3 & 0 & 0 & 1/3 & 1 \\ 1/3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1/3 & 0 \end{bmatrix}$$

IEIIT-CNR

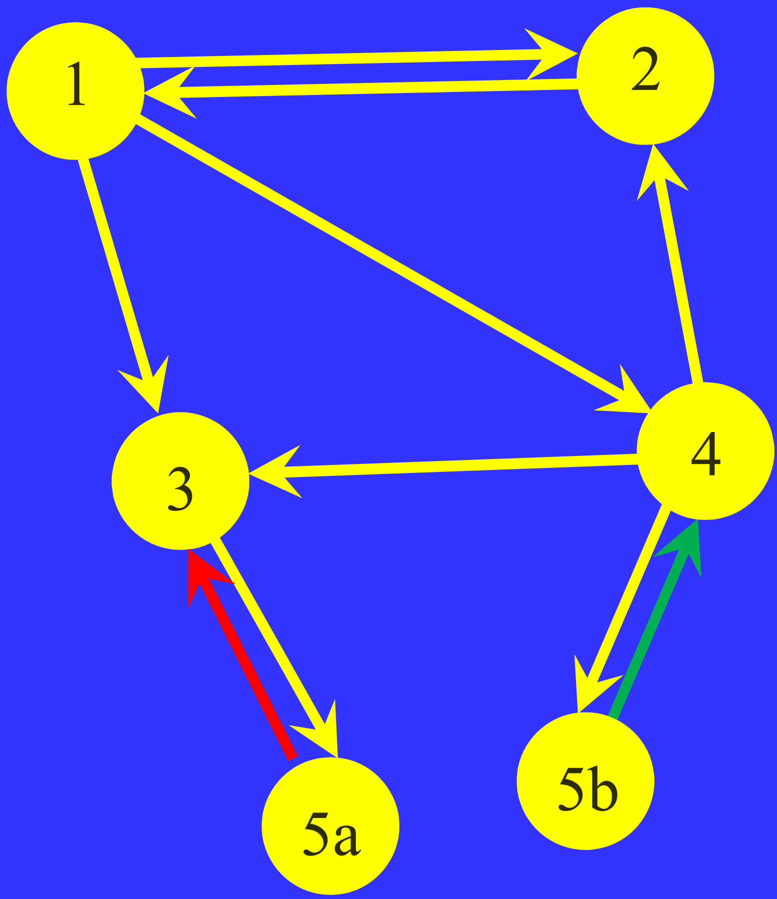
# But in General the Fix is not so Easy…

❖ … or do we add an outgoing link from 5 to 4?



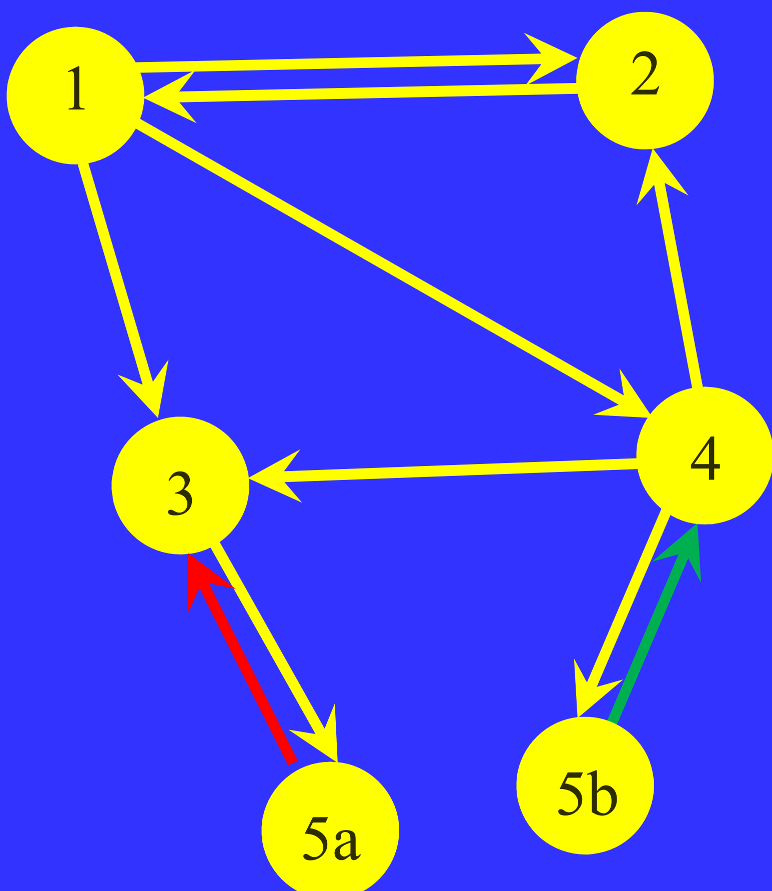

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/3 & 0 \\ 1/3 & 0 & 0 & 1/3 & 0 \\ 1/3 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1/3 & 0 \end{bmatrix}$$

# Modified Hyperlink Matrix




- Solution is to break page 5 into two pages 5a and 5b
- This artificially changes the number of pages (not only the number of links) and the topology of the network

# Modified Hyperlink Matrix




$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/3 & 0 & 0 \\ 1/3 & 0 & 0 & 1/3 & 1 & 0 \\ 1/3 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 0 & 0 \end{bmatrix}$$

IEIIT-CNR

# Assumption: No Dangling Nodes

- This is a web modeling problem
- Assume that there are *no dangling nodes*
- This implies that $A$ is a nonnegative stochastic matrix (instead of substochastic) having at least one eigenvalue equal to one
- Second issue: This eigenvalue is not necessarily unique

IEIIT-CNR

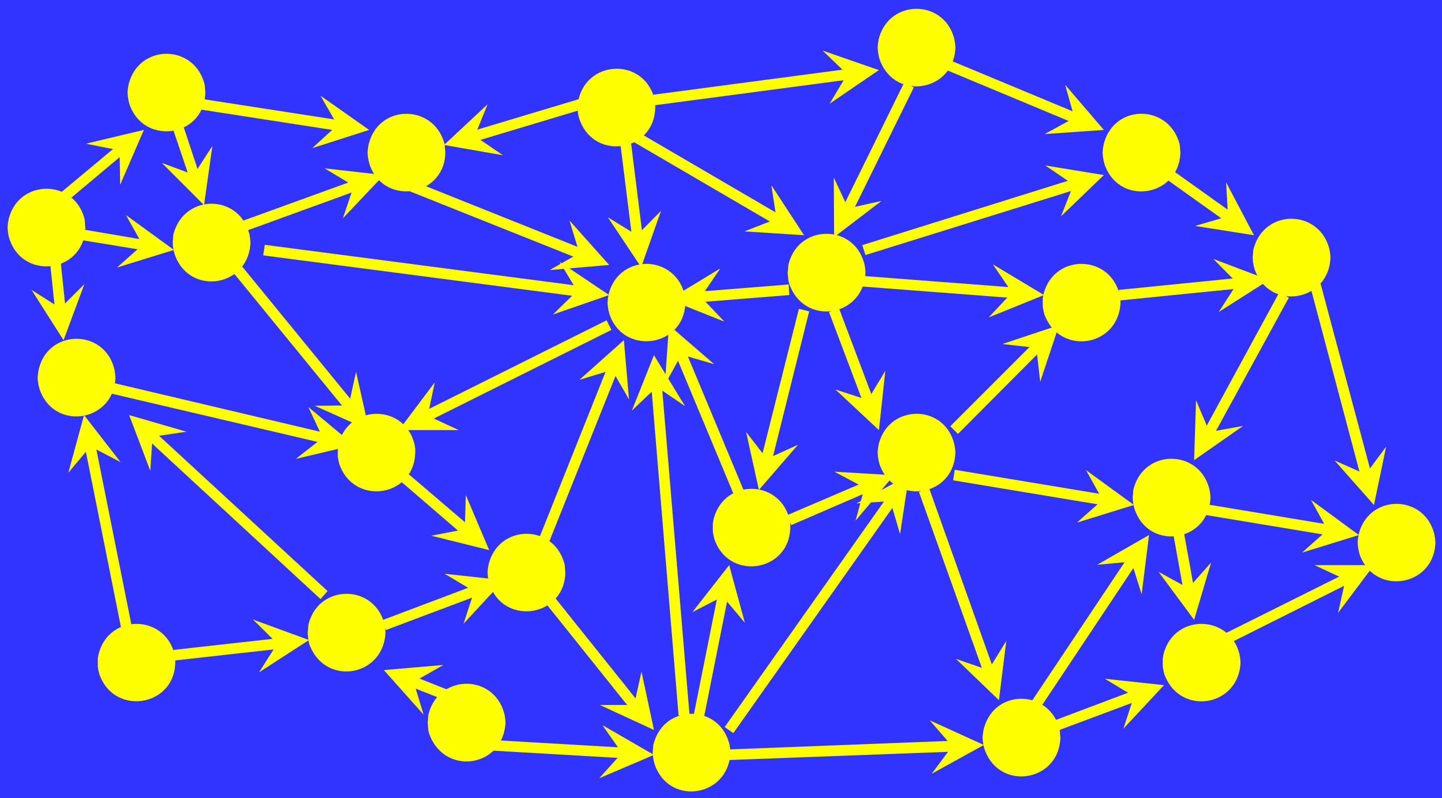
# Teleportation Matrix

- The random surfer may get bored after a while, and decides to “jump” to another page not directly connected to that currently visited

IEIIT-CNR

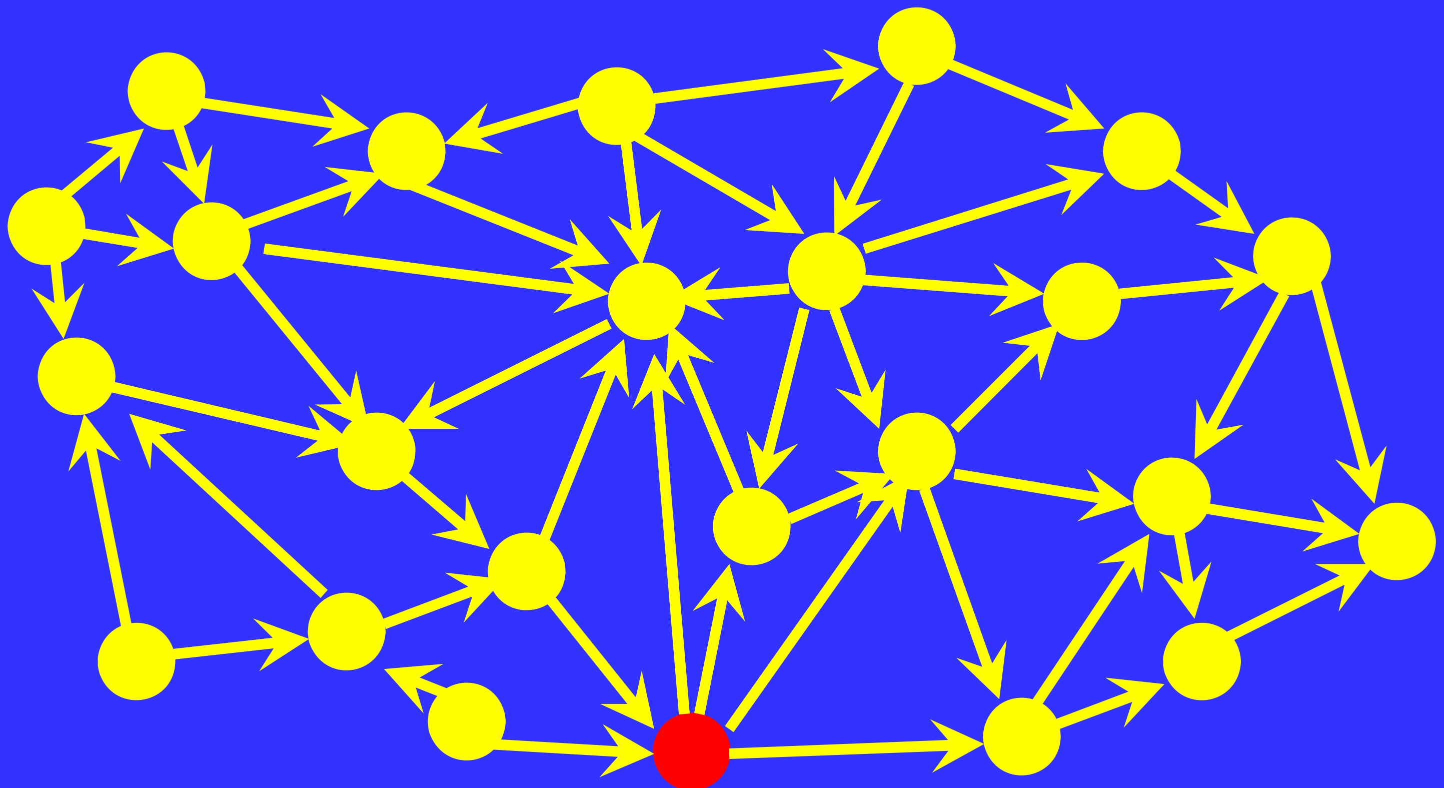
# Recall the Random Surfer Model

- Web representation with incoming and outgoing links

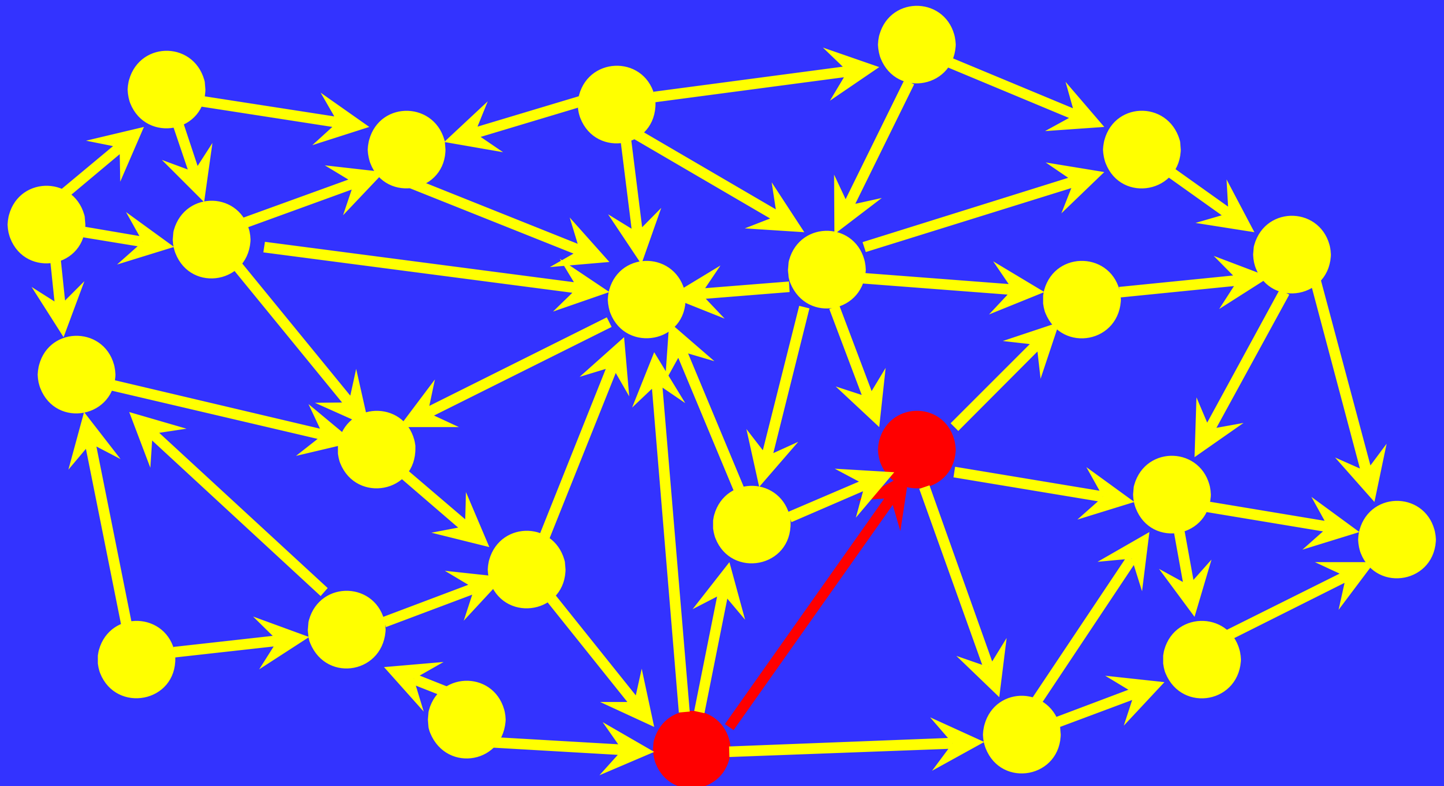


IEIIT-CNR

# Recall the Random Surfer Model

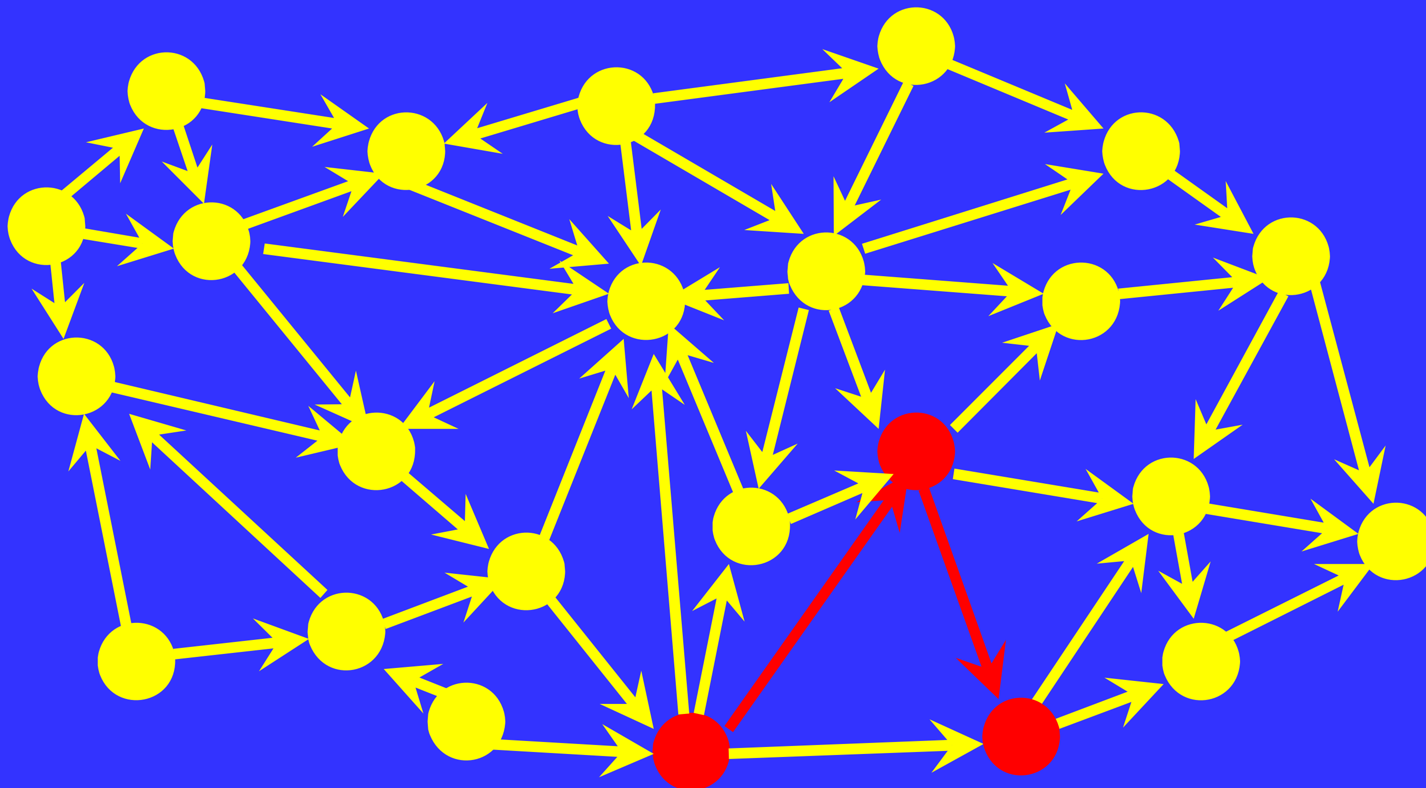


# Recall the Random Surfer Model



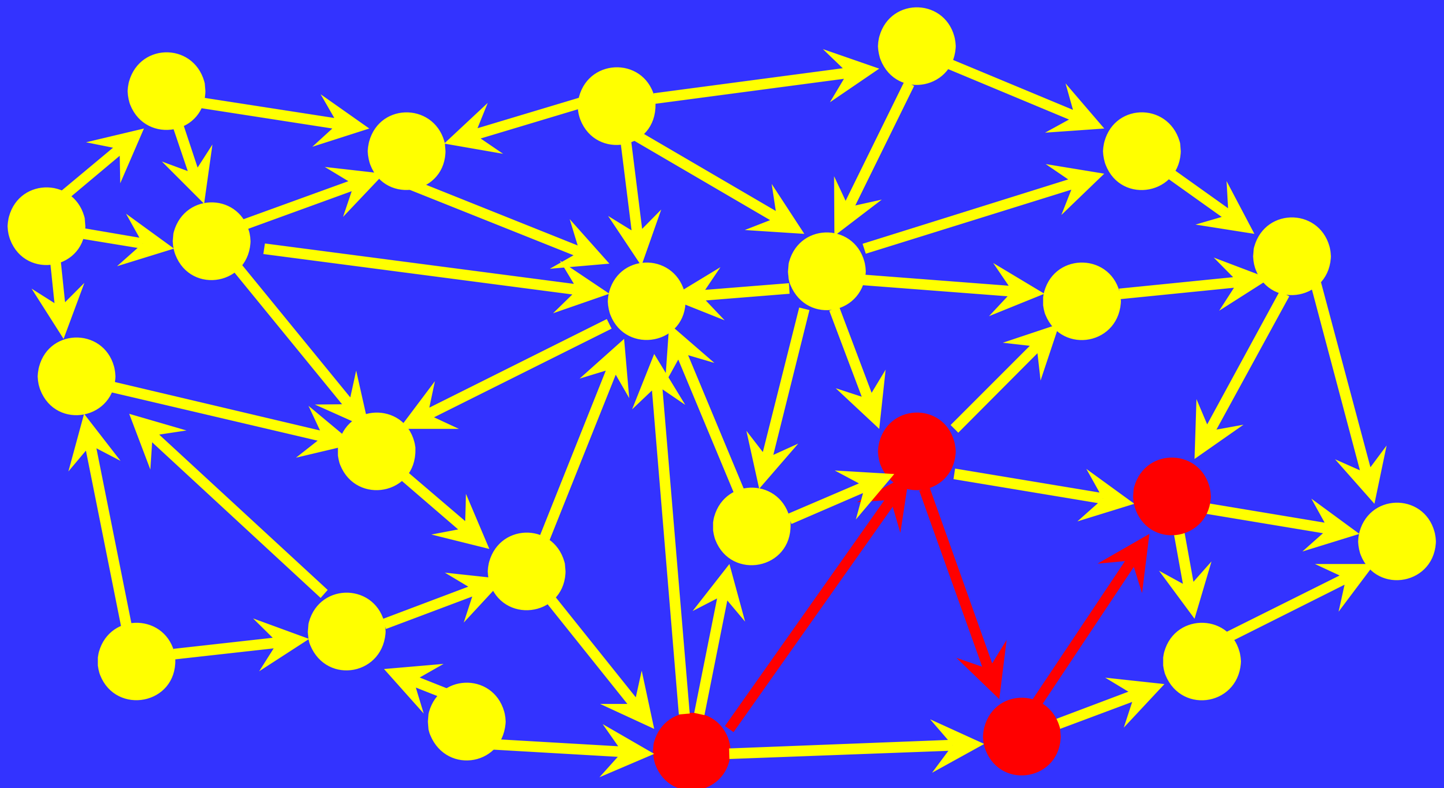
IEIIT-CNR

# Recall the Random Surfer Model



IEIIT-CNR

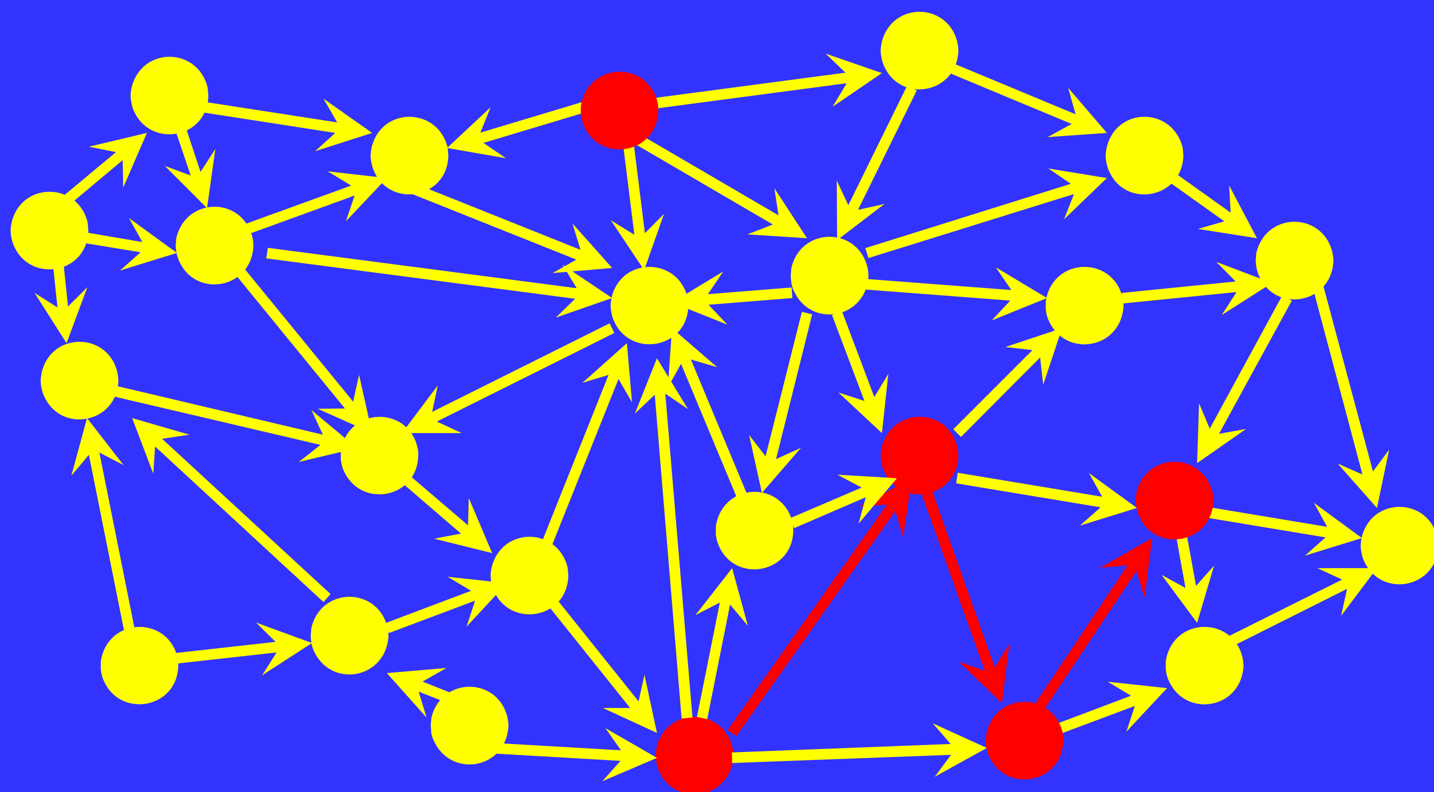
# Recall the Random Surfer Model



IEIIT-CNR

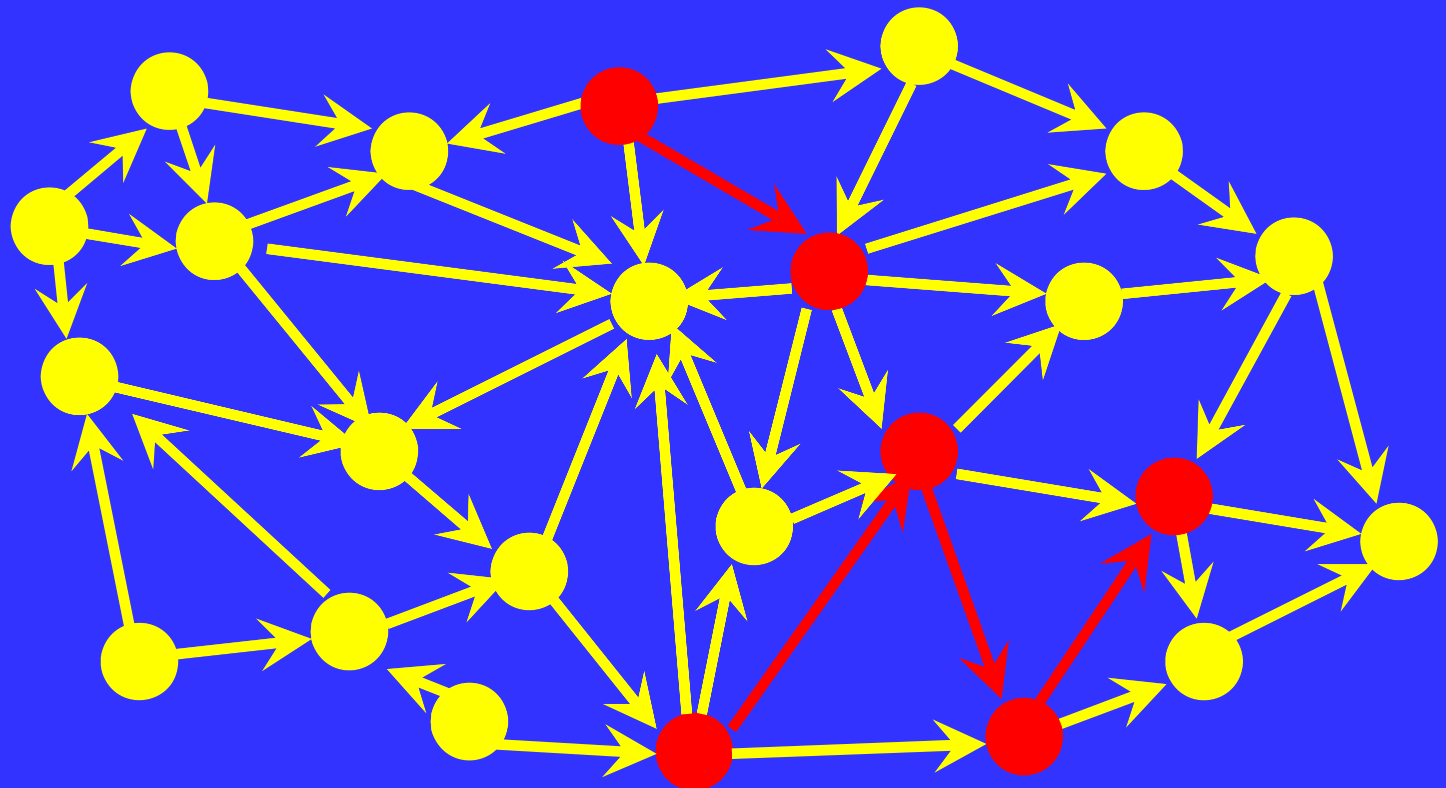
# Teleportation Model

- We are "teleported" to a web page located far away



# Random Surfer Model Again

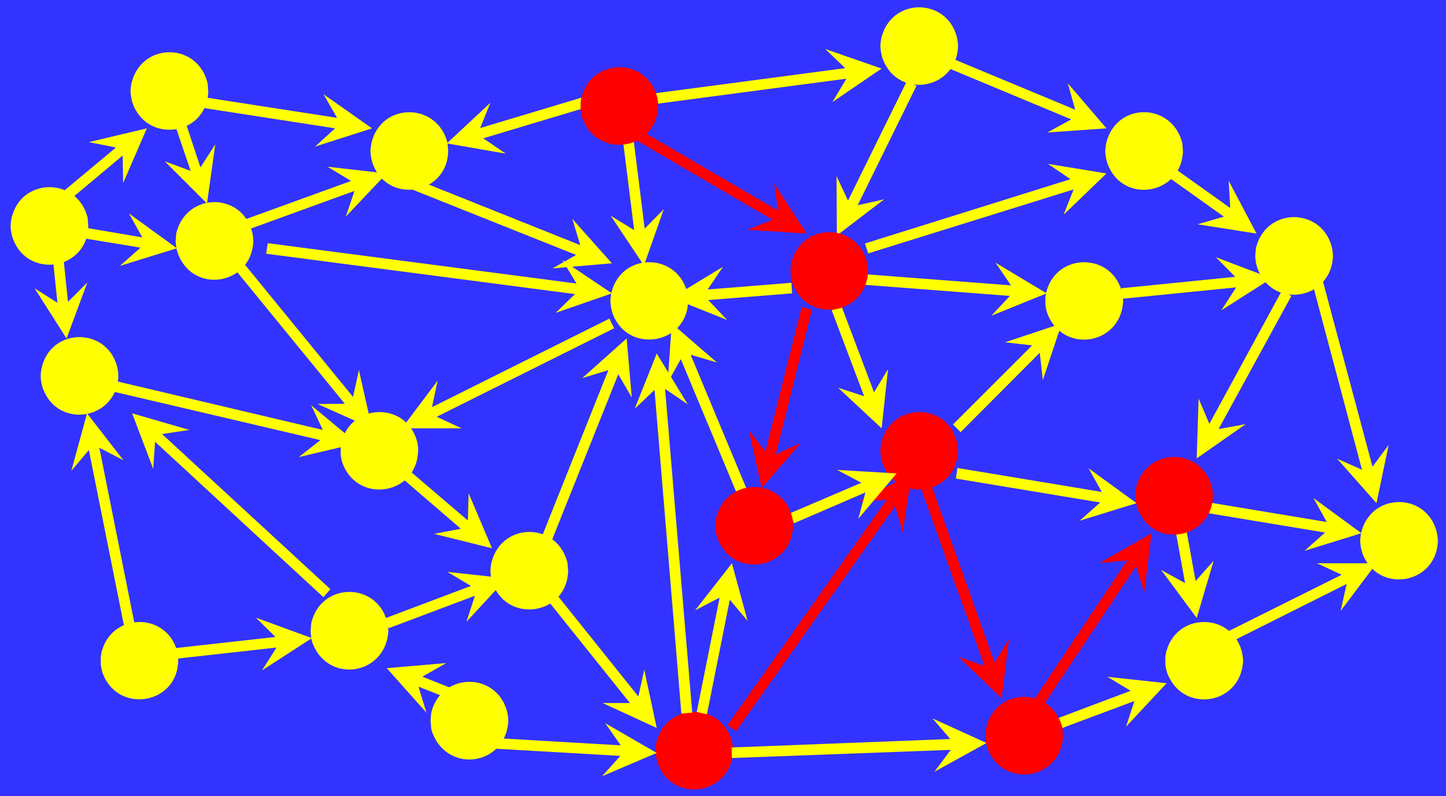
- Pick another outgoing link at random



IEIIT-CNR

# Random Surfer Model Again

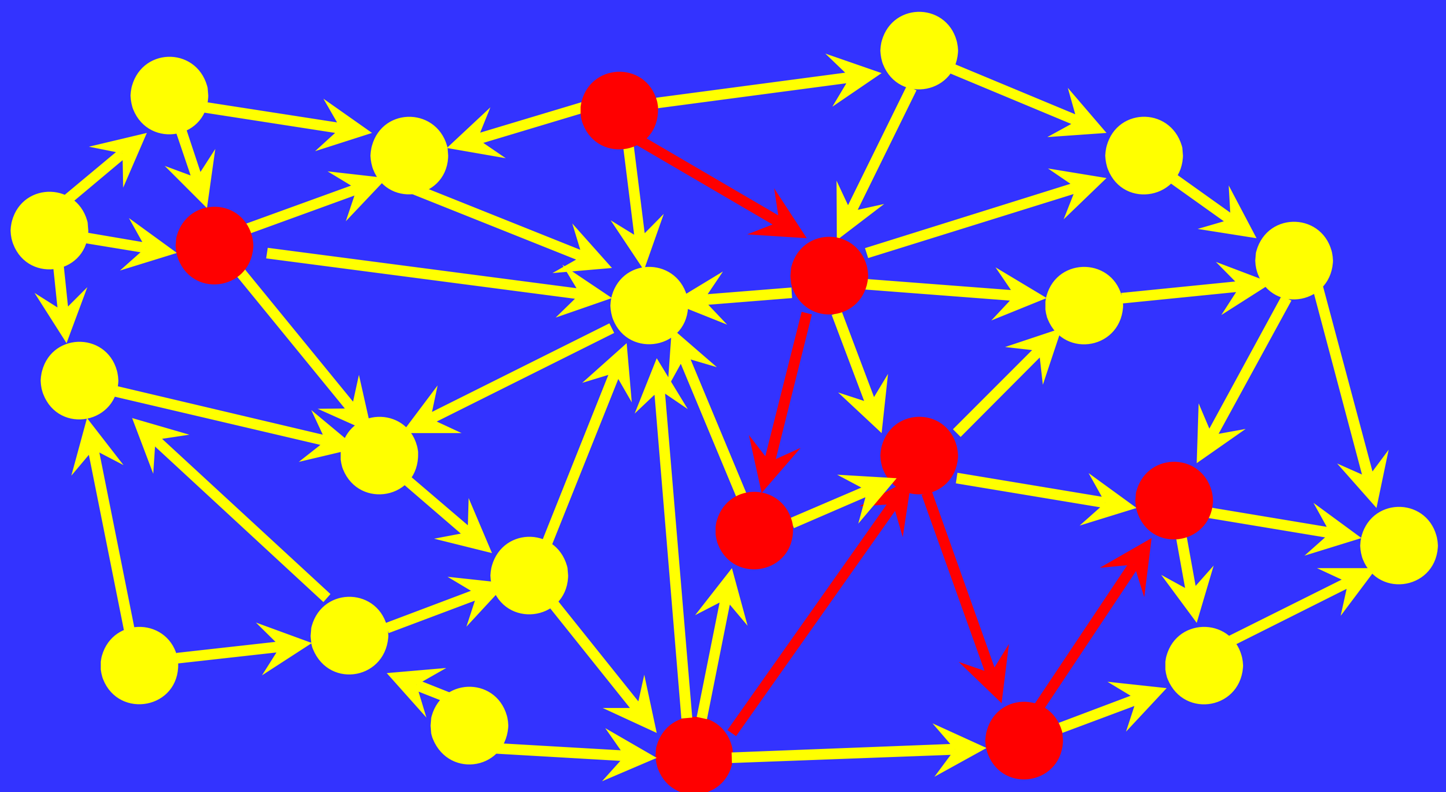
- Pick another outgoing link at random



IEIIT-CNR

# Teleportation Model Again

- We are teleported to another web page located far away



IEIIT-CNR

# Convex Combination of Matrices

- Teleported model is represented as a convex combination of matrices $A$ and $S$
- $S$ is a matrix with all entries equal to 1

$$S = \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix}$$

- Consider a matrix $M$ defined as

$$M = (1 - m) A + m/n\, S \qquad m \in (0,1)$$

where $n$ is the number of pages

- The value $m = 0.15$ is proposed and used at Google[1]

[1] S. Brin, L. Page (1998)

IEIIT-CNR

# Matrix $M$

- Property: $M$ is positive stochastic (convex combination of two stochastic matrices and $m \in (0,1)$)

IEIIT-CNR

# Matrix *M* and Perron Theorem

- Matrix *M* is primitive ($M^k$ is positive for some *k*)
- *M* is irreducible and the corresponding graph is strongly connected (every page is connected to every page)
- The eigenvalue 1 is the *unique* eigenvalue of maximum modulus (simple eigenvalue)
- The corresponding eigenvector is positive

IEIIT-CNR

# PageRank Computation

IEIIT-CNR

# PageRank Computation with Power Method

- PageRank is computed with the power method

$$x(k+1) = M\, x(k)$$

- Convergence is guaranteed by Perron Theorem because $M$ is a primitive matrix

$$x(k) \to x^* \quad \text{for} \quad k \to \infty$$

provided that $\sum_i x_i(0) = 1$

- Remark: PageRank computation may be interpreted as finding the stationary distribution of a Markov Chain

IEIIT-CNR

# Rate of Convergence of Power Method

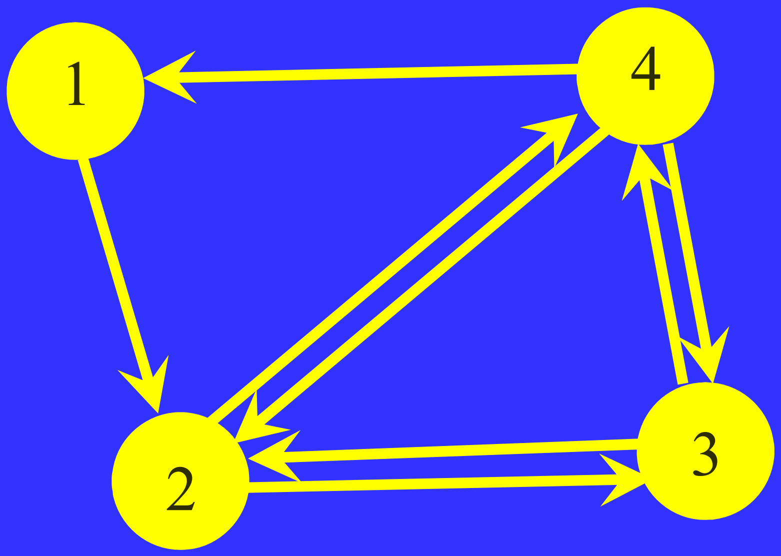
- Asymptotic rate of convergence of power method is *exponential* and depends on the ratio

$$|\lambda_2(M) / \lambda_1(M)|$$

- We have

$$\lambda_1(M) = 1 \qquad \lambda_2(M) \leq 1 - m = 0.85$$

# PageRank Computation with Power Method




$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix} \qquad m = 0.15$$

$$M = \begin{bmatrix} 0.038 & 0.037 & 0.037 & 0.321 \\ 0.887 & 0.037 & 0.462 & 0.321 \\ 0.037 & 0.462 & 0.037 & 0.321 \\ 0.037 & 0.462 & 0.462 & 0.037 \end{bmatrix}$$

$$x^* = \begin{bmatrix} 0.12 & 0.33 & 0.26 & 0.29 \end{bmatrix}^T$$

# Size of the Web

- The size of $M$ is 8 billion!
- The PageRank computation requires 50-100 iterations
- This computation takes about a week and it is performed centrally at Google once a month
- More and more computing power is needed…

IEIIT-CNR

# Columbia River, The Dalles, Oregon

# Randomized Decentralized Approach

IEIIT-CNR

# Randomized Decentralized Approach

- **Main idea:** Develop a decentralized approach for computing PageRank (instead of a centralized approach which involves the entire web)

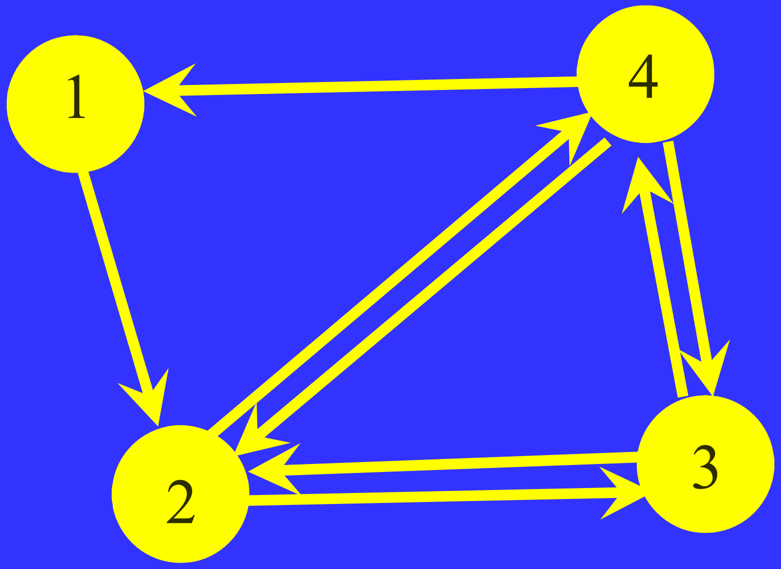
# Randomized Decentralized Approach

- **Main idea:** Develop a decentralized approach for computing PageRank (instead of a centralized approach which involves the entire web)

- Approach is randomization-based (Las Vegas type)
- **RA**[1]: algorithm making random choices during execution
- **LVRA**: RA requiring a *finite* number of random choices (contrary to Monte Carlo)

[1] R. Tempo, G. Calafiore, F. Dabbene (2005)

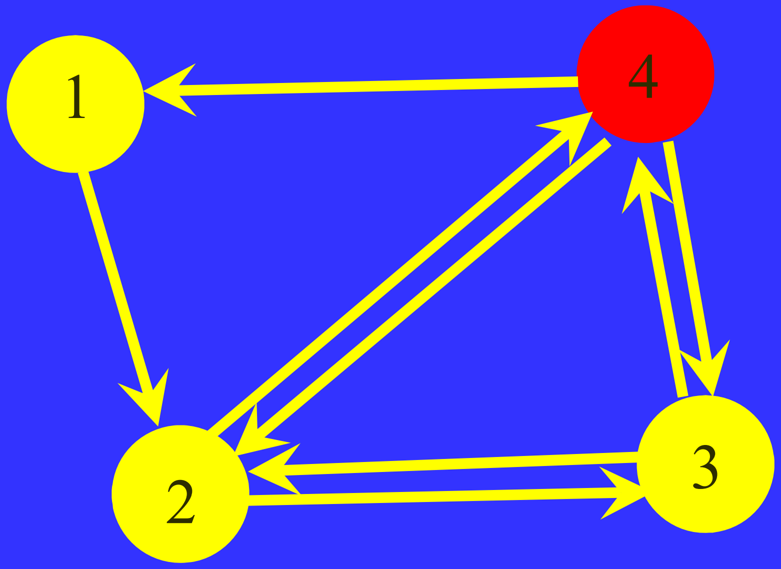
# Basic Communication Protocol




Communication protocol: at time $k$ randomly select page $i$ ($i$=4) for PageRank update

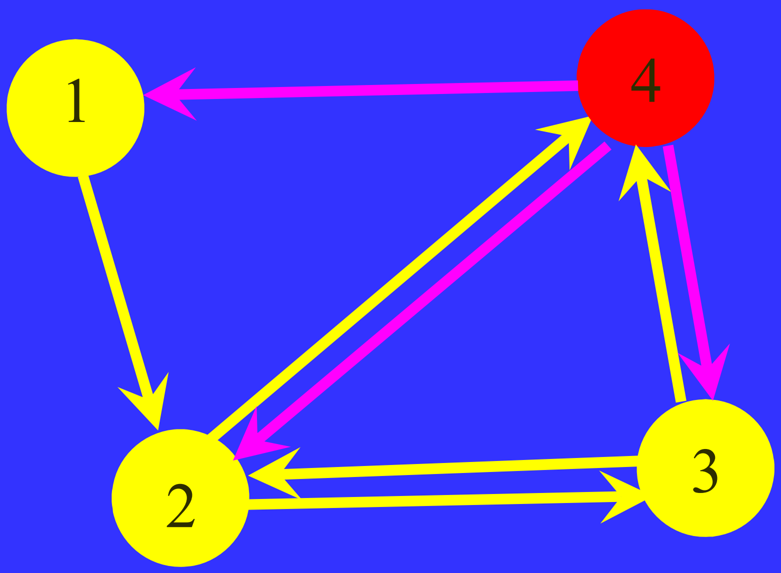
IEIIT-CNR

# Basic Communication Protocol




Communication protocol: at time $k$ randomly select page $i$ ($i$=4) for PageRank update

# Basic Communication Protocol

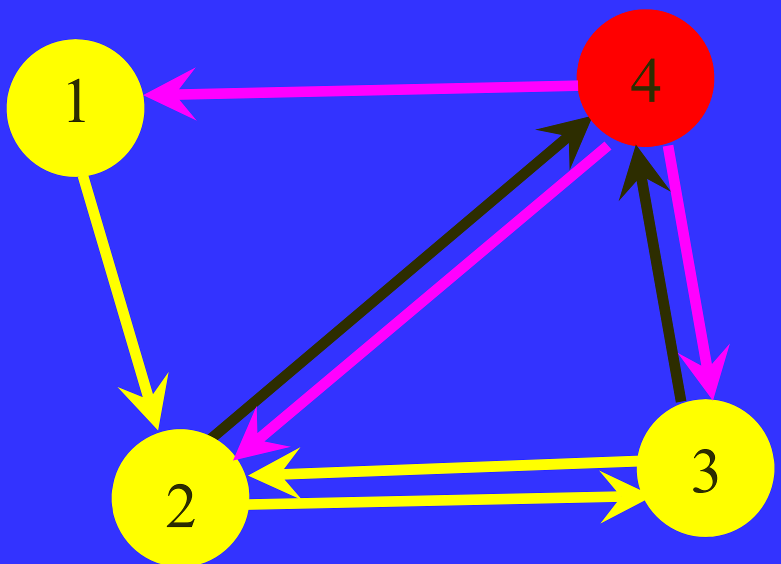



Communication protocol: at time $k$ randomly select page $i$ ($i$=4) for PageRank update

1. send PageRank value of page $i$ to the outgoing pages that are linked to $i$

IEIIT-CNR

# Basic Communication Protocol




Communication protocol: at time $k$ randomly select page $i$ ($i$=4) for PageRank update

1. send PageRank value of page $i$ to the outgoing pages that are linked to $i$

2. request PageRank values from incoming pages that are linked to page $i$

# Las Vegas Randomized Approach

- The pages taking action are determined via a random process $\theta(k) \in \{1, \ldots, n\}$
- $\theta(k)$ is assumed to be i.i.d. with uniform probability

$$\mathrm{Prob}\{\theta(k)=i\} = 1/n$$

- If at time $k$ $\theta(k) = i$ then page $i$ initiates PageRank update

# Distributed Randomized Update Scheme

- Consider the randomized update scheme

$$x(k+1) = A_{\theta(k)}\, x(k)$$

where $A_{\theta(k)}$ are the *distributed link matrices* (example next)

- Define the time average

$$y(k) = 1/(k+1) \sum_i x(i)$$

IEIIT-CNR

# Distributed Link Matrices - 1

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} & & & 1/3 \\ & & & 1/3 \\ & & & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

IEIIT-CNR

# Distributed Link Matrices - 2

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} & 0 & 0 & 1/3 \\ 0 & & 0 & 1/3 \\ 0 & 0 & & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

IEIIT-CNR

# Distributed Link Matrices - 3

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 1 & 0 & 0 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

IEIIT-CNR

# Distributed Link Matrices - 4

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 0 & 1/2 & 2/3 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 1 & 0 & 0 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

IEIIT-CNR

# Distributed Link Matrices - 5

$$A = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 0 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 0 & 0 & 0 & 1/3 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 2/3 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1/2 & 1/3 \\ 0 & 1/2 & 1/2 & 0 \\ 0 & 1/2 & 0 & 2/3 \end{bmatrix}$$

# Average Matrix $\underline{A}$

- Average matrix $\underline{A} = E[\, A_{\theta(k)} \,]$
- Taking uniform distribution in the random process $\theta(k)$ we have

$$\underline{A} = 1/n \sum_i A_i$$

- **Lemma** (properties of the average matrix $\underline{A}$)

  (i) $\underline{A} = 2/n\, A + (1-2/n)\, I$

  (ii) Matrices $A$ and $\underline{A}$ have the same eigenvector corresponding to the eigenvalue 1

IEIIT-CNR

# Modified Distributed Update Scheme

- Need to work with positive stochastic matrices
- Consider the modified randomized distributed update scheme

$$x(k+1) = M_{\theta(k)}\ x(k)$$

where $M_{\theta(k)}$ are the *modified distributed link matrices* computed as

$$M_i = (1- r)\, A_i + r/n\, S \quad i = 1, 2, \ldots, n$$

and $r \in (0,1)$ is a design parameter (defined next)

IEIIT-CNR

# Average Matrix $\underline{M}$

- Average matrix $\underline{M} = E[\, M_{\theta(k)} \,]$
- Define $r = 2m/(n - mn + 2m)$
- Lemma (properties of the average matrix $\underline{M}$)

  (i) $r \in (0,1)$ and $r < m = 0.15$

  (ii) $\underline{M} = r/m\, M + (1 - r/m)\, I$

  (iii) The eigenvalue 1 for $\underline{M}$ is the unique eigenvalue of maximum modulus (simple eigenvalue). PageRank is the corresponding eigenvector

IEIIT-CNR

# Convergence Properties[1]

- **Theorem** (convergence properties)
- The time average $y(k)$ of the modified randomized distributed update scheme converges to PageRank $x^*$ in MSE

$$E[\ \|y(k) - x^*\|^2\ ] \to 0 \quad \text{for} \quad k \to \infty$$

provided that $\sum_i x_i(0) = 1$

- **Proof:** Based on the theory of ergodic matrices

[1] H. Ishii, R. Tempo (2010)

IEIIT-CNR

# Comments

- Time average $y(k)$ can be computed recursively as a function of $y(k-1)$
- Sparsity of the matrix $A_i$ can be preserved because

$$x(k+1) = M_i\, x(k) = (1-r)\, A_i\, x(k) + r/n\, \mathbf{1}$$

where **1** is a vector with all entries equal to one

- Convergence rate is $1/k$
- Stopping criteria to compute approximately PageRank
- Similar convergence results for different update schemes based only on outgoing links (not incoming)

IEIIT-CNR

# PageRank for Systems and Control

The PageRank problem is useful for developing novel ideas for systems and control

IEIIT-CNR

# Two Questions

The PageRank problem is useful for developing novel ideas for systems and control

1. How is PageRank related to systems and control?
2. What can we learn from PageRank which is useful for systems and control?

IEIIT-CNR

# Consensus and PageRank

IEIIT-CNR

# Torre Aquila, Trento, Italy

- Torre Aquila (Eagle Tower): 31 meters-tall tower located in the Buonconsiglio castle in Trento
- Group of 11 frescoes named "Ciclo dei Mesi" (Cycle of the Months)
- Painted in the 15th century by an unknown artist from Bohemia
- Noteworthy examples of gothic art
- For increasing traffic, need to build road tunnel creating vibrations

IEIIT-CNR

# WSN (Wireless Sensor Network) in Torre Aquila, Trento, Italy[1]

- WSN with *environmental sensors* measuring temperature, humidity, light; *acceleration sensors* measure vertical deformation of the tower and sliding of two walls
- data collection from heterogeneous sensors
- data dissemination to enable remote tasking

[1] M. Ceriotti, L. Mottola, G. P. Picco, A. L. Murphy, S. Guna, M. Corrà, M. Pozzi, D. Zonta, P. Zanon (2009)

# WSN and Consensus

- SN consist of a large number of ultra-small autonomous devices (sensor nodes)
- Each sensor node is battery powered, has data processing capabilities, and short-range radio communications
- To achieve security in WSN we need to encrypt messages sent among sensor nodes
- Keys for encryption purposes must be agreed upon by communicating nodes
- Due to resource constraints, achieving such key agreement (consensus) is non-trivial

IEIIT-CNR

# Graph of Agents (Sensor Nodes)

- Consider a graph of agents (sensor nodes) which communicate using a random protocol
- The value (e.g. temperature) of agent $i$ at time $k$ is $x_i$
- **Objective:** reach consensus[1] in the presence of faults[2]
- Values of agents are updated using a LVRA

$$x(k+1) = A_{\theta(k)}\ x(k)$$

[1] P. J. Antsaklis, J. Baillieul (2007)
[2] M. Pease, R. Shostak, L. Lamport (1980)

IEIIT-CNR

# Random Communication Pattern

- Communication pattern is random
- The patterns are determined via a random process $\theta(k) \in \{1, \ldots, d\}$ where $d$ is the number of patterns
- $\theta(k)$ is assumed to be i.i.d. with uniform probability

$$\mathrm{Prob}\{\theta(k)=i\} = 1/d$$

- If at time $k$ $\theta(k) = i$ then agent $i$ initiates update

# Consensus

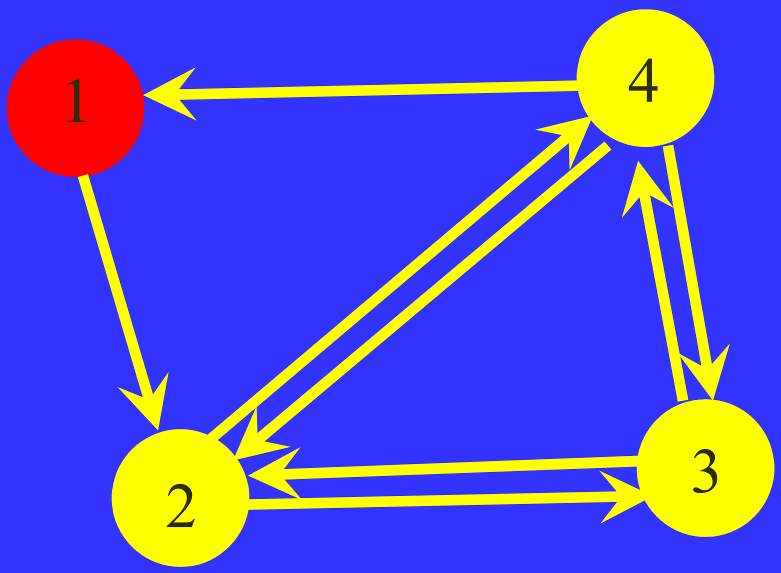
- **Definition:** Consensus is achieved if for any initial condition $x(0)$ we have

$$|\, x_i(k) - x_j(k) \,| \rightarrow 0 \quad \text{for} \;\; k \rightarrow \infty$$

with probability one for all agents $i, j$

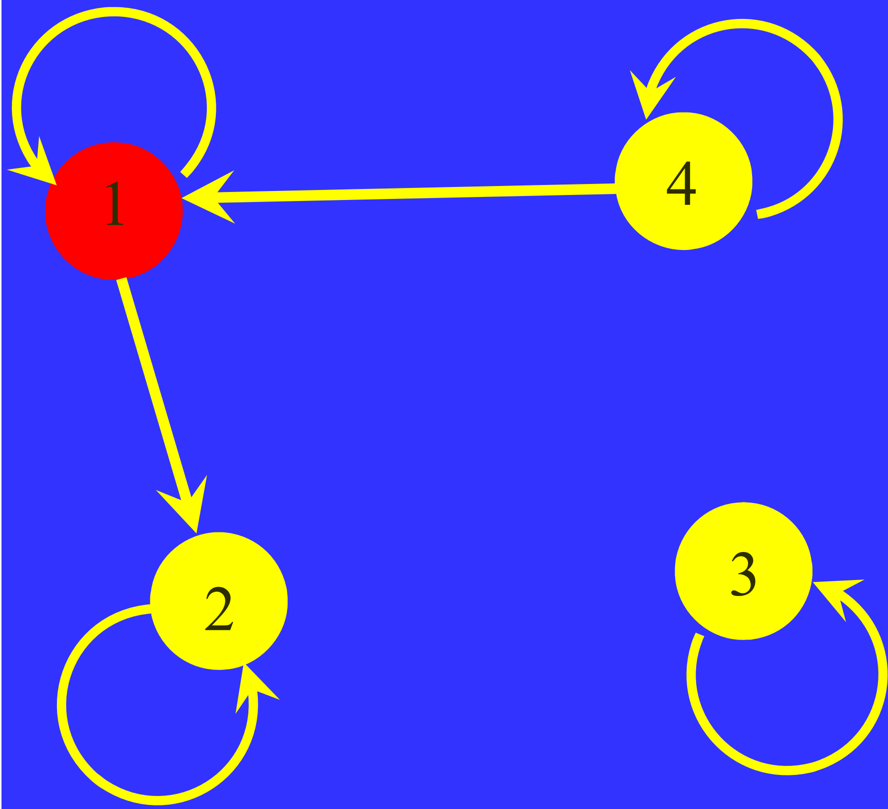
IEIIT-CNR

# Communication Pattern for Agent 1




For agent 1 we consider only the agents communicating to it

# Communication Pattern for Agent 1




$$A_1 = \begin{bmatrix} * & 0 & 0 & * \\ * & * & 0 & 0 \\ 0 & 0 & * & 0 \\ 0 & 0 & 0 & * \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 1/2 & 0 & 0 & 1/2 \\ 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$A$ is row stochastic

IEIIT-CNR

# Convergence Properties[1]

- Lemma (convergence properties)
- Assume that the graph is strongly connected
- Then, the scheme

$$x(k+1) = A_{\theta(k)}\ x(k)$$

achieves consensus

$$|\ x_i(k) - x_j(k)\ | \to 0 \quad \text{for} \quad k \to \infty$$

with probability one for all agents $i, j$ and for any initial condition $x(0)$

[1] A. Tahbaz-Salehi, A. Jadbabaie (2008)

IEIIT-CNR

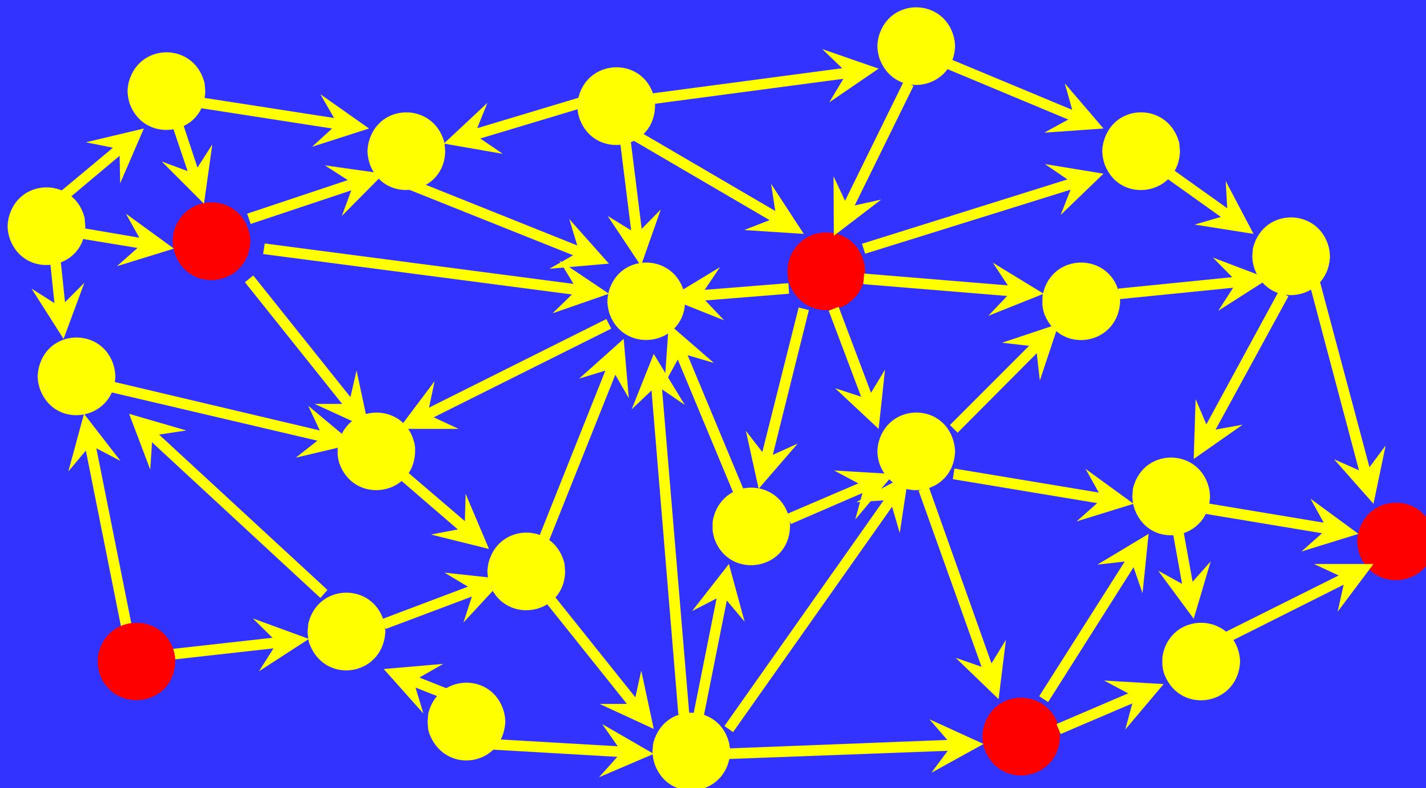
# PageRank and Consensus

| Consensus | PageRank |
|---|---|
| all agent values become equal | page values converge to constant |
| graph is strongly connected | web is not strongly connected |
| convergence w.p.1 for all $x_i$, $x_j$ $\lvert x_i(k) - x_j(k)\rvert \to 0,\ k \to \infty$ | time average MSE convergence $E[\ \lVert y(k) - x^*\rVert^2\ ] \to 0,\ \ k \to \infty$ |
| average is not necessary | average crucial for convergence |
| matrices $A_i$ are row stochastic | matrices $M_i$ are column stochastic |

# Extensions

# Simultaneous Random Updates of Multiple Pages

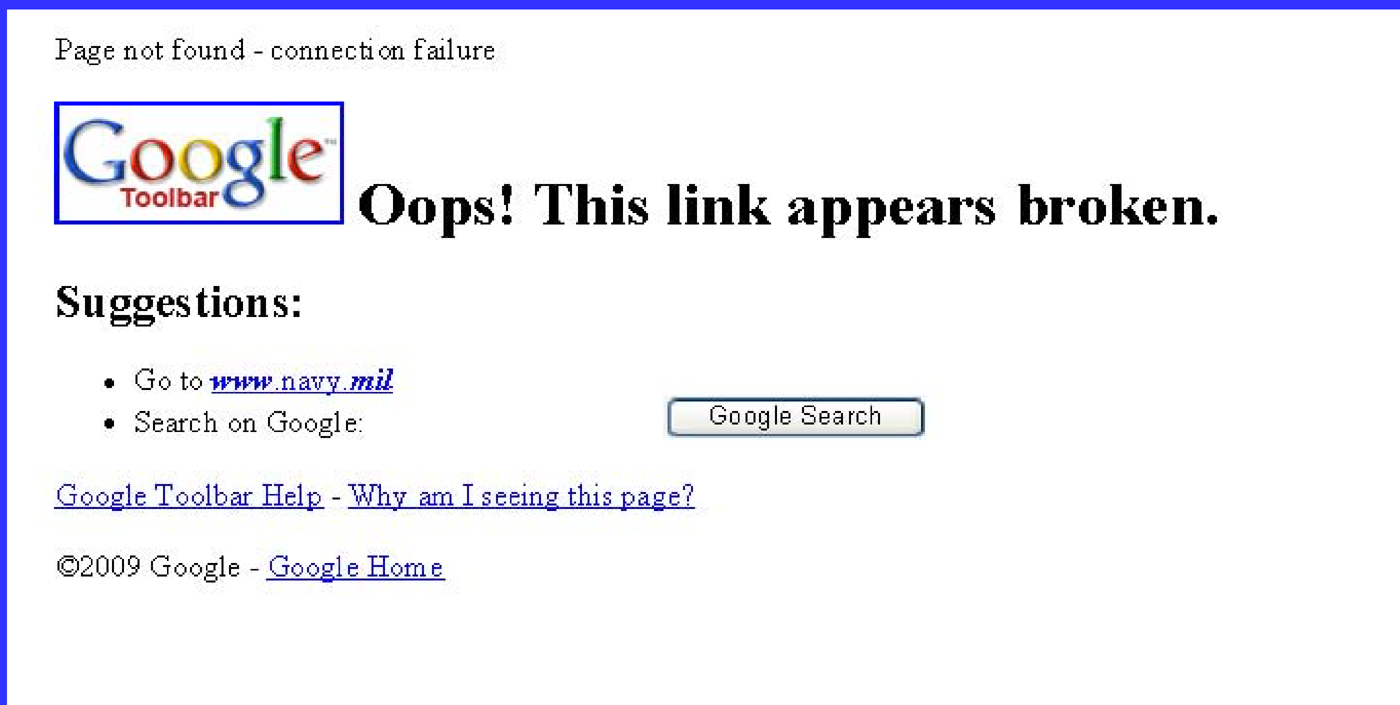
- Simultaneous random update of multiple webpages



# Robustness with Fragile Links

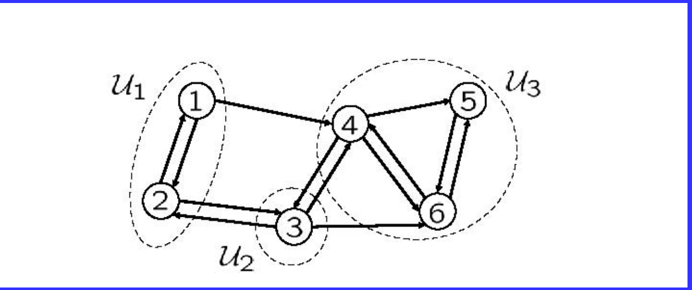
IEIIT-CNR

❖ Robustness for fragile, time-varying and broken links



# Aggregation and Clustering

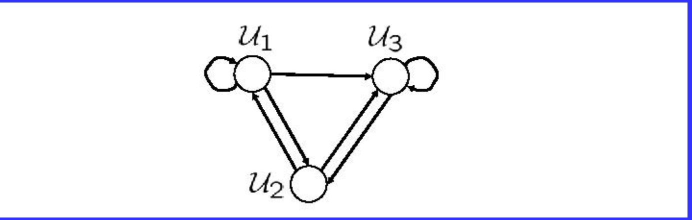
single pages do not have computational capabilities




original graph

aggregate pages into server domains (e.g. University of Tokyo)

deal only with domains at macroscopic level

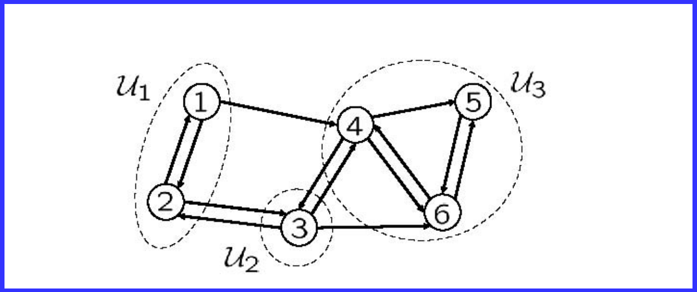



aggregated graph

# Ranking and Aggregation of Control Journals

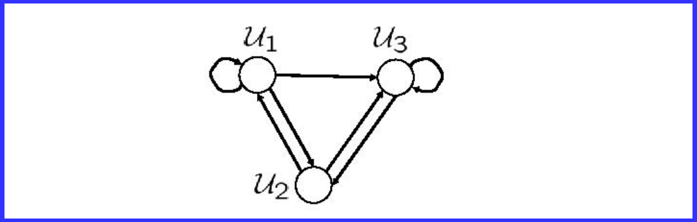
# Aggregation and Clustering

aggregate journal into categories




original graph

deal with journal categories




aggregated graph

IEIIT-CNR

# Ranking and Aggregation of Control Journals

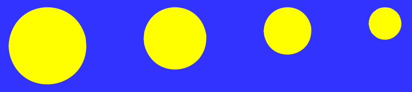
- Ranking journals in order of importance (impact factor)
- Eigenfactor is a good example of a different metric
- Visualize and aggregate journals into categories
- Construction of specific graphs with nodes and links

IEIIT-CNR

# Ranking and Aggregation of Control Journals

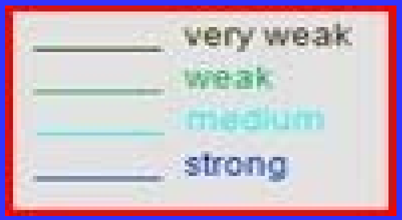
- Example: Leydesdorff citation maps
- Represent control journals by nodes

  Size of nodes given by the impact factor



- Journals are connected by undirected links

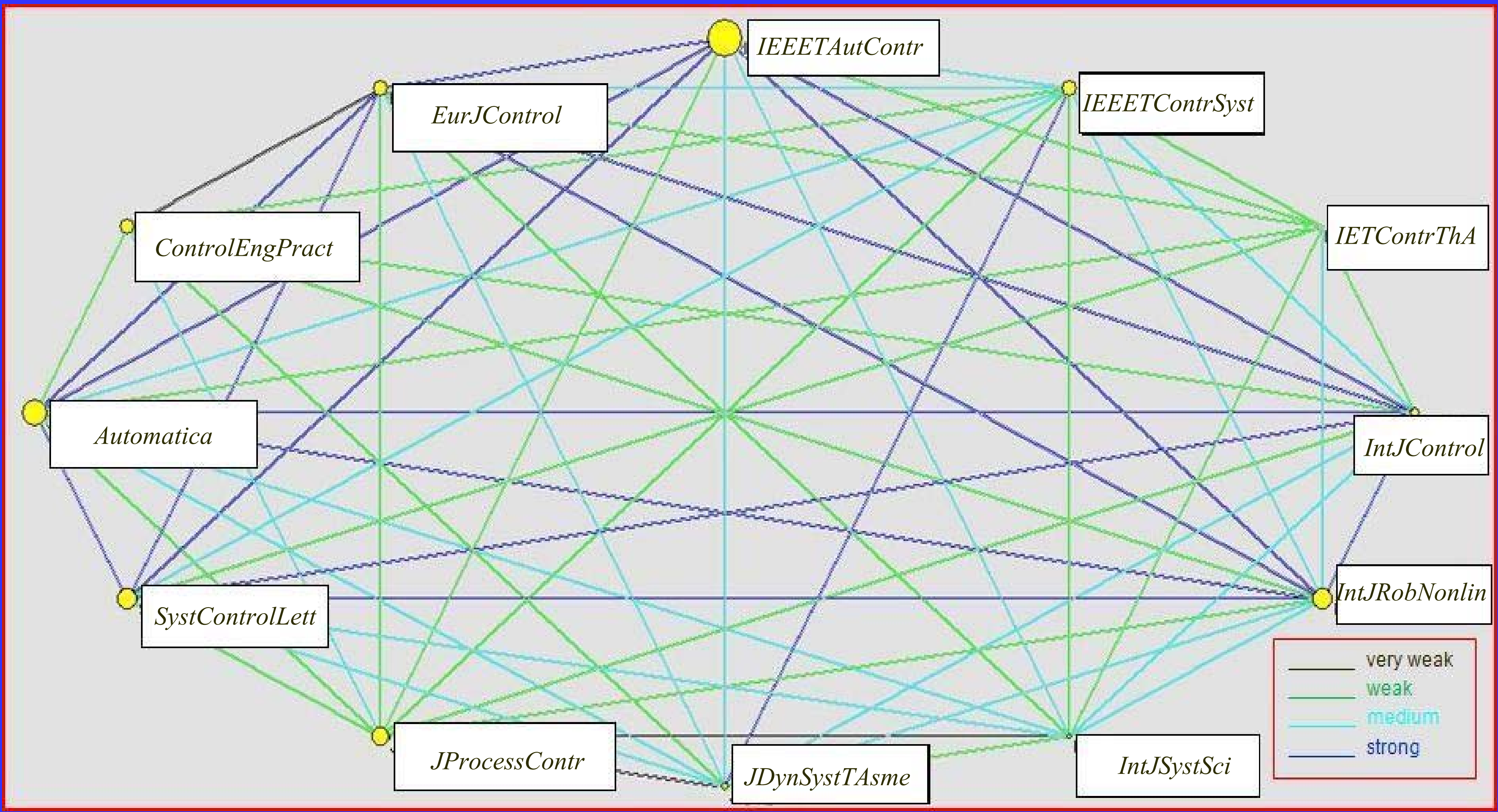
  strength of link shown by different colors




- Show the flow of information and aggregate control journals in various categories

IEIIT-CNR

# Citation Maps of Control Journals



IEIIT-CNR

# References and Software

- "*A Distributed Randomized Approach for the PageRank Computation*," H. Ishii and R. Tempo, IEEE TAC, 2010
- "*Randomized Algorithms for Analysis and Control of Uncertain Systems*", R. Tempo, G. Calafiore and F. Dabbene, Springer-Verlag, 2005
- "*Probabilistic and Randomized Tools for Control Design*," F. Dabbene and R. Tempo, The Control Handbook (W.S. Levine Ed.), *Taylor & Francis*, 2010 (to appear)
- "*RACT: Randomized Algorithms Control Toolbox*"

*http://staff.polito.it/roberto.tempo/*

IEIIT-CNR

# Acknowledgments

Acknowledgments: Research on PageRank is joint work with

- Hideaki Ishii
- Er-Wei Bai
- Fabrizio Dabbene
- Shinji Hara

*http://staff.polito.it/roberto.tempo/*

# Conclusions

- Search engines (PageRank computation in Google)
- Randomized algorithms (Las Vegas type)
- Consensus of multi-agent systems and sensor networks
- Aggregation and clustering of control journals

IEIIT-CNR

# Conclusions

- Search engines (PageRank computation in Google)
- Randomized algorithms (Las Vegas type)
- Consensus of multi-agent systems and sensor networks
- Aggregation and clustering of control journals

Learn from the PageRank problem to develop novel ideas for systems and control