



## HY463 - Συστήματα Ανάκτησης Πληροφοριών Information Retrieval (IR) Systems

# Web Searching

I: History and Basic Notions, Crawling

**II: Link Analysis Techniques**

III: Web Spam Page Identification

Γιάννης Τζιτζίκας

Διάλεξη : 9

Ημερομηνία : 25 / 4 / 2007



## Διάθρωση

- Bibliometrics
  - citation analysis, impact factor, bibliographic coupling, co-citation, citations vs links
- Authorities and Hubs (HITS algorithm)
- PageRank
  - Personalized PageRank
- Other applications of Link Analysis
  - Crawling
  - Reverse Engineering



## Ανάκτηση Πληροφοριών από τον Ιστό: Προκλήσεις και Απαιτήσεις

- Gathering techniques
- Scalable Index Structures efficiently updatable
- Improve the discrimination ability

Θα δούμε τεχνικές που συμβάλουν σε αυτό



## Bibliometrics: Citation Analysis

- Πολλά έγγραφα περιλαμβάνουν **βιβλιογραφία**, δηλαδή **μνείες (αναφορές)** σε ήδη δημοσιευμένα άρθρα.
- Θεωρώντας τις μνείες ως συνδέσμους, μπορούμε να δούμε μια συλλογή εγγράφων ως έναν διευθυνόμενο γράφο.
- Η δομή αυτού του γράφου είναι ανεξάρτητη των περιεχομένων και από αυτόν μπορούμε να εξαγάγουμε συμπεράσματα για την ομοιότητα των εγγράφων και τη δομή του χώρου.

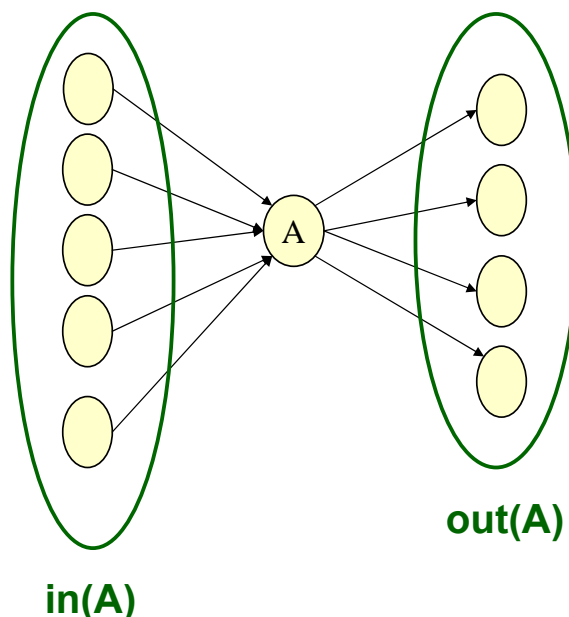


## Impact Factor (Βαθμός Επιρροής)

- **Μέτρο σπουδαιότητας** (ποιότητας, επίδρασης) των επιστημονικών περιοδικών που προτάθηκε από τον Garfield το 1972.
- Μετρά πόσο συχνά τα άρθρα του περιοδικού αναφέρονται από άλλα (μεταγενέστερα) άρθρα
  - Υπολογίζεται και δημοσιεύεται ετησίως από το Institute for Scientific Information (ISI).
- **Ο βαθμός επιρροής** ενός περιοδικού  $J$  το έτος  $Y$ 
  - είναι ο μέσος αριθμός των αναφορών σε άρθρα δημοσιευμένα στο περιοδικό  $J$  τα έτη  $Y-1$  ή  $Y-2$ , από άρθρα δημοσιευμένα σε άλλα περιοδικά το έτος  $Y$ .
  - Δεν λαμβάνει υπόψη την «ποιότητα» των άρθρων που κάνουν τις αναφορές



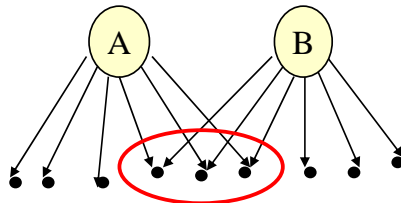
## Συμβολισμοί





## Bibliographic Coupling (Βιβλιογραφική Ζεύξη)

- **Μέτρο ομοιότητας** εγγράφων που προτάθηκε από τον Kessler το 1963
- Η **βιβλιογραφική ζεύξη** 2 εγγράφων A και B ισούται με το πλήθος των εγγράφων που αναφέρονται και από το A και από το B.
  - Το μέγεθος της τομής των βιβλιογραφιών τους
- Κανονικοποίηση βάσει του μεγέθους των βιβλιογραφιών



$$|out(A) \cap out(B)|$$

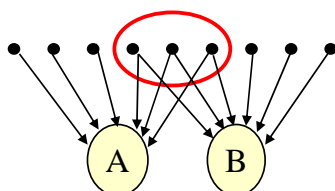
$$|out(A) \cap out(B)|$$

$$|out(A) \cup out(B)|$$



## Co-Citation

- Ένα διαφορετικό **μέτρο ομοιότητας** που προτάθηκε από τον Small το 1973
- Η **βαθμός co-citation** 2 εγγράφων A και B ισούται με το πλήθος των εγγράφων που αναφέρουν και το A και το B.
- Κανονικοποίηση βάσει του συνολικού αριθμού εγγράφων που αναφέρουν ή το A ή το B



$$|in(A) \cap in(B)|$$

$$|in(A) \cap in(B)|$$

$$|in(A) \cup in(B)|$$



## Μνείες vs. Σύνδεσμοι (Citations vs. Links)

Οι σύνδεσμοι του Ιστού είναι κάπως διαφορετικοί από τις αναφορές:

- Many links are navigational.
- Many pages with high in-degree are portals (not content providers).
- Not all links are endorsements.
- Company websites don't point to their competitors.
- Citations to relevant literature is enforced by peer-review.



## Ο Γράφος του Ιστού

Θεωρούμε τον Ιστό ως έναν διευθυνόμενο γράφο  $G=(V,E)$

- Διαγράφουμε τους κυκλικούς συνδέσμους (αυτοσυνδέσμους self-hyperlinks)
- Οι πολλαπλοί σύνδεσμοι (από μια σελίδα  $p$  σε μια  $q$ ) καταπίπτουν σε έναν σύνδεσμο  $(p,q)$  in  $E$



## Authorities (Αυθεντίες)

- *Authorities* are pages that are recognized as providing significant, trustworthy, and useful information on a topic.
- A simple measure of authority could be  $|in(p)|$
- However in-degree treats all links as equal (όπως στον βαθμό επιρροής).
- Should links from pages that are themselves authoritative count more?



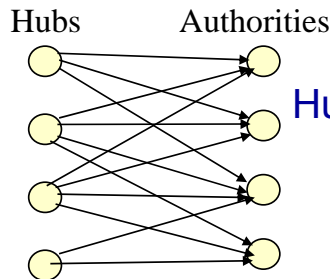
## Hubs (Κομβικά Σημεία)

- *Hubs* are index pages that provide lots of useful links to relevant content pages (topic authorities).
- Παραδείγματα Hub pages για ανάκτηση πληροφοριών:
  - <http://trec.nist.gov/>
  - <http://www-a2k.is.tokushima-u.ac.jp/member/kita/NLP/IR.html>
- A simple measure for identifying hubs could be  $|out(p)|$



## HITS (Hyperlink-Induced Topic Search)

- Αλγόριθμος που προτάθηκε από τον Kleinberg το 1998.
- Προσπαθεί να διακρίνει authorities και hubs για ένα συγκεκριμένο θέμα (topic), αναλύοντας το σχετικό υπογράφο του Ιστού.
- Βασίζεται στις εξής (αμοιβαίως οριζόμενες και αναδρομικές) προτάσεις:
  - **Hubs** point to lots of **authorities**.
  - **Authorities** are pointed to by lots of **hubs**.



Hubs and Authorities tend to form a bipartite graph  
– (nodes can be partitioned into 2 groups such that there are no links between the nodes of the same group):



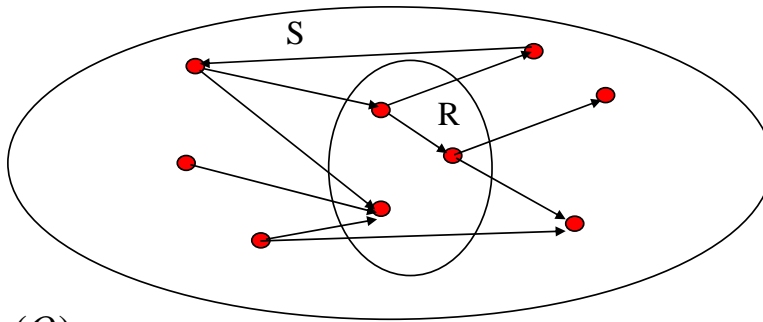
## Ο Αλγόριθμος HITS

- Εντοπίζει τα hubs και τα authorities για ένα συγκεκριμένο θέμα (topic) που προσδιορίζεται από μια επερώτηση  $q$
- Κατ' αρχάς προσδιορίζεται το σύνολο  $S$  των σχετικών σελίδων με το  $q$  και αυτό ονομάζεται βάση (base set)
- Κατόπιν, αναλύει τη δομή των συνδέσμων στον υπογράφο του ιστού που ορίζεται από το  $S_1$  και διακρίνει hubs και authorities.



## Κατασκευή του Υπογράφου Βάσης (Base Subgraph)

- For a specific query  $Q$ , let the set of documents returned by a standard search engine be called the *root set*  $R$  (i.e.  $R=Ans(Q)$ ).
- Initialize  $S$  to  $R$ .
- Add to  $S$  all pages pointed to by any page in  $R$ .
- Add to  $S$  all pages that point to any page in  $R$ .



$$R = ans(Q)$$

$$S := R \cup (\cup \{out(p) \mid p \in R\}) \cup (\cup \{in(p) \mid p \in R\})$$



## Περιορίζοντας το μέγεθος της Βάσης

- To limit computational expense:
  - Limit number of root pages to the top 200 pages retrieved for the query.
  - Limit number of “back-pointer” pages to a random set of at most 50 pages returned by a “reverse link” query.
- To eliminate purely navigational links:
  - Eliminate links between two pages on the same host.
- To eliminate “non-authority-conveying” links:
  - Allow only  $m$  ( $m \cong 4-8$ ) pages from a given host as pointers to any individual page.





## Authorities and In-Degree

- Even within the base set  $S$  for a given query, the nodes with highest in-degree are not necessarily authorities (may just be generally popular pages like Yahoo or Amazon).
- True authority pages are pointed to by a number of hubs (i.e. pages that point to lots of authorities).



## HITS: Επαναληπτικός αλγόριθμος

- Use an **iterative** algorithm to slowly converge on a mutually reinforcing set of hubs and authorities.
- Maintain for each page  $p \in S$ :
  - Authority score:  $\mathbf{a}(p)$  (vector  $\mathbf{a}$ )
  - Hub score:  $\mathbf{h}(p)$  (vector  $\mathbf{h}$ )
- Initialize all  $a(p)=h(p) = 1$
- Maintain normalized scores:

$$\sum_{p \in S} a(p)^2 = 1 \qquad \sum_{p \in S} h(p)^2 = 1$$



## HITS: Κανόνες Ενημέρωσης (Update Rules)

- Authorities are pointed to by lots of good hubs:

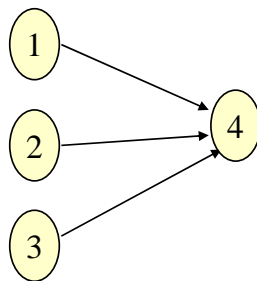
$$a(p) = \sum_{q \in \text{in}(p)} h(q)$$

- Hubs point to lots of good authorities:

$$h(p) = \sum_{q \in \text{out}(p)} a(q)$$

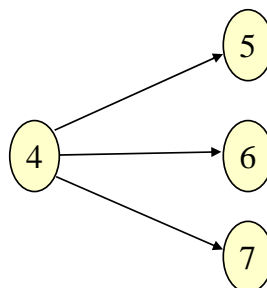


## Παράδειγμα Κανόνων Ενημέρωσης



$$a(4) = h(1) + h(2) + h(3)$$

$$h(4) = a(5) + a(6) + a(7)$$





## HITS: Επαναληπτικός Αλγόριθμος

Initialize for all  $p \in S$ :  $a(p)=h(p) = 1$

For  $i = 1$  to  $k$ :

For all  $p \in S$ : *(update auth. scores)*

$$a(p) = \sum_{q \in \text{in}(p)} h(q)$$

For all  $p \in S$ : *(update hub scores)*

$$h(p) = \sum_{q \in \text{out}(p)} a(q)$$

For all  $p \in S$ :

$$a(p) = a(p)/c \quad c = \sum_{p \in S} a(p)^2 \quad \text{(normalize } \mathbf{a} \text{)}$$

For all  $p \in S$ :

$$h(p) = h(p)/c \quad c = \sum_{p \in S} h(p)^2 \quad \text{(normalize } \mathbf{h} \text{)}$$



## HITS: Σύγκλιση

- Με άπειρες επαναλήψεις ο αλγόριθμος συγκλίνει σε ένα σταθερό σημείο (*fix-point*).
- Define  $A$  to be the adjacency matrix for the subgraph defined by  $S$ .
  - $A_{ij} = 1$  for  $i \in S, j \in S$  iff  $i \rightarrow j$
- Authority vector,  $\mathbf{a}$ , converges to the principal eigenvector of  $A^T A$
- Hub vector,  $\mathbf{h}$ , converges to the principal eigenvector of  $A A^T$
- Στην πράξη, 20 επαναλήψεις συνήθως επαρκούν.



## HITS: Αποτελέσματα

- Authorities for query: “Java”
  - java.sun.com
  - comp.lang.java FAQ
- Authorities for query “search engine”
  - Yahoo.com
  - Excite.com
  - Lycos.com
  - Altavista.com
- Authorities for query “Gates”
  - Microsoft.com
  - roadahead.com

- Σχόλια

- In most cases, the final authorities were not in the initial root set generated using Altavista.
- Authorities were brought in from linked and reverse-linked pages and then HITS computed their high authority score.



## Εύρεση παρόμοιων σελίδων αξιοποιώντας τη δομή συνδέσμων

- Given a page  $p$ , let  $R$  (the root set) be  $k$  (e.g. 200) pages that point to  $p$  ( $\approx R=in(p)$ )
- Grow a base set  $S$  from  $R$ .
- Run HITS on  $S$ .
- Return the best authorities in  $S$  as the best similar-pages for  $p$ .
  - θυμηθείτε το co-citation
- Finds authorities in the “link neighbor-hood” of  $p$ .

- Αποτελέσματα για “honda.com”

- toyota.com
- ford.com
- bmwusa.com
- saturncars.com
- nissanmotors.com
- audi.com
- volvocars.com



## PageRank

- Μια διαφορετική τεχνική ανάλυσης συνδέσμων που χρησιμοποιείται από το Google (Brin & Page, 1998).
- Δεν κάνει διάκριση μεταξύ αυθεντιών και κομβικών σημείων
- Διατάσσει τις σελίδες βάσει κύρους (authority).
- Εφαρμόζεται σε όλες τις σελίδες του ιστού (δεν περιορίζεται στη γειτονιά των σελίδων της απάντησης μιας επερώτησης)



## PageRank: Η αρχική έκδοση

- Just measuring in-degree (citation count) doesn't account for the authority of the source of a link.
- Initial page rank equation for page  $p$ :

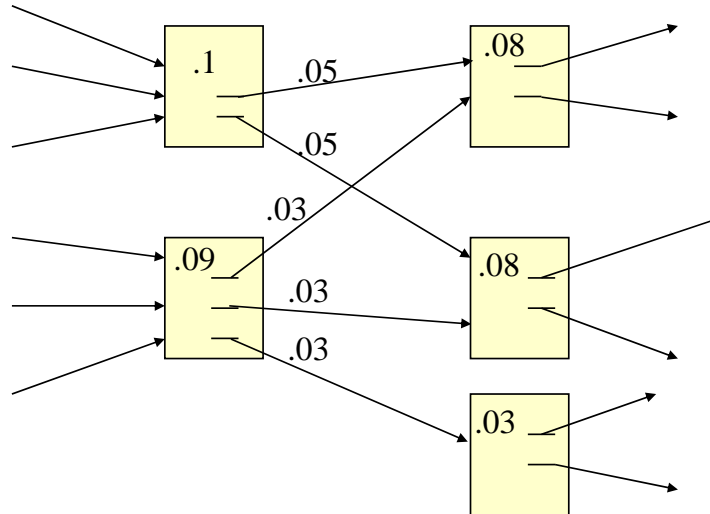
$$R(p) = c \sum_{q \in \text{in}(p)} \frac{R(q)}{|\text{out}(q)|}$$

- A page  $q$ , "gives" an equal fraction of its authority to all the pages it points to (e.g.  $p$ ).
- $c$  is a normalizing constant set so that the rank of all pages always sums to 1.



## PageRank: Η αρχική έκδοση (II)

- Can view it as a process of PageRank “flowing” from pages to the pages they cite.



## PageRank: Ο Αρχικός Αλγόριθμος

- Iterate rank-flowing process until convergence:

Let  $S$  be the total set of pages.

Initialize  $\forall p \in S: R(p) = 1/|S|$

Until ranks do not change (much) (*convergence*)

For each  $p \in S$ :

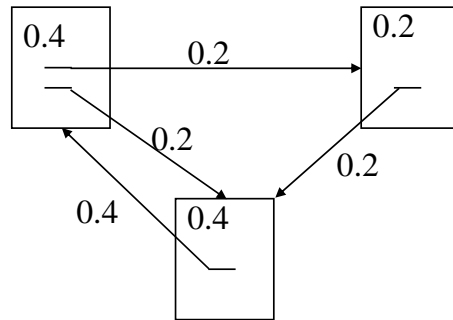
$$R'(p) = \sum_{q \in \text{in}(p)} \frac{R(q)}{|\text{out}(q)|}$$

For each  $p \in S: R(p) = R'(p)/c$  (*normalize*)

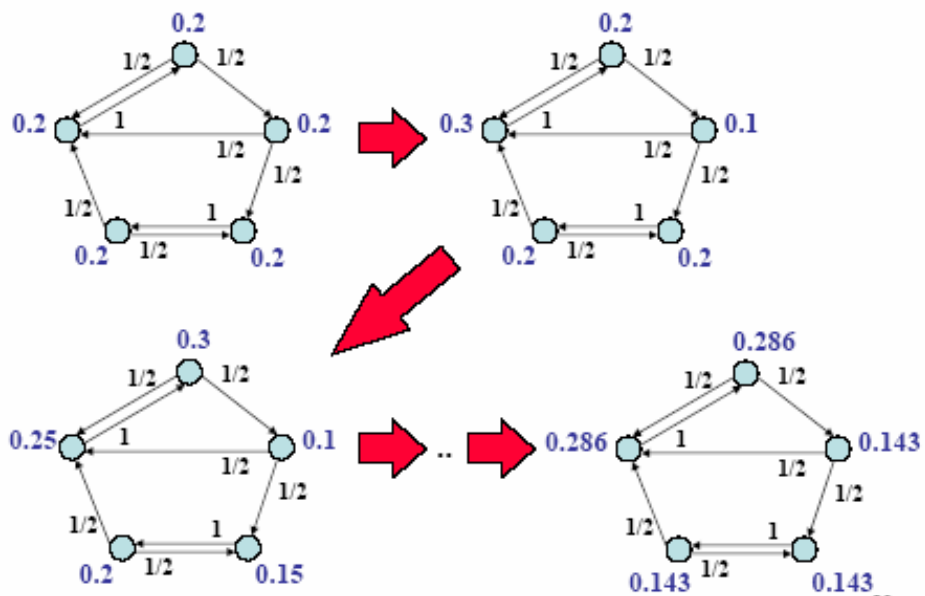
$$c = \sum_{p \in S} R'(p)$$



# Sample Stable Fixpoint



# Παράδειγμα Επαναλήψεων



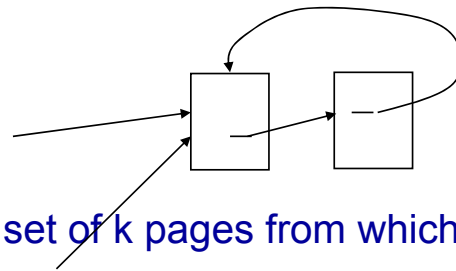


## Random Surfer Model (Μοντέλο Τυχαίου Περιηγητή)

- PageRank can be seen as modeling a “random surfer” that
  - starts on a random page and then at each point:
  - **randomly follows a link on the current page.**
- $R(p)$  models the probability that this random surfer will be on page  $p$  at any given time.



## Οι αδυναμίες της αρχικής έκδοσης: Rank Sinks and Rank Leaks

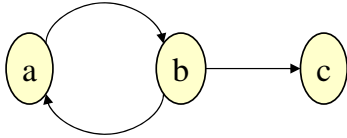


- **Rank sink**: any strongly connected set of  $k$  pages from which no links point outwards
  - problem: nodes not in the sink receive 0 rank
    - a random surfer would enclave for ever within the sink
- **Rank leak**: any individual page with no outgoing link
  - any rank reaching a rank leak is lost forever
    - will cause all the ranks to eventually converge to 0
- Rank leak is a special case of Rank sink (for  $k=1$ )





## Rank Leak: Παράδειγμα



a	b	c
0.3	0.3	0.3
0.15	0.3	0.15
0.15	0.15	0.15
0.075	0.15	0.075
0.075	0.075	0.075
0.0375	0.075	0.0375
0.0375	0.0375	0.0375
0.01875	0.0375	0.01875
0.01875	0.01875	0.01875
0.009375	0.01875	0.009375
0.009375	0.009375	0.009375
0.004688	0.009375	0.004688
0.004688	0.004688	0.004688
0.002344	0.004688	0.002344
0.002344	0.002344	0.002344
0.001172	0.002344	0.001172
0.001172	0.001172	0.001172
0.000586	0.001172	0.000586
0.000586	0.000586	0.000586
0.000293	0.000586	0.000293



## Τρόποι Αντιμετώπισης

- **Leak nodes:**
  - Απαλοιφή όλων των leak nodes (those with out-degree 0)
  - Υπόθεση ότι κάθε leak node έχει έναν σύνδεσμο προς κάθε άλλη σελίδα
- **Sink nodes**
  - .... “teleporting”

$$R(p) = c \left( \sum_{q \in in(p)} \frac{R(q)}{|out(q)|} + E(p) \right)$$



## Αναθεωρώντας το Μοντέλο του Τυχαίου Περιηγητή

- PageRank can be seen as modeling a “random surfer” that
    - starts on a random page and then at each point:
    - **with probability  $E(p)$  randomly jumps to page  $p$ .**
    - otherwise, randomly follows a link on the current page.
  - $R(p)$  models the probability that this random surfer will be on page  $p$  at any given time.
- // “E jumps” are needed to prevent the random surfer from getting “trapped” in web sinks with no outgoing links.



## Ο αλγόριθμος PageRank

Let  $S$  be the total set of pages.

Let  $\forall p \in S: E(p) = \alpha/|S|$  (for some  $0 < \alpha < 1$ , e.g. 0.15)

Initialize  $\forall p \in S: R(p) = 1/|S|$

Until ranks do not change (much) (*convergence*)

For each  $p \in S$ :

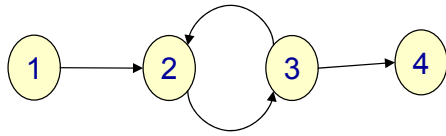
$$R'(p) = \sum_{q \in \text{in}(p)} \frac{R(q)}{|\text{out}(q)|} + E(p)$$

For each  $p \in S: R(p) = R'(p)/c$  (*normalize*)

$$c = \sum_{p \in S} R'(p)$$



# PageRank: Διατύπωση με Γραμμική Αλγεβρα



Adjacency matrix M

$$M = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Transition matrix T

$$T(p, q) = \begin{cases} 0 & \text{if } (q, p) \notin M \\ 1/|\text{out}(q)| & \text{if } (q, p) \in M \end{cases}$$

$$T = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1/2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/2 & 0 \end{pmatrix}$$

- The PageRank score  $R(p)$  of a page is defined as

$$R(p) = a \cdot \sum_{q \in \text{in}(p)} \frac{R(q)}{|\text{out}(q)|} + (1-a) \frac{1}{N}$$

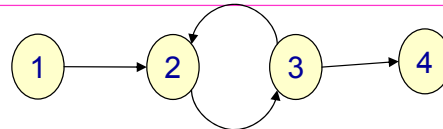
- The equivalent matrix equation:

$$R = a \cdot T \cdot R + (1-a) \frac{1}{N} \mathbf{1}_N$$



# PageRank: Διατύπωση με Γραμμική Αλγεβρα

$$R = a \cdot T \cdot R + (1-a) \frac{1}{N} \mathbf{1}_N$$



$$\begin{bmatrix} r1 \\ r2 \\ r3 \\ r4 \end{bmatrix} = a \cdot \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1/2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/2 & 0 \end{bmatrix} \cdot \begin{bmatrix} r1 \\ r2 \\ r3 \\ r4 \end{bmatrix} + (1-a) \frac{1}{4} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} r1 \\ r2 \\ r3 \\ r4 \end{bmatrix} = a \cdot \begin{bmatrix} 0 \\ r1+r3/2 \\ r2 \\ r3/2 \end{bmatrix} + (1-a) \frac{1}{4} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} r1 \\ r2 \\ r3 \\ r4 \end{bmatrix} = \begin{bmatrix} (1-a)/4 \\ a(r1+r3/2)+(1-a)/4 \\ ar2+(1-a)/4 \\ ar3/2+(1-a)/4 \end{bmatrix}$$



## Ο Αλγόριθμος PageRank

```
function PageRank
  Input   T: transition matrix,      N: number of pages,
          ab: decay factor for PageRank,  Mb: number of iterations
  output  R* : PageRank scores

  (1) d = 1/N * 1N           // initial score for all pages is 1/N
  (2) R* = d
  (3) for i=1 to Mb do         // evaluates PageRank scores
      R* = ab T R* + (1 - ab) d
  return R*
```



## PageRank: Ταχύτητα σύγκλισης (Speed of Convergence)

- Early experiments on Google used 322 million links.
- PageRank algorithm converged (within small tolerance) in about 52 iterations.
- Number of iterations required for convergence is empirically  $O(\log n)$  (where  $n$  is the number of links).
- Therefore calculation is quite efficient.



## Personalized PageRank

- Μπορούμε να εξατομικεύσουμε / προκαταβάλουμε το PageRank, τροποποιώντας κατάλληλα το **E**
  - (ώστε να μην περιγράφει μια ομοιόμορφη κατανομή)
- Για παράδειγμα, με τον τρόπο αυτό μπορούμε να περιορίσουμε τα «τυχαία άλματα» σε ένα συγκεκριμένο σύνολο σελίδων

Παράδειγμα:

Αν  $p = \text{www.csd.uoc.gr/~hy463}$  τότε  $E(p) = \alpha$  αλλιώς  $E(p) = 0$

// ευνοεί τις ιστοσελίδες που είναι κοντά (στο γράφο) στην ιστοσελίδα

// του μαθήματος



## Simple Title Search with PageRank (Google Ranking)



- Use simple Boolean search to search web-page titles and rank the retrieved pages by their PageRank.
- Sample search for “university”:
  - Altavista returned a random set of pages with “university” in the title (seemed to prefer short URLs).
  - Primitive Google returned the home pages of top universities.
- Complete Google ranking includes (based on university publications prior to commercialization).
  - Vector-space similarity component.
  - Keyword proximity component.
  - HTML-tag weight component (e.g. title preference).
  - PageRank component.
- Details of current commercial ranking functions are trade secrets



## Ανάλυση Συνδέσμων: Συμπεράσματα

- Η Ανάλυση συνδέσμων αξιοποιεί τη δομή του γράφου του Ιστού προκειμένου να βοηθήσει την ανάκτηση πληροφοριών
- Είναι ίσως η μεγαλύτερη καινοτομία στην αναζήτηση στον Ιστό
- Ο βασικό ατού της επιτυχίας του Google.



## Άλλες Εφαρμογές του PageRank: Crawling/Spidering

- Αξιοποίηση του PageRank για εστίαση της διάσχισης στις «σημαντικές σελίδες»

### Τρόπος

- Υπολογισμός του PageRank βάσει των σελίδων που έχουν ήδη συλλεχθεί
- Ταξινόμηση των σελίδων στην ουρά του crawler βάσει του εκτιμώμενου PageRank.



## Ανάλυση Συνδέσμων: Άλλες εφαρμογές

- **Αναγνώριση κοινοτήτων (communities)**
  - Έχει παρατηρηθεί ότι κάθε κοινότητα χαρακτηρίζεται από ένα σύνολο authority και hub σελίδων
- **Αναγνώριση σελίδων “spam”** (θα παρουσιαστεί στην επόμενη διάλεξη)
  - Web-spam page identification
- **Κατανόηση και Οπτικοποίηση μεγάλων Εννοιολογικών Σχημάτων**
- **Node Reputability in P2P Networks**
- ...



## SALSA (Stochastic Approach for Link-Structured Analysis)

- Ο αλγόριθμος SALSA, όπως συμβαίνει και με τον HITS, διατάσει τις σελίδες μια απάντησης βάσει των υπερσυνδέσμων και στην διάκριση authority και hub σελίδων.
- Η διαφοροποίηση του από το HITS εντοπίζεται στα εξής :
  - καταφέρνει να αναγνωρίσει και να ανιχνεύσει περισσότερες σελίδες ως authorities, σε θεματικές ομάδες εγγράφων όπου το HITS αδυνατεί.
  - θεωρεί λιγότερο στενή τη σχέση ανάμεσα στις authority και hub σελίδες