Application of the Google matrix methods for characterization of directed networks

Laboratoire de Physique Théorique de Toulouse - 13 October 2014

Vivek Kandiah

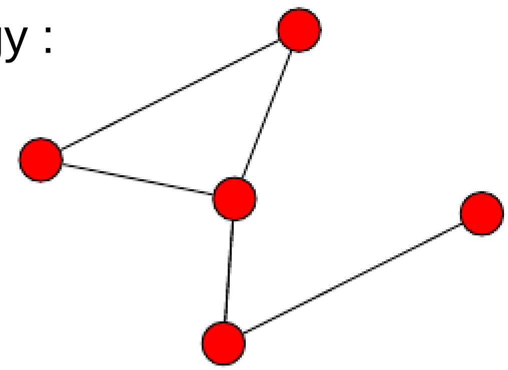
Supervisors : Bertrand Georgeot and Dima Shepelyansky

Introduction – Networks/Graphs

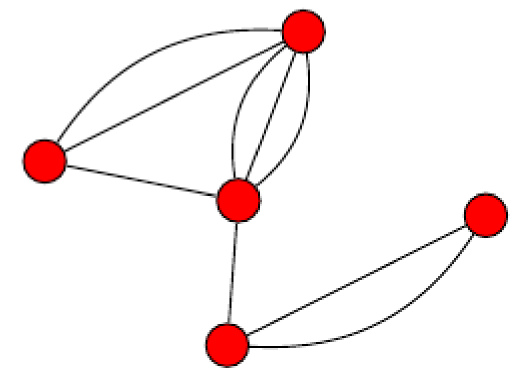
Physics (math) notation and terminology :

G : network (graph)
V : node (vertex)
E : link (edge)

A graph's formal notation : G=(V,E)



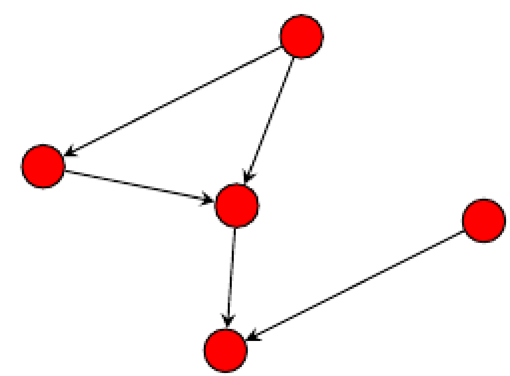
Weighted graph



Characteristic quantity ? $\boxed{\text{Degree distribution : } p(k)}$

Probability that a randomly picked node has k connections

Directed graph



For directed networks ?

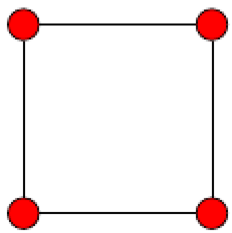
In-degree distribution : $p^{in}(k_{in})$

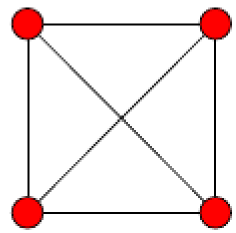
Out-degree distribution : $p^{out}(k_{out})$

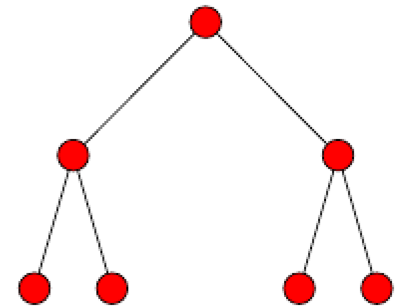
In this work : we consider networks with a fixed number of nodes **N** and a fixed number of links **L**

Introduction - Network Science

Mathematician derived rigorous results about several simplified graphs structure



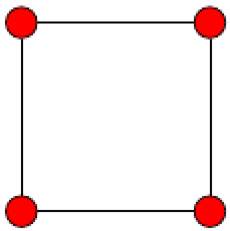
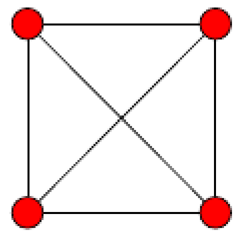
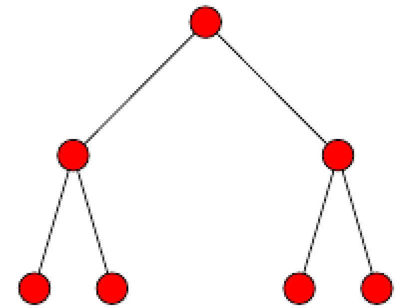




...

Introduction - Network Science

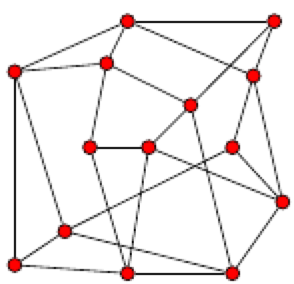
Mathematician derived rigorous results about several simplified graphs structure

...

~1960s : Paul Erdös and Alfréd Rényi, random graph models (RGM). These models are an ensemble of all possible graphs with specific constraints.

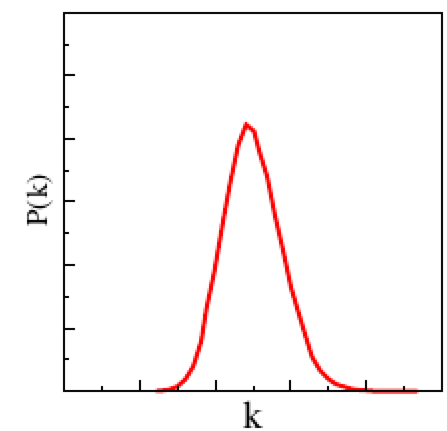
Ex : in G(n,p) model, there are n vertices and each edge exists with probability p independently from other edges



Large N

Poisson
degree distribution

$$p(k) = \frac{\bar{k}^k}{k!} e^{-\bar{k}}$$



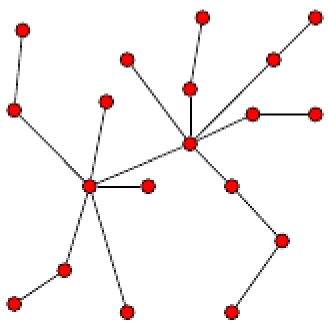
Scale : average degree

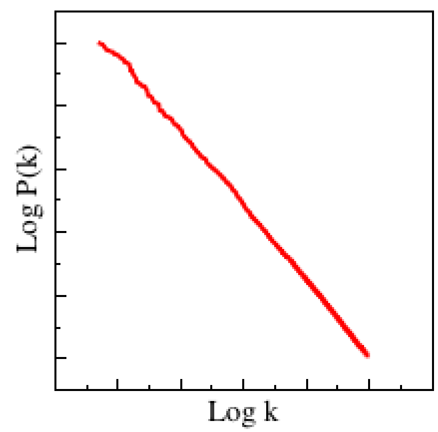
Introduction - Network Science

~1990s : Empirical observations, degree distribution is not Poissonian !

Introduction - Network Science

~1990s : Empirical observations, degree distribution is not Poissonian !





Power law
Degree distribution

$p(k) \propto k^{-\gamma}$

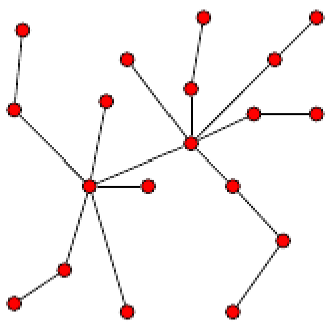
Typically : $2 \leq \gamma \leq 3$

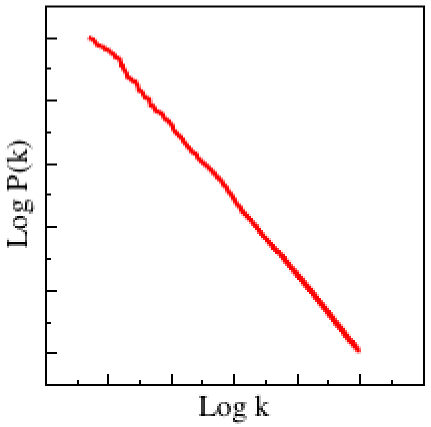
Scale-free → Hubs !

Introduction - Network Science

~1990s : Empirical observations, degree distribution is not Poissonian !

~1999s : Barabási-Albert model (preferential attachment model) suggested a mechanism to the appearance of scale-free networks in real systems





Power law
Degree distribution
$$p(k) \propto k^{-\gamma}$$

Typically : $2 \leq \gamma \leq 3$

Scale-free → Hubs !

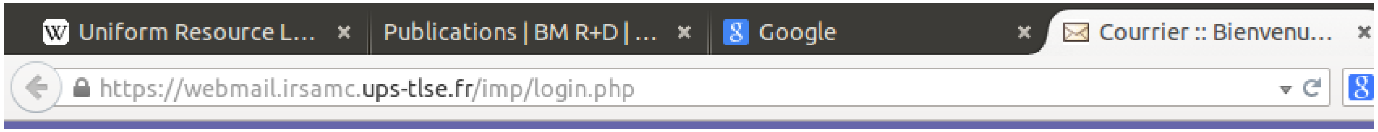
New network structure → new behaviour → new questions and researches

How to find/detect hubs or important nodes ? To what extent are they important ? ...

Introduction - Information Technology

~1937 : Turing machine concept

~1981 : First PC by IBM



~1990s : URL protocol

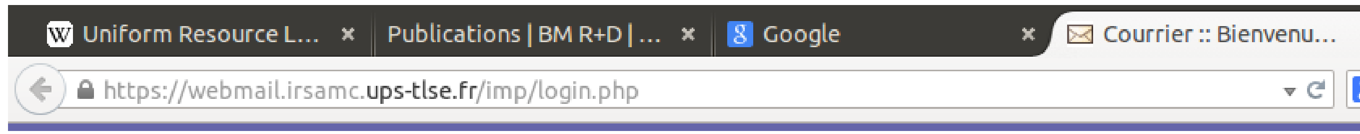
Great success : ~ 48 x 10^9 indexed webpages by Google inc. (end of 2013)

Introduction - Information Technology

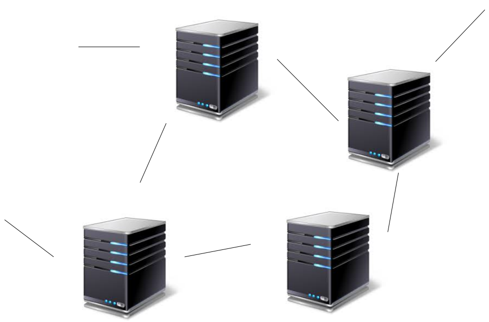
~1937 : Turing machine concept

~1981 : First PC by IBM

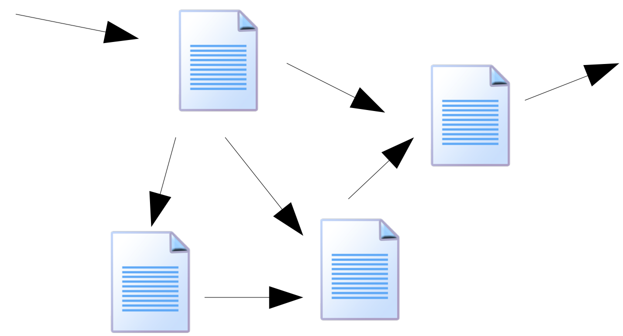
~1990s : URL protocol



Great success : ~ 48 x 10^9 indexed webpages by Google inc. (end of 2013)



Internet : physical network and undirected

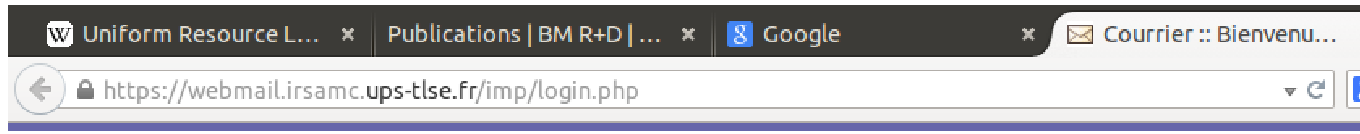


World Wide Web (WWW) : virtual network and directed

Introduction - Information Technology

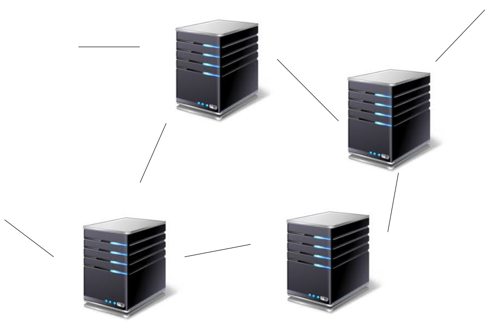
~1937 : Turing machine concept

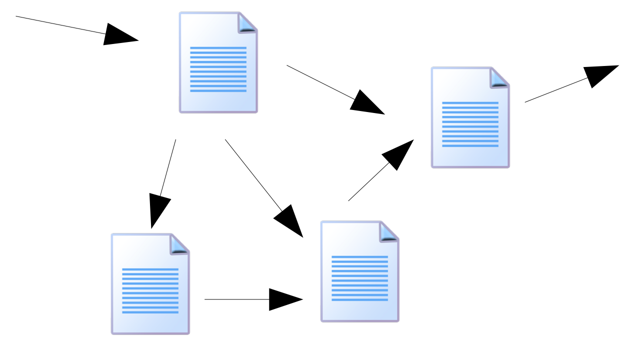
~1981 : First PC by IBM



~1990s : URL protocol

Great success : ~ 48×10^9 indexed webpages by Google inc. (end of 2013)





Internet : physical network and undirected

World Wide Web (WWW) : virtual network and directed

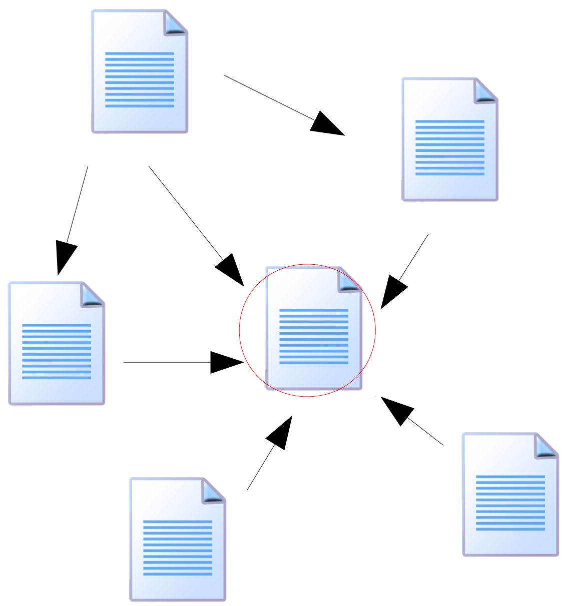
WWW is unorganized : How do we retrieve information ? Search Engines !

Search engines are automated programs working in 2 steps : first they collect the information on the network and second they provide a ranking of relevant pages to the user

First attempts were unsuccessful $\longrightarrow$ Need for a new approach for a better ranking

Introduction - Information Technology

~1995/1996 : Sergey Brin and Larry Page : new approach (through the viewpoint of recommendation)



- A site having many incoming links is important
- A site having many outgoing links gives lower scores to whom he points to
- A site is important if it is pointed by important sites

How to find the hub ? … Is there an unambiguous way to score and rank the nodes ?

PageRank : $p(i) = \sum_{j \in B_i} \frac{p(j)}{|j|}$

Self-coherent formula of PageRank score.

(sites j belongs to the set of sites pointing to i and | j | is the number of outgoing links of j)

Theory – Google Matrix

Transform the self-coherent formula into an iterative one (analogy with equilibrium solution)

$$p(i) = \sum_{j \in B_i} \frac{p(j)}{|j|} \longrightarrow p_{t+1}(i) = \sum_{j \in B_i} \frac{p_t(j)}{|j|}$$

Theory – Google Matrix

Transform the self-coherent formula into an iterative one (analogy with equilibrium solution)

$$p(i) = \sum_{j \in B_i} \frac{p(j)}{|j|} \longrightarrow p_{t+1}(i) = \sum_{j \in B_i} \frac{p_t(j)}{|j|}$$

- Does the stationary solution **p** exist ?
- Do we converge to **p** from any initial distribution ?

(**p** : vector of scores i.e PageRank)

Can a random surfer explore the network continuously without being trapped ?

There are traps, to understand and remove them we switch to matrix representation

The random surfer image is related to the Markov chain theory which can be described by matrices

Theory – Google Matrix

Transform the self-coherent formula into an iterative one (analogy with equilibrium solution)

$$p(i) = \sum_{j \in B_i} \frac{p(j)}{|j|} \longrightarrow p_{t+1}(i) = \sum_{j \in B_i} \frac{p_t(j)}{|j|}$$

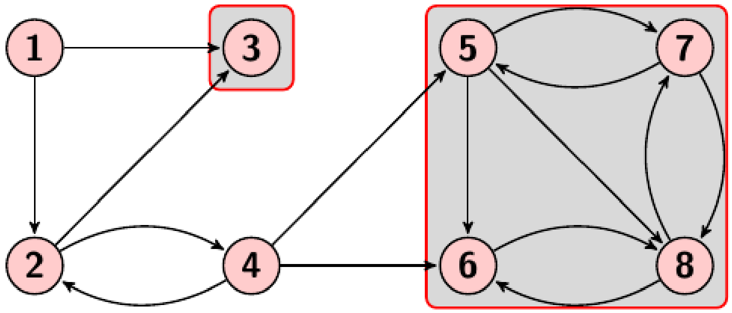
- Does the stationary solution **p** exist ?
- Do we converge to **p** from any initial distribution ?

(**p** : vector of scores i.e PageRank)

Can a random surfer explore the network continuously without being trapped ?

There are traps, to understand and remove them we switch to matrix representation

The random surfer image is related to the Markov chain theory which can be described by matrices



$$A = \begin{cases} m \text{ if } j \rightarrow i \text{ (m times).} \\ 0 \text{ otherwise.} \end{cases}$$

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

Theory – Google Matrix

Column normalization :

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} \longrightarrow A' = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 1/3 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/3 & 1 & 1/2 & 0 \end{pmatrix}$$

- Outgoing flows are treated equally
- Transition probabilities

Theory – Google Matrix

Column normalization :

$$A = \begin{pmatrix} 0&0&0&0&0&0&0&0\\ 1&0&0&1&0&0&0&0\\ 1&1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&0&1&0&0&1&0\\ 0&0&0&1&1&0&0&1\\ 0&0&0&0&1&0&0&1\\ 0&0&0&0&1&1&1&0 \end{pmatrix} \longrightarrow A' = \begin{pmatrix} 0&0&0&0&0&0&0&0\\ 1/2&0&0&1/3&0&0&0&0\\ 1/2&1/2&0&0&0&0&0&0\\ 0&1/2&0&0&0&0&0&0\\ 0&0&0&1/3&0&0&1/2&0\\ 0&0&0&1/3&1/3&0&0&1/2\\ 0&0&0&0&1/3&0&0&1/2\\ 0&0&0&0&1/3&1&1/2&0 \end{pmatrix}$$

- Outgoing flows are treated equally
- Transition probabilities

Removing traps

Dangling nodes : Nodes that have no outgoing links

$$S = \begin{pmatrix} 0&0&1/8&0&0&0&0&0\\ 1/2&0&1/8&1/3&0&0&0&0\\ 1/2&1/2&1/8&0&0&0&0&0\\ 0&1/2&1/8&0&0&0&0&0\\ 0&0&1/8&1/3&0&0&1/2&0\\ 0&0&1/8&1/3&1/3&0&0&1/2\\ 0&0&1/8&0&1/3&0&0&1/2\\ 0&0&1/8&0&1/3&1&1/2&0 \end{pmatrix}$$

- Ensures stochasticity
- Virtual links from dangling node to rest of the network

Dangling groups : Subgroup of nodes connected between themselves but not to the rest

$$G = \alpha S + (1-\alpha)\frac{1}{N}\mathbf{e}\mathbf{e}^T$$

$$G = \begin{pmatrix} 1/40&1/40&1/8&1/40&1/40&1/40&1/40&1/40\\ 17/40&1/40&1/8&7/24&1/40&1/40&1/40&1/40\\ 17/40&17/40&1/8&1/40&1/40&1/40&1/40&1/40\\ 1/40&17/40&1/8&1/40&1/40&1/40&1/40&1/40\\ 1/40&1/40&1/8&7/24&1/40&1/40&17/40&1/40\\ 1/40&1/40&1/8&7/24&7/24&1/40&1/40&17/40\\ 1/40&1/40&1/8&1/40&7/24&1/40&1/40&17/40\\ 1/40&1/40&1/8&1/40&7/24&33/40&17/40&1/40 \end{pmatrix}$$

- Ensures primitivity
- Google : α = 0.85

Theory – Google Matrix

The Google matrix : $G = \alpha S + (1-\alpha)\frac{1}{N}\mathbf{e}\mathbf{e}^T$ (with damping factor $0 \leq \alpha \leq 1$)

G stochastic ⟶ Spectral radius r =1 is an eigenvalue of G

G primitive ⟶ Perron-Frobenius Theorem can be applied to G

Let G be a primitive matrix

- The spectral radius r of G is a simple eigenvalue of G.
- r is the only eigenvalue on the spectral circle of G.
- There is a unique real eigenvector $\mathbf{v}$ such that $G\mathbf{v} = r\mathbf{v}$ and $v_i > 0 \quad \forall i$.

A positive eigenvector of Google matrix G at r =1 exists, it is computed as $G\mathbf{p} = \mathbf{p}$

Unique and can be sorted and has the meaning of a probability distribution over the nodes when normalized as $\sum_i p_i = 1$

Ex : $p^T = (0.0318, 0.0594, 0.0683, 0.0556, 0.1187, 0.1948, 0.1800, 0.2914)$

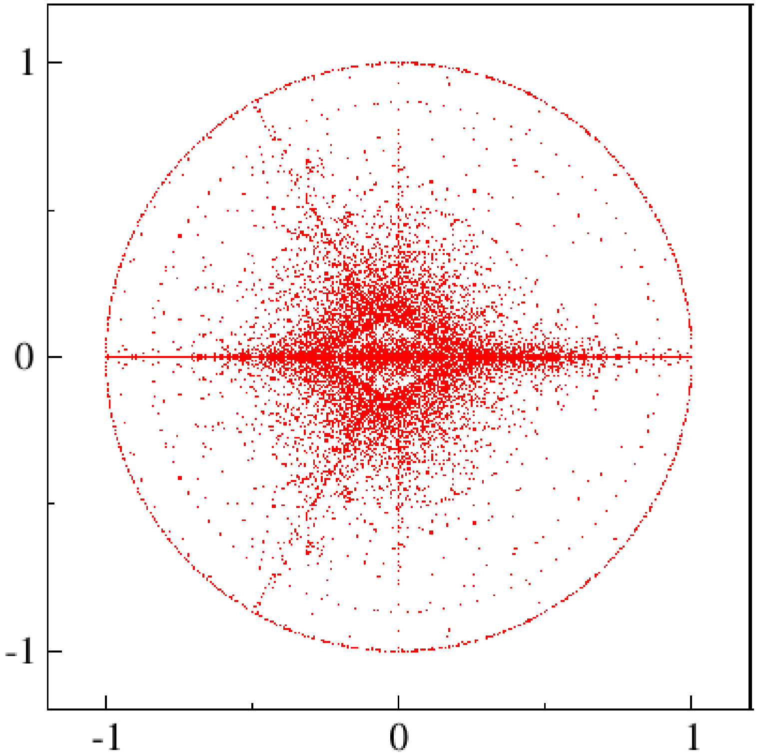
$\sigma = (8, 7, 6, 5, 3, 2, 4, 1)$

Rank index **K** (i.e top rank denoted by K=1, next to top by K=2,...)

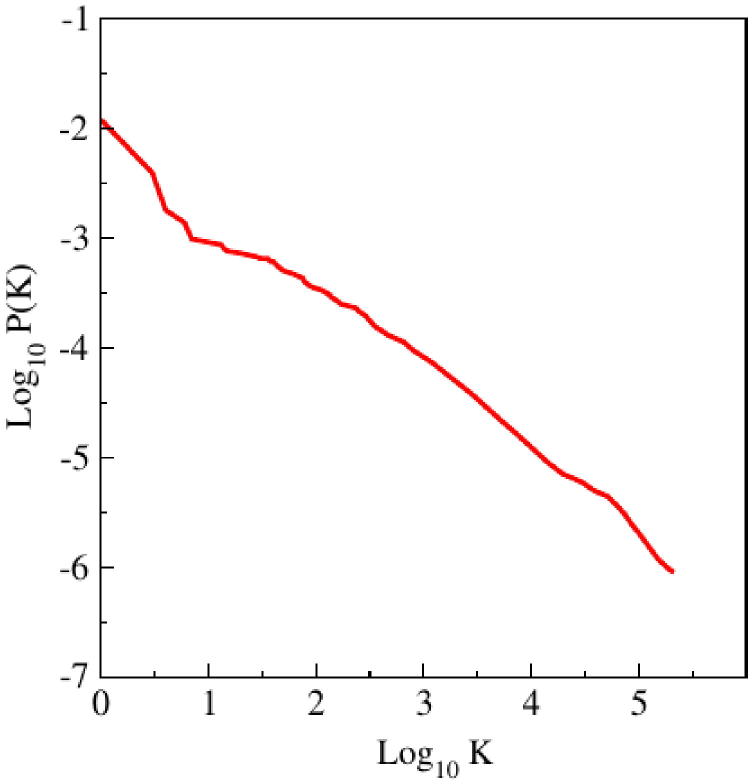
Theory – Spectrum and PageRank properties

Cambridge webpages network : N ~ 2 x 10^5 and L ~ 2 x 10^6 *(Frahm et al., 2014)*

Spectrum of G at α = 1



PageRank distribution decay



Observations :

$$P(K) \sim \frac{1}{K^\beta}$$

WWW : $\beta \approx 0.9$

and

$$\beta = \frac{1}{\mu - 1}$$

where

$$p^{in}(k_{in}) \sim 1/k_{in}^{\mu}$$

Introducing a damping factor α < 1 ⟶ gap between r = 1 and other eigenvalues

⟶ Facilitates the numerical iterative computation of PageRank $G\mathbf{p} = \mathbf{p}$

Application

- PageRank : well studied in WWW context,
  efficient (in large scale-free networks) and easy to compute
- What about other eigenvalues and eigenvectors properties ?
- What about systems other than WWW ?

Aim of the Thesis : Explore the use of this method to various real world systems

- Structural properties analysis : comparing topological features (WWW as benchmark)
- Beyond topological features : similarity measure (DNA) / move community (Go)

Studied systems : Network of C.elegans neurons
Network of DNA sequences
Network of moves in the game of Go
Opinion formation using PageRank (not presented here)

Application - C.elegans Neuron Network

(V. K and D. Shepelyansky, Phys. let. A, 2014)

Well-known animal in biology (Nobel prize : 2002, 2006 and 2008)

~ 1mm

H M T

Small system N = 279 neurons whose roles are known

Nodes = neurons

(2 types) Links = neural connections

1) Gap junction : inter-membrane communication (undirected)

$$Sgap = \begin{cases} 1 \text{ if neuron i and j are connected} \\ 0 \text{ otherwise} \end{cases}$$

3 types of neurons :

Sensory neurons
Interneurons
Motor neurones

2) Synaptic junction : axon connecting from one neuron to an other one (directed)

$$Ssyn = \begin{cases} 1 \text{ if neuron j points to neuron i} \\ 0 \text{ otherwise} \end{cases}$$

S = *Sgap*+*Ssyn* and $G = \alpha S + (1-\alpha)\frac{1}{N}\mathbf{e}\mathbf{e}^T$ G is of small size : exact diagonalization

Dataset : Neurons and connectivity structure available at wormatlas.org

Application - C.elegans Neuron Network

PageRank

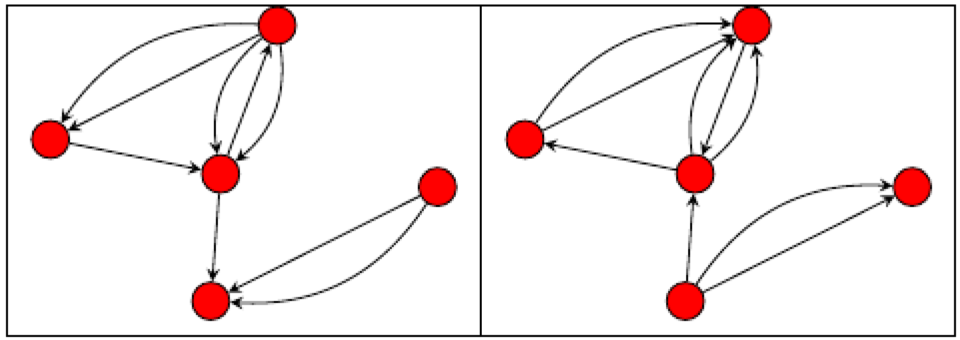
PR
AVAR
AVAL
PVCR
RIH
AIAL
PHAL
PHAR
ADEL
HSNR
RMGR
VC03
AIAR
AVBL
PVPL
AVM
AVKL
HSNL
RMGL
AVHR
AVFL

Application - C.elegans Neuron Network

PageRank

PR
AVAR
AVAL
PVCR
RIH
AIAL
PHAL
PHAR
ADEL
HSNR
RMGR
VC03
AIAR
AVBL
PVPL
AVM
AVKL
HSNL
RMGL
AVHR
AVFL

CheiRank : Inverted Network



Transposed S ⟶ G* ⟶ **P***

CheiRank = PageRank of inverted network

Ex : from Wikipedia articles about personalities
(*Zhirov et al., 2010*)

- PageRank highlights influential nodes
  (Napoleon I, G.W.Bush, Elizabeth II,...)

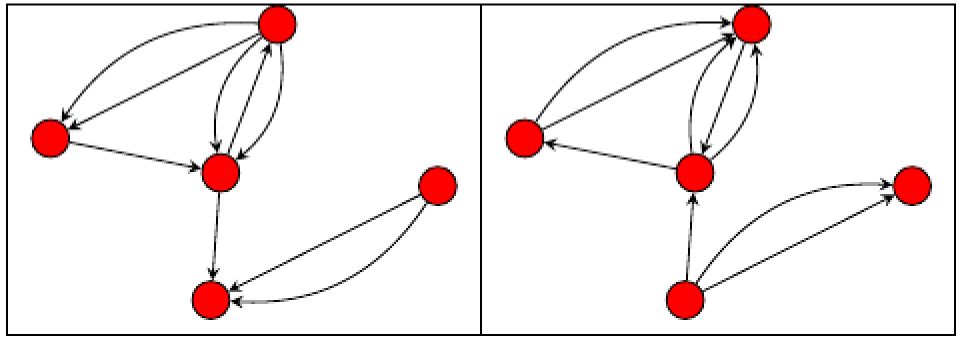
- CheiRank highlights communicative nodes
  (K.S.Pipes, R. Calmel, Y.G.Chernavsky,...)

Application - C.elegans Neuron Network

PageRank

PR
AVAR
AVAL
PVCR
RIH
AIAL
PHAL
PHAR
ADEL
HSNR
RMGR
VC03
AIAR
AVBL
PVPL
AVM
AVKL
HSNL
RMGL
AVHR
AVFL

CheiRank : Inverted Network



Transposed S ⟶ G* ⟶ **P***

CheiRank = PageRank of inverted network

Ex : from Wikipedia articles about personalities (*Zhirov et al., 2010*)

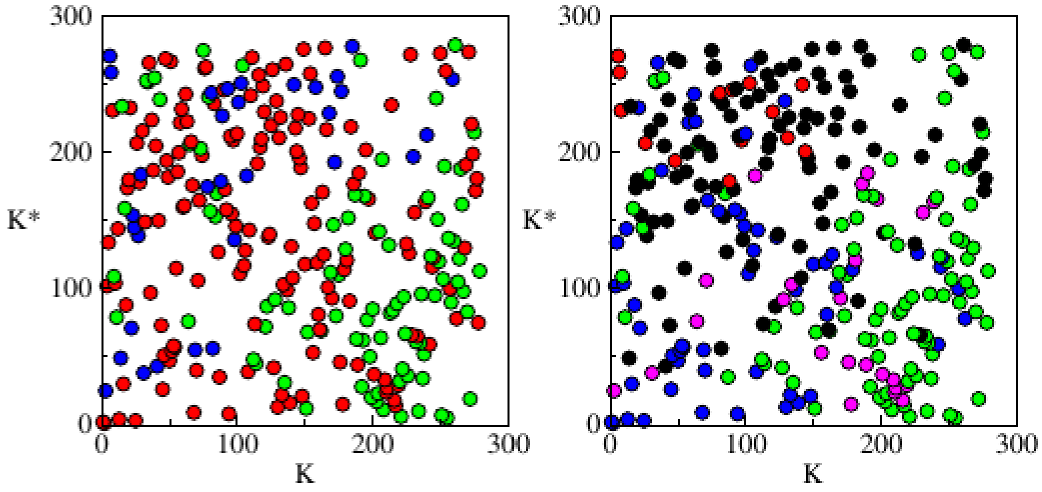
- PageRank highlights influential nodes *(Napoleon I, G.W.Bush, Elizabeth II,...)*
- CheiRank highlights communicative nodes *(K.S.Pipes, R. Calmel, Y.G.Chernavsky,...)*

CheiRank

CR
AVAL
AVAR
AVBR
AVBL
DD02
VD02
DD01
RIBL
RIBR
VD04
VD03
VD01
AVER
RMEV
RMDVR
AVEL
VD05
SMDDR
DD03
VA02

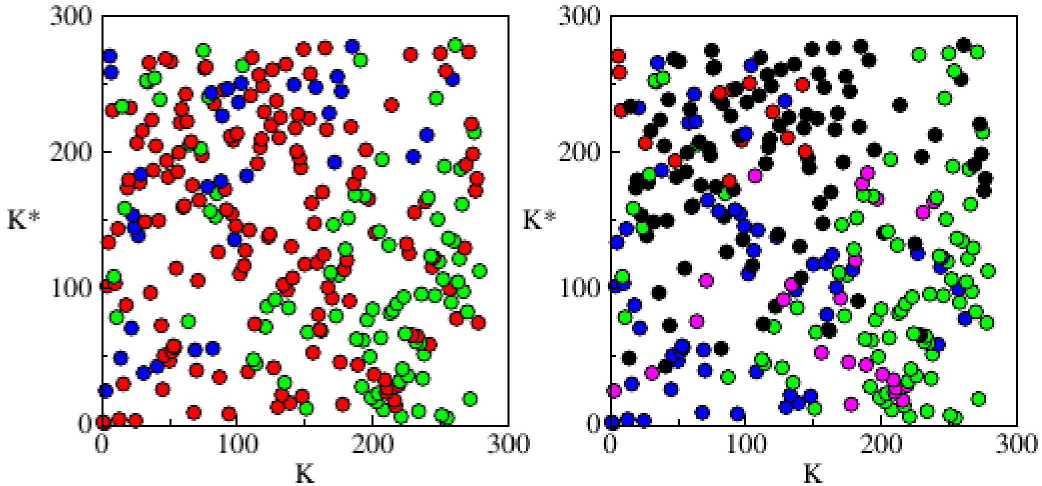
Application - C.elegans Neuron Network

PageRank – CheiRank correlation plot

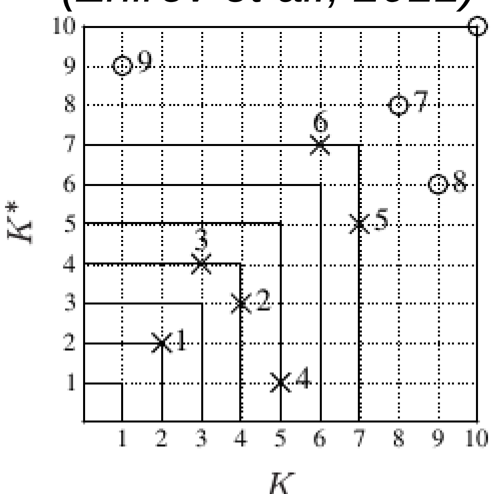


Application - C.elegans Neuron Network

PageRank – CheiRank correlation plot



(Zhirov et al., 2011)

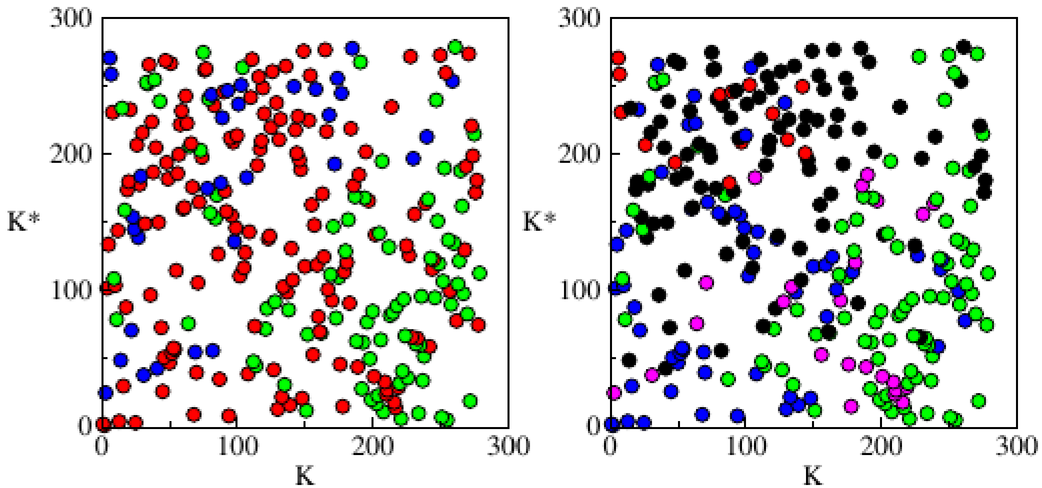


Explanation of 2DRank

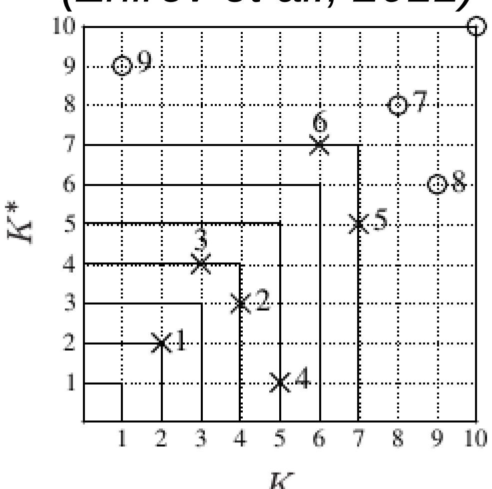
Other ranking :
2D Rank combining both
PageRank and CheiRank

Application - C.elegans Neuron Network

PageRank – CheiRank correlation plot



(Zhirov et al., 2011)



Explanation of 2DRank

Other ranking :
2D Rank combining both PageRank and CheiRank

PageRank, CheiRank and 2D Rank

	PR	CR	2DR
1	AVAR	AVAL	AVAL
2	AVAL	AVAR	AVAR
3	PVCR	AVBR	AVBL
4	RIH	AVBL	AVBR
5	AIAL	DD02	PVCR
6	PHAL	VD02	AVKL
7	PHAR	DD01	PVCL
8	ADEL	RIBL	PVPR
9	HSNR	RIBR	RIGL
10	RMGR	VD04	PVPL
11	VC03	VD03	RIS
12	AIAR	VD01	AVDR
13	AVBL	AVER	RIGR
14	PVPL	RMEV	AVDL
15	AVM	RMDVR	AVKR
16	AVKL	AVEL	RIBR
17	HSNL	VD05	DVC
18	RMGL	SMDDR	AIBL
19	AVHR	DD03	DVA
20	AVFL	VA02	AVJL

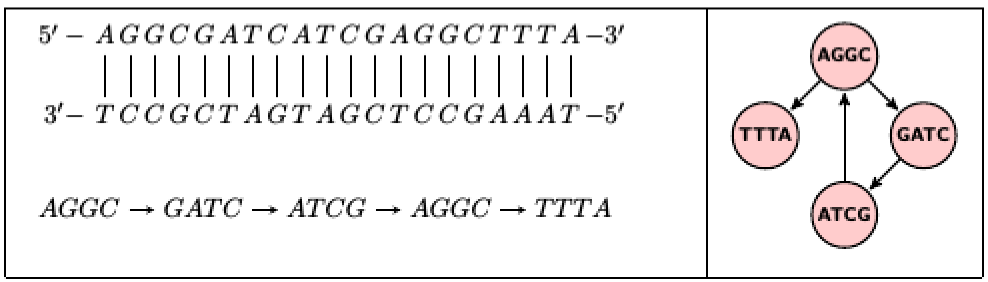
Application - DNA Sequence Network

(V. K and D. Shepelyansky, PLoS ONE, 2013)

Understanding the statistical properties of DNA,
Huge dataset available (low cost),
Original point of view of directed network

~ 4 nm

Dataset : 5 species (bull/cow - BT, dog - CH, elephant - LA, zebrafish - DR, human - HS) available at ensembl.org and sequence length L*seq* ~ 2 x 10^9 bp



Nodes = words of length m

Links = transition between words

Word length m (fixed) + 4 possible letters (A, C, T, G) ⟶ N = 4^m

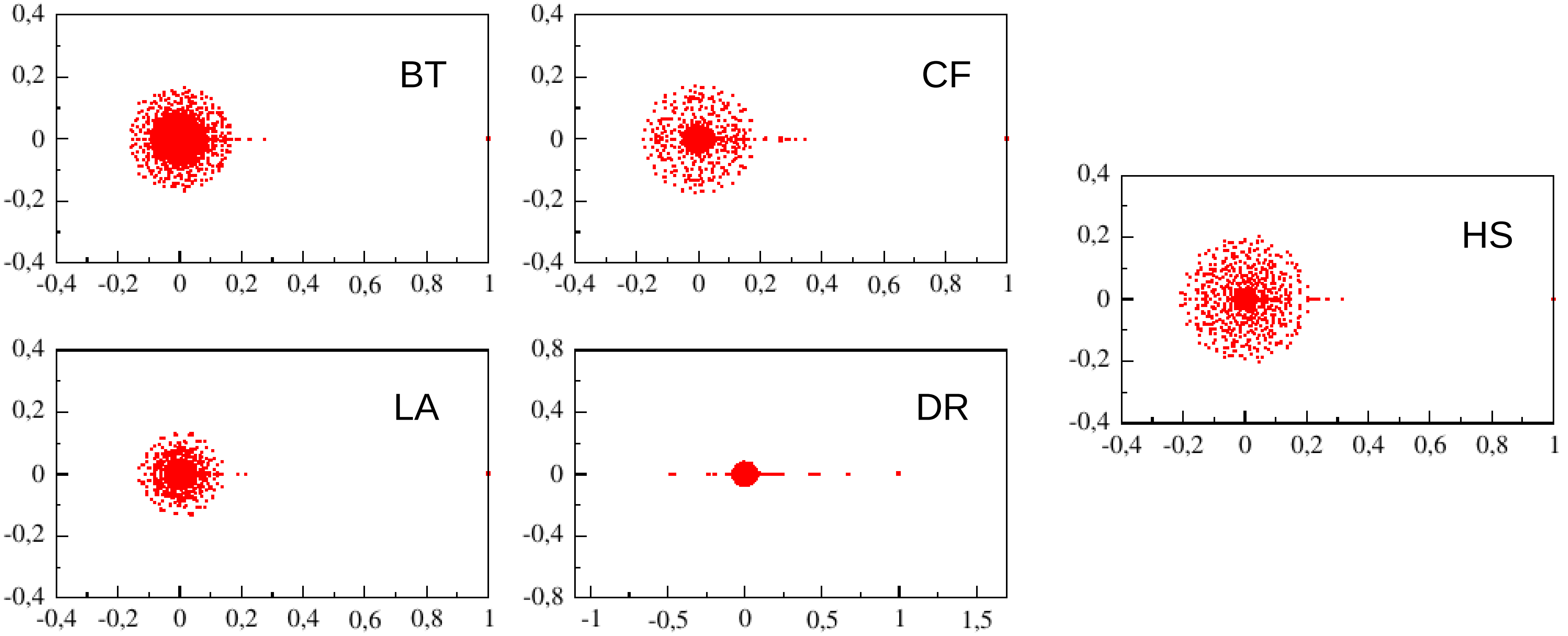
(for m=6 : N=4^6 =4096)

S = { m if word i follows word j m times in the database
0 otherwise

$$G = \alpha S + (1-\alpha)\frac{1}{N}\mathbf{e}\mathbf{e}^T$$

Application - DNA Sequence Network

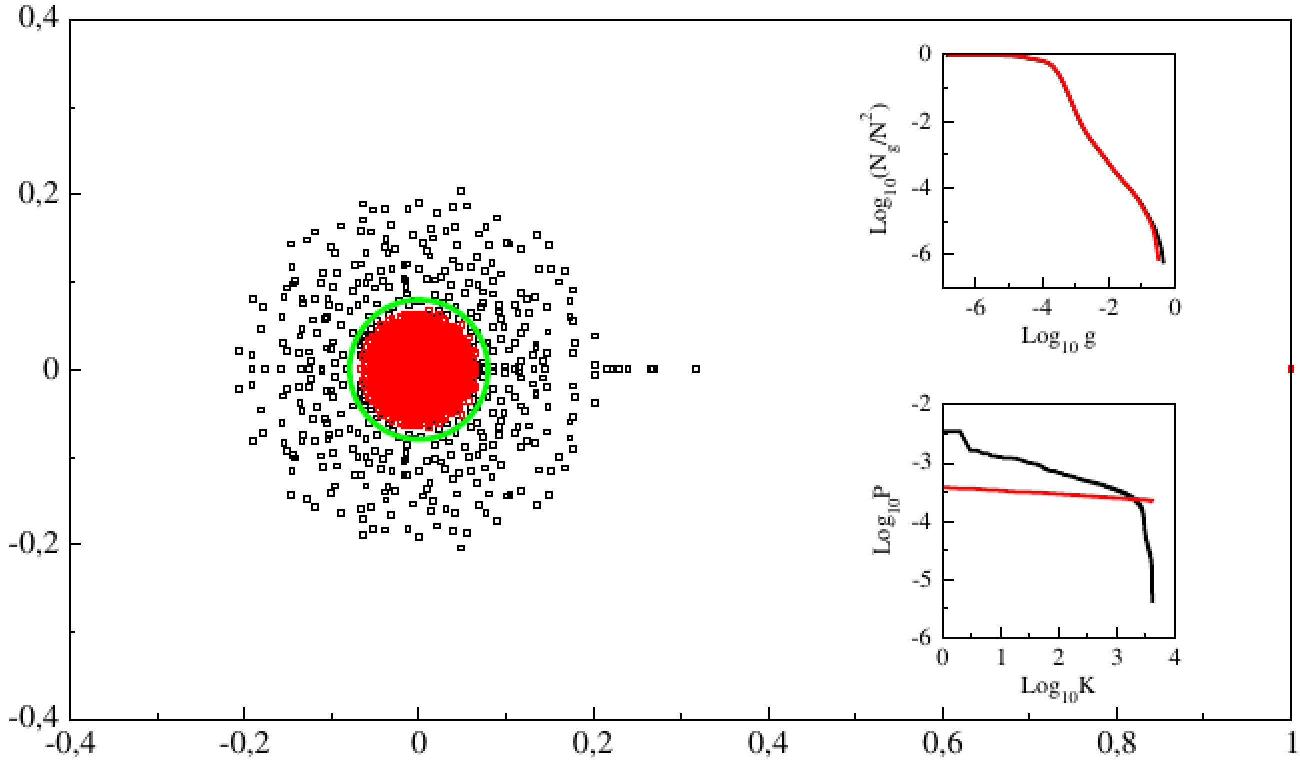
Spectrum of Google matrix G at α = 1 for various DNA sequences
bull(BT), dog (CF), elephant (LA), zebrafish (DR) and human (HS) at word length m=6



- All species have a large natural gap
- The spectrum can show differences between mammalian and non mammalian species

Application - DNA Sequence Network

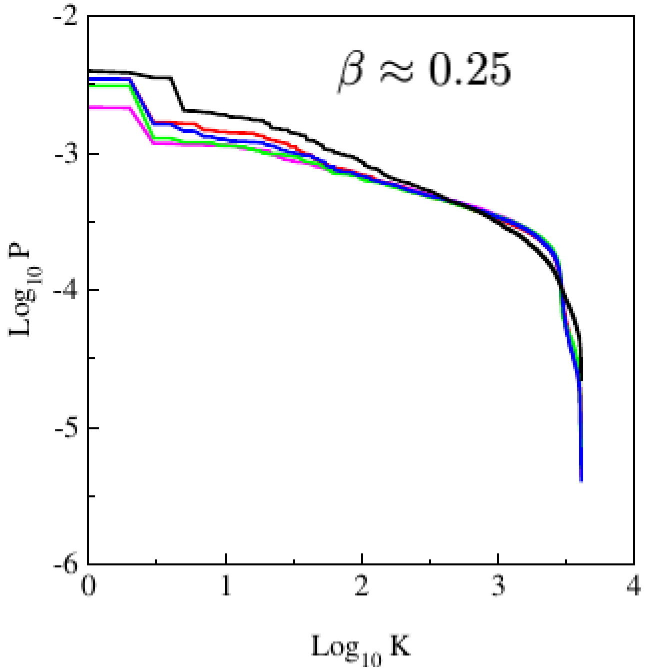
Comparison of Google matrix of Human DNA sequence (black) with Random Matrix Model (red)



The distribution of matrix elements alone cannot explain the structure of eigenvalues

Application - DNA Sequence Network

PageRank probability decay



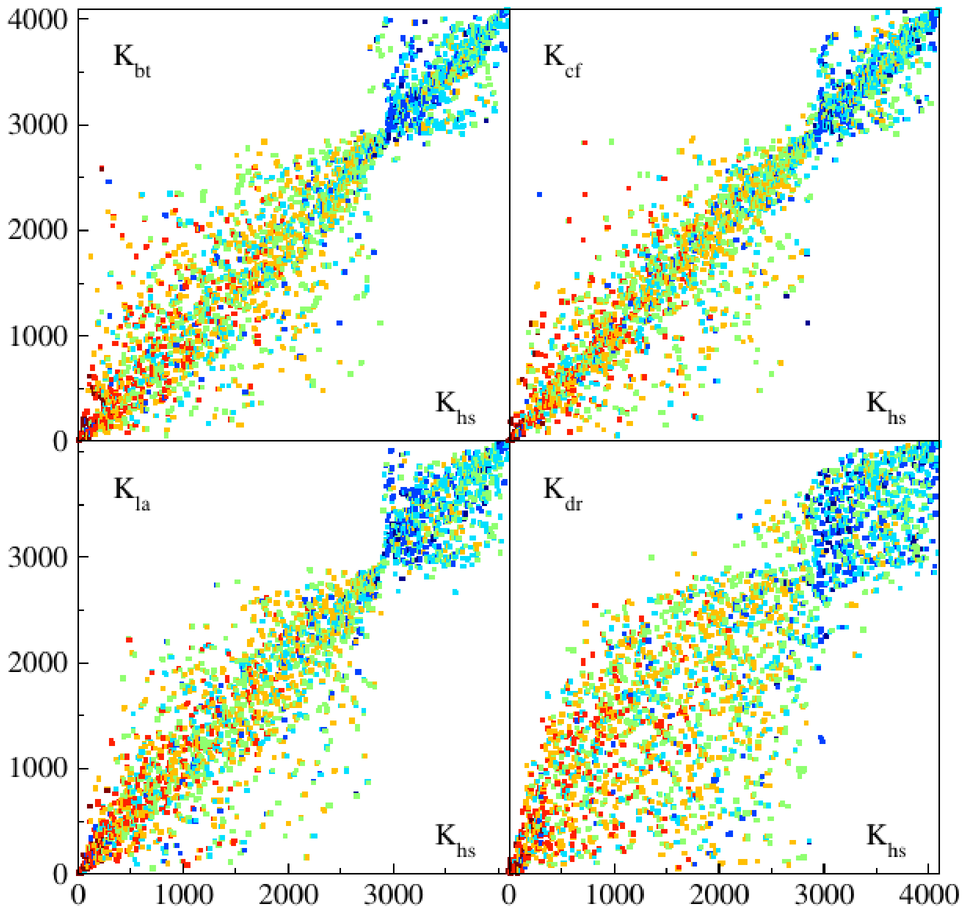
$$P(K) \sim \frac{1}{K^{\beta}}$$

- Similar behaviour of PageRank for various species
- Lower decay rate than in WWW

Top 10 PageRank entries					Last 10 PageRank entries				
BT	CF	LA	HS	DR	BT	CF	LA	HS	DR
TTTTTT	TTTTTT	AAAAAA	TTTTTT	ATATAT	CGCGTA	TACGCG	CGCGTA	TACGCG	CCGACG
AAAAAA	AAAAAA	TTTTTT	AAAAAA	TATATA	TACGCG	CGCGTA	TACGCG	CGCGTA	CGTCGG
ATTTTT	AATAAA	ATTTTT	ATTTTT	AAAAAA	CGTACG	TCGCGA	ATCGCG	CGTACG	CGTCGA
AAAAAT	TTTATT	AAAAAT	AAAAAT	TTTTTT	CGATCG	CGTACG	TCGCGA	TCGACG	TCGACG
TTCTTT	AAATAA	AGAAAA	TATTTT	AATAAA	ATCGCG	CGATCG	CGCGAT	CGTCGA	TCGTCG
TTTTAA	TTATTT	TTTTCT	AAAATA	TTTATT	CGCGAT	CGAACG	GTCGCG	CGATCG	CCGTCG
AAAGAA	AAAAAT	AAGAAA	TTTTTA	AAATAA	TCGACG	CGTTCG	CGATCG	CGTTCG	CGACGG
TTAAAA	ATTTTT	TTTCTT	TAAAAA	TTATTT	CGTCGA	TCGACG	CGCGAC	CGAACG	CGACCG
TTTTCT	TTTTTA	TTTTTA	TTATTT	CACACA	CGTTCG	CGTCGA	TCGCGC	CGACGA	CGGTCG
AGAAAA	TAAAAA	TAAAAA	AAATAA	TGTGTG	TCGTCG	ACGCGA	ACGCGA	CGCGAA	CGACGA

Application - DNA Sequence Network

PageRank – PageRank comparison between species



Empirical way of quantifying the similarity ($\zeta \sim \sigma$)

$$\sigma(s_1, s_2) = \sqrt{\sum_{i=1}^{N} (K_{s_1}(i) - K_{s_2}(i))^2 / N}$$

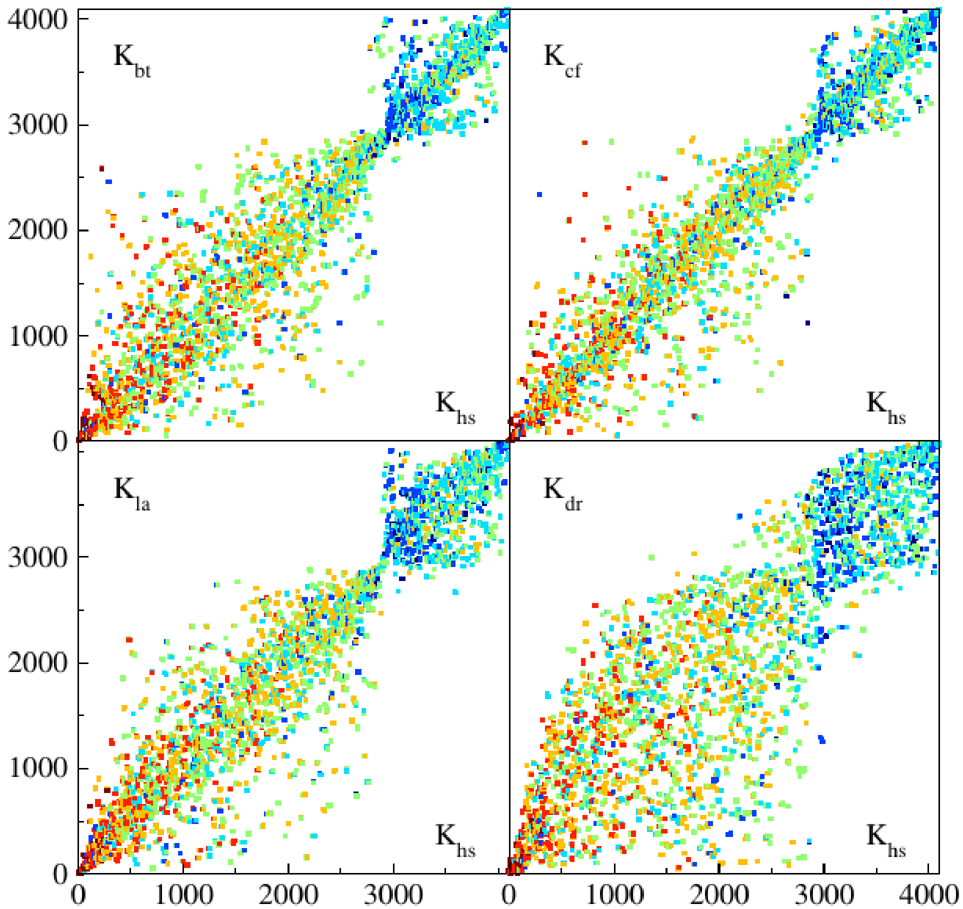
ζ	BT	CF	LA	HS	DR
BT	0.000	0.308	0.324	0.246	0.425
CF	0.308	0.000	0.303	0.206	0.414
LA	0.324	0.303	0.000	0.238	0.422
HS	0.246	0.206	0.238	0.000	0.375
DR	0.425	0.414	0.422	0.375	0.000

Human and dog are the most similar

Rank correlation allows to determine the similarity between species from directed network viewpoint

Application - DNA Sequence Network

PageRank – PageRank comparison between species



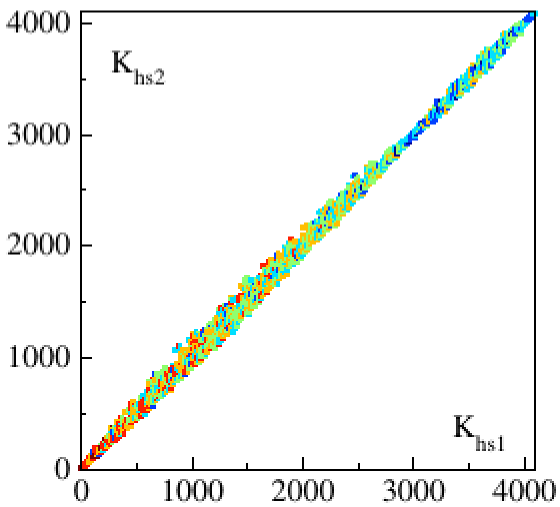
Rank correlation allows to determine the similarity between species from directed network viewpoint

Empirical way of quantifying the similarity (ζ ~ σ)

$$\sigma(s_1, s_2) = \sqrt{\sum_{i=1}^{N} (K_{s_1}(i) - K_{s_2}(i))^2 / N}$$

ζ	BT	CF	LA	HS	DR
BT	0.000	0.308	0.324	0.246	0.425
CF	0.308	0.000	0.303	0.206	0.414
LA	0.324	0.303	0.000	0.238	0.422
HS	0.246	0.206	0.238	0.000	0.375
DR	0.425	0.414	0.422	0.375	0.000

Human and dog are the most similar



Application - Network of the Game of Go

(V. K, B. Georgeot and O. Giraud, EPJB, 2014)

Understanding decision making process through study of gaming

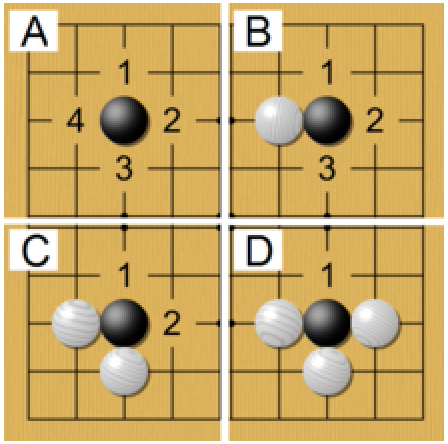
New approach of using directed networks to study games

No computer program has been able to beat a strong human player

Ancient Asian game, very popular. Played by two opponents on board containing 19 x 19 intersections

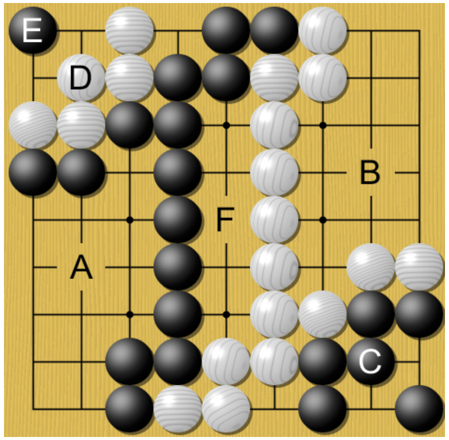
Goal : building large territories

Traditional Goban



Black stone being surrounded, in (D) it is in atari status, if white plays in 1, the black stone is captured and removed from the Goban

Example of chains delimiting some territories



Application - Network of the Game of Go

Obstacles hindering the creation of efficient Go programs :

1) The size of the Goban is huge and the number of configurations is too large to be handled

2) Difficult for the computer to estimate the relevancy of a move in a given context

Current approach :

1) Monte Carlo Go algorithm, evaluates a move's value by playing randomly many games until the end and counting how many times it leads to a win

2) Improvements thanks to tricks added to explore more efficiently the tree of possible moves

→ Tricks are not enough to beat strong players on 19 x 19 Goban

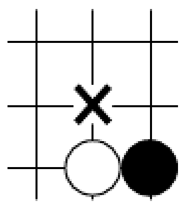
Hope :

Directed network approach might help in evaluating moves to improve the Monte Carlo Go

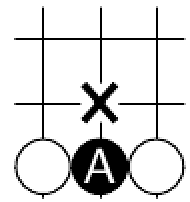
Application - Network of the Game of Go

How are the nodes defined ? Example of a node in each case :

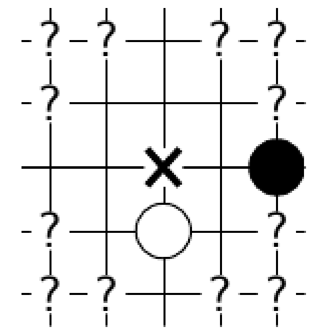
Network I



Network II



Network III



The nodes (also called plaquettes here) have been filtered by shape symmetry
And color swap symmetry : we retain only non equivalent plaquettes

Network I : N = 1107 non equivalent plaquettes

Network II : N = 2051 non equivalent plaquettes

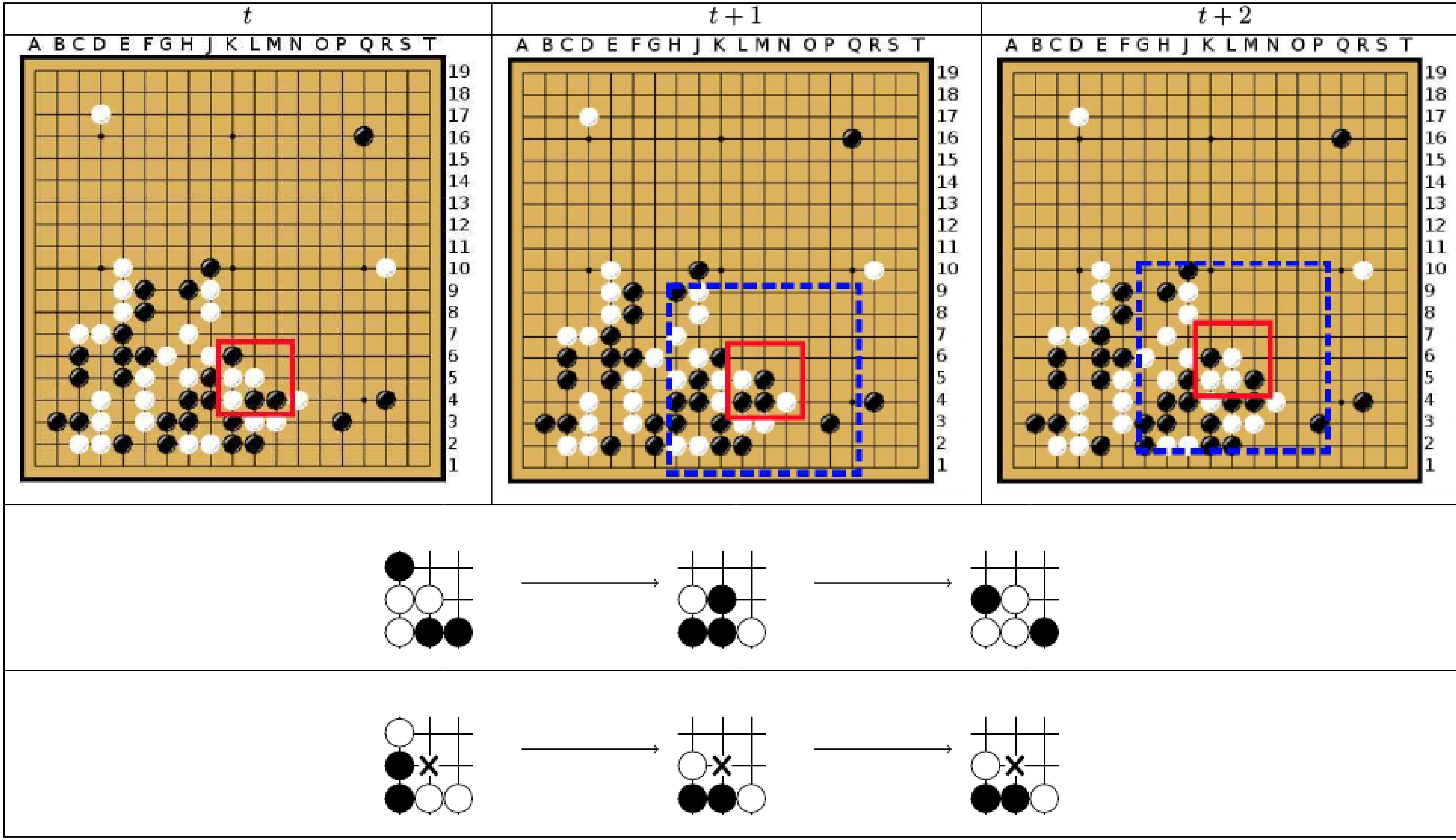
Network III : N = 193995 non equivalent plaquettes

Database : U-go.net
~ 135000 recorded game files in .sgf format, player levels are given

```
HA[4]
;W[jp];B[jd];W[jj];B[pj];W[cf];B[dj];W[cn];B[en];W[fc];B
    [ee];W[fq];B[el];W[cj];B[ci];W[ck];B[di];W[cp];B[cq
    ];W[do];B[eo];W[dq];B[ep];W[cr];B[eq];W[bq];B[dr];W[
    cc];B[dc];W[db];B[cd];W[cb];B[bc];W[bb];B[bd];W[gd];
    B[fr];W[nq];B[pn];W[nc];B[oc];W[nd];B[pf];W[nf];B[jg
    ];W[ff];B[dg];W[kf];B[jf];W[je];B[ie];W[ke];B[id];W[
```

Application - Network of the Game of Go

How are the links defined ? Explanation through an example

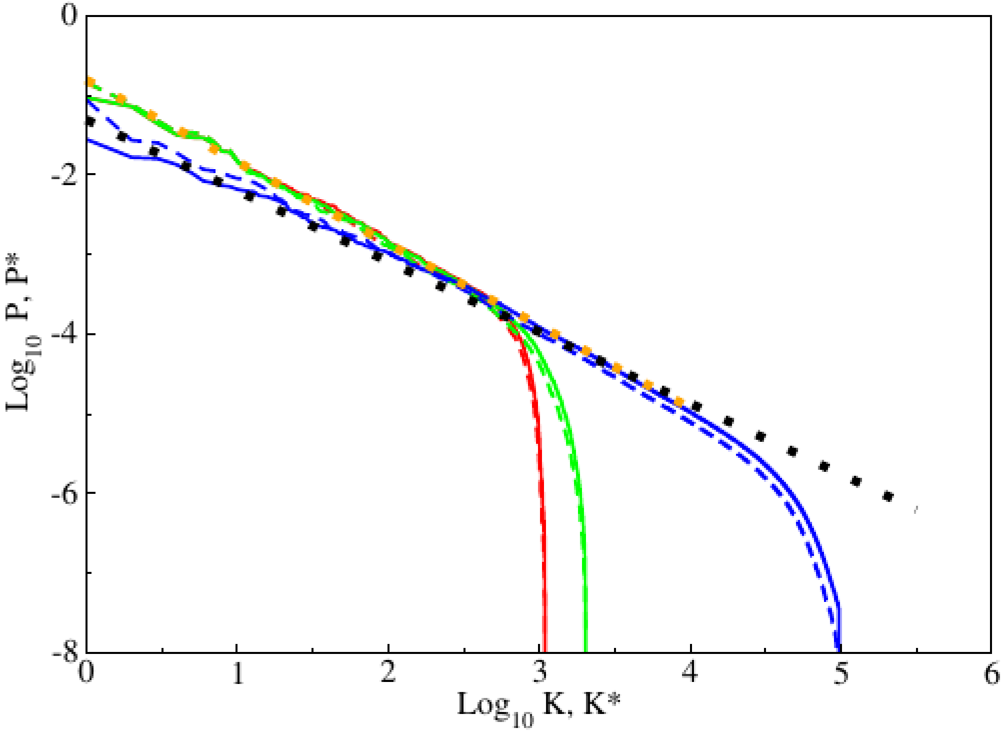


Nodes = plaquettes

Links = succession from a plaquette to another

Application - Network of the Game of Go

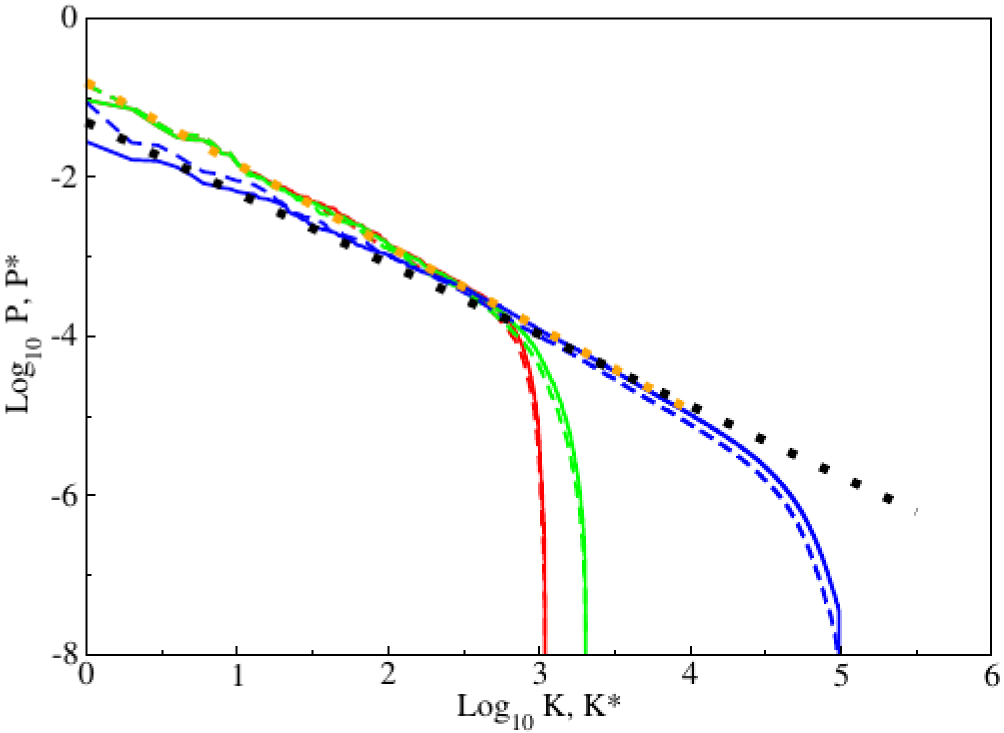
Rank distribution decay for the three networks



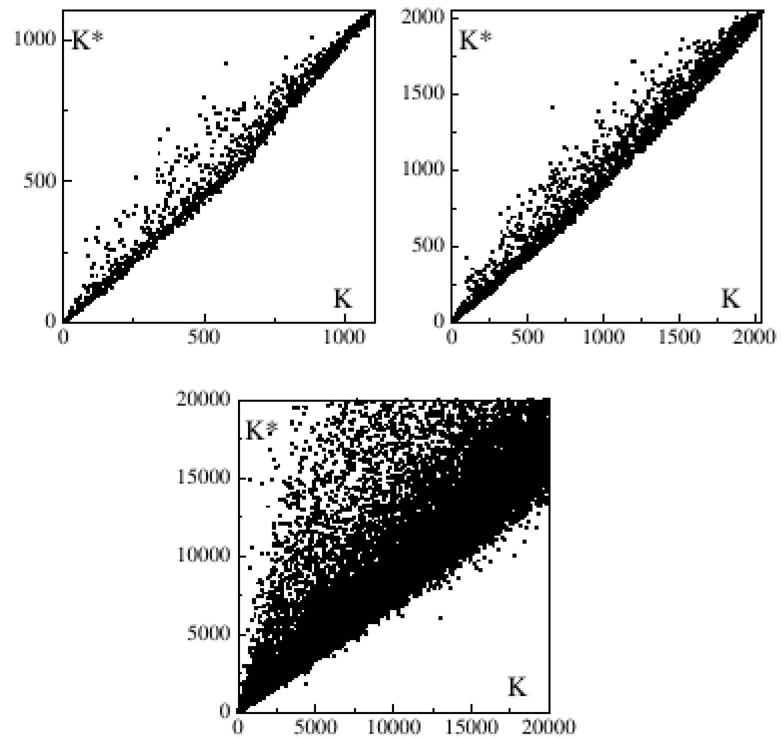
- There is a symmetry between PageRank and CheiRank distribution decay (it is not the case in usual WWW like networks)

Application - Network of the Game of Go

Rank distribution decay for the three networks



PageRank – CheiRank correlations

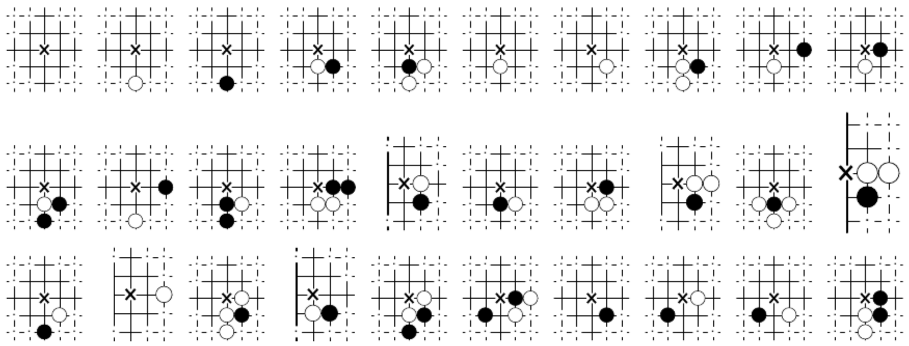


- There is a symmetry between PageRank and CheiRank distribution decay (it is not the case in usual WWW like networks)
- The symmetry is not perfect, it is also weaker in the largest network

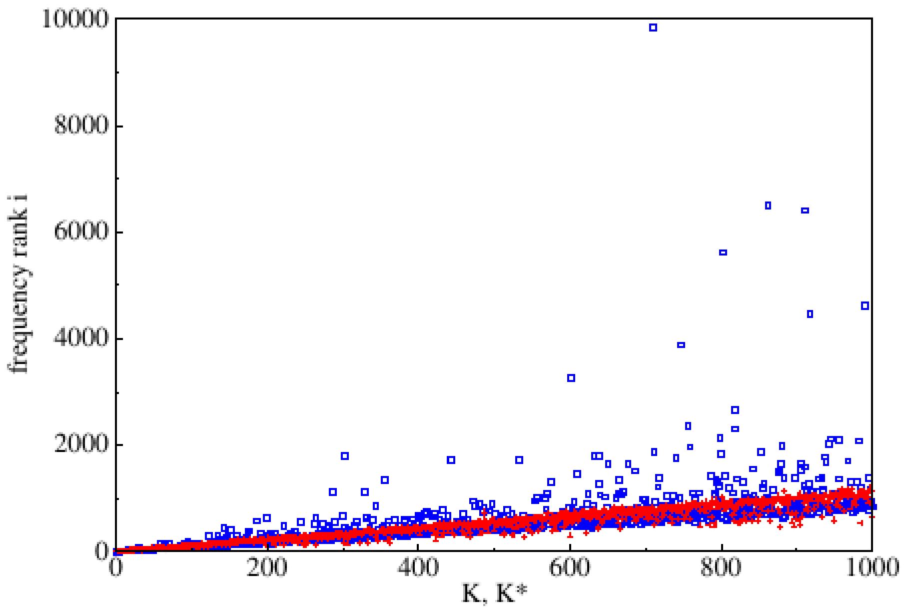
Application - Network of the Game of Go

PageRank and CheiRank highlight moves similar to frequency rank but not exactly the same

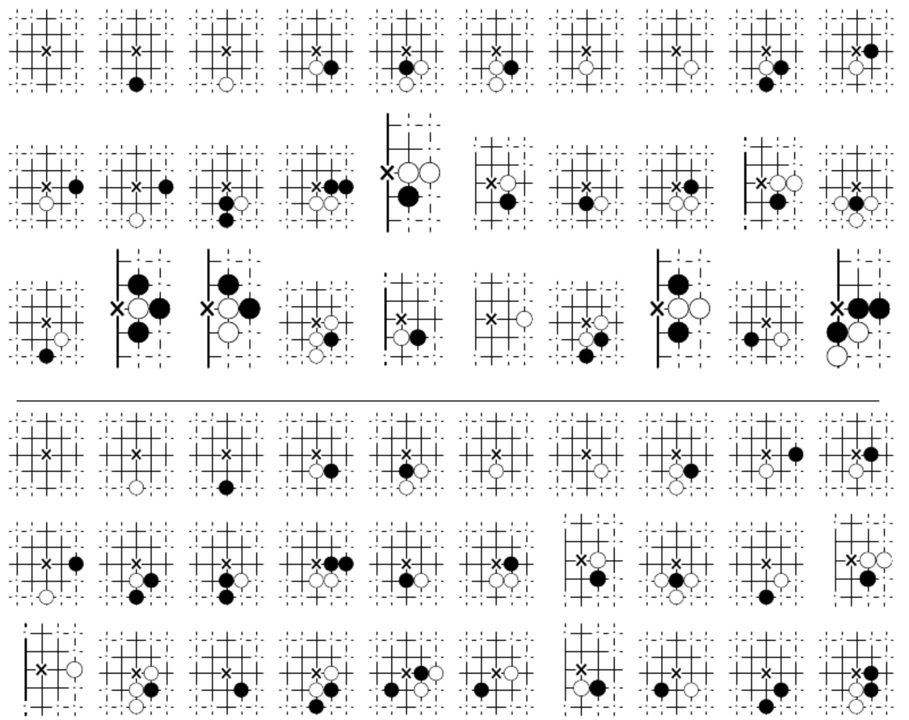
Top 30 moves by frequency rank



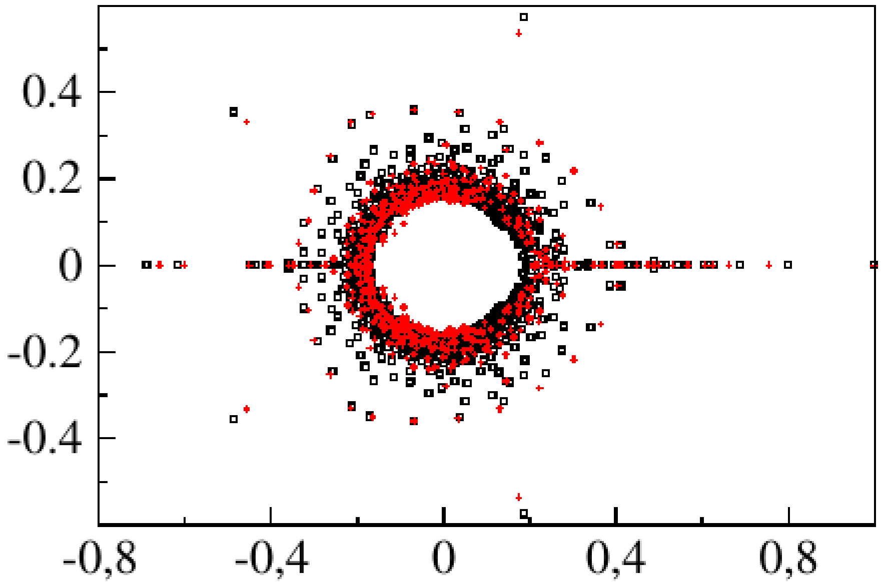
Frequency rank vs PageRank/CheiRank



Top 30 moves by PageRank/CheiRank

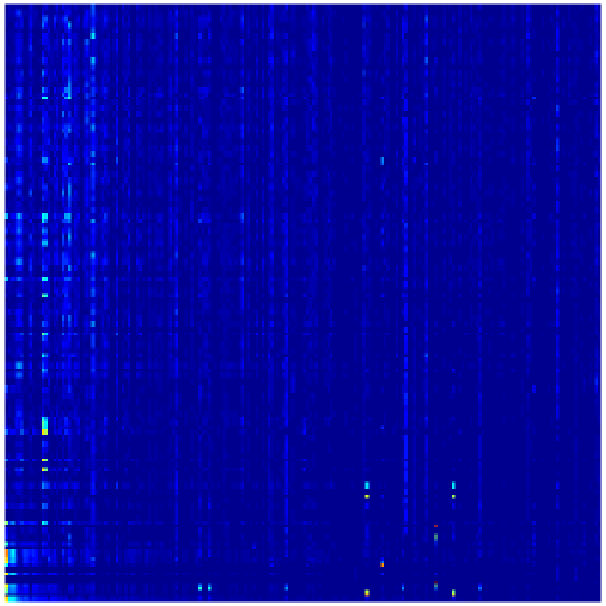


Application - Network of the Game of Go



Spectrum of Google matrices G (black) and G* (red) at α = 1, computed with Arnoldi method

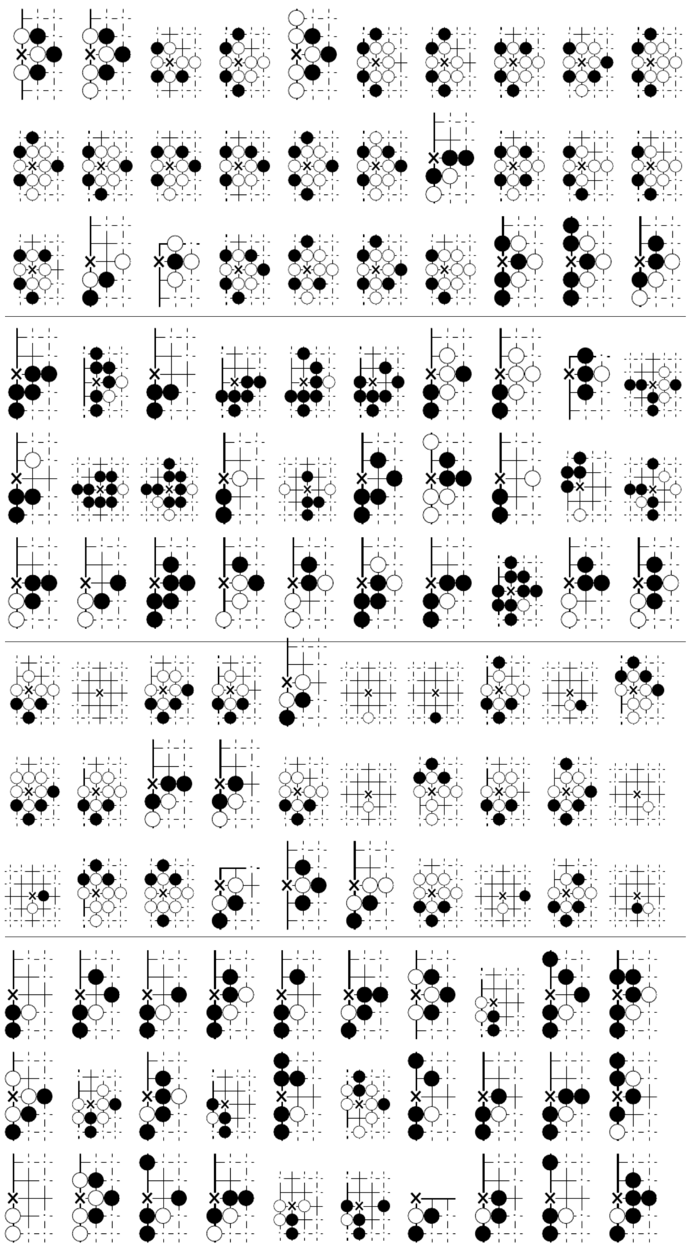
Only a few hundreds of largest eigenvalues are shown, the eigenvalues of large modulus indicate the presence of community of moves



A few hundreds of eigenvectors of G stacked horizontally from bottom to top and only a few hundreds of elements are shown in PageRank basis. Colors represent the modulus of eigenvector elements.

Presence of correlations (visible lines) not necessarily at high values of PageRank : Indication that the eigenvectors do contain some information about group of moves different than those highlighted by PageRank

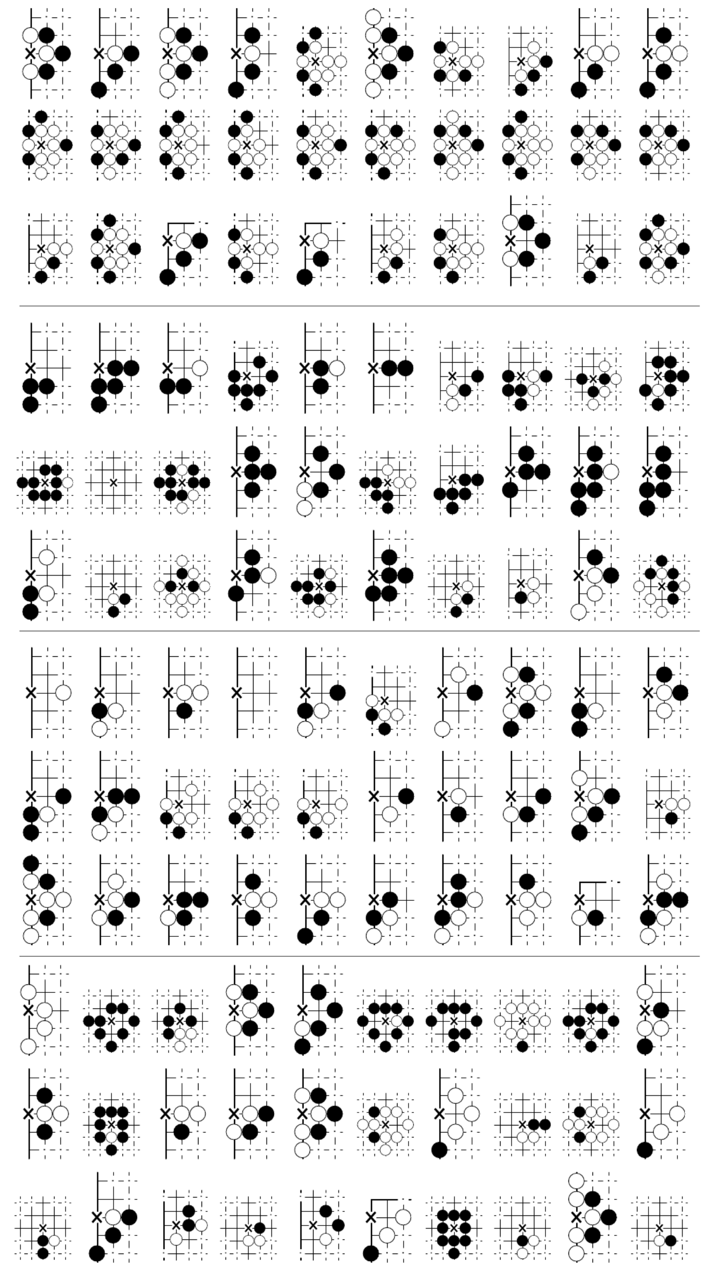
Application - Network of the Game of Go



Examples of top 30 moves where eigenvectors of G (left) and G*(right) are localized

From top to bottom : 7th, 11th, 18th and 21st eigenvectors

Impression : different groups mixed in the same eigenvector



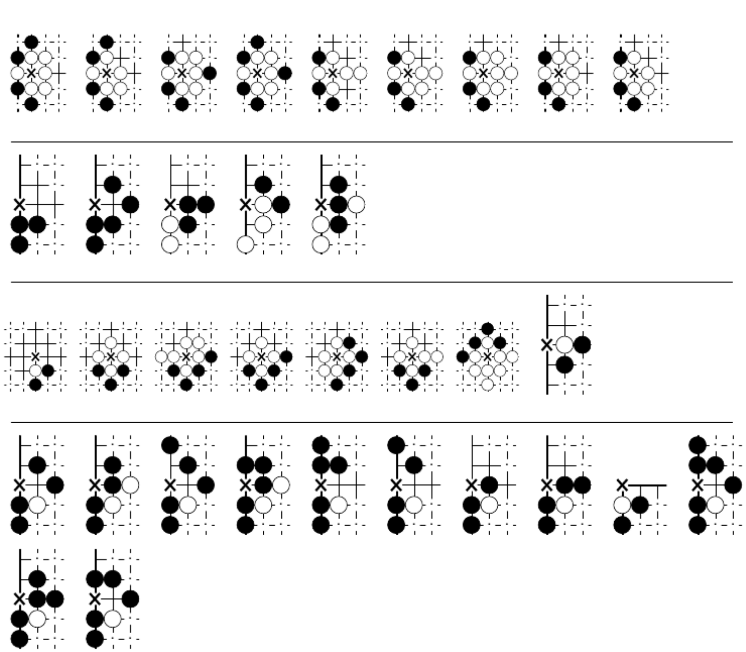
Application - Network of the Game of Go

Need for a method to extract those mixed groups :

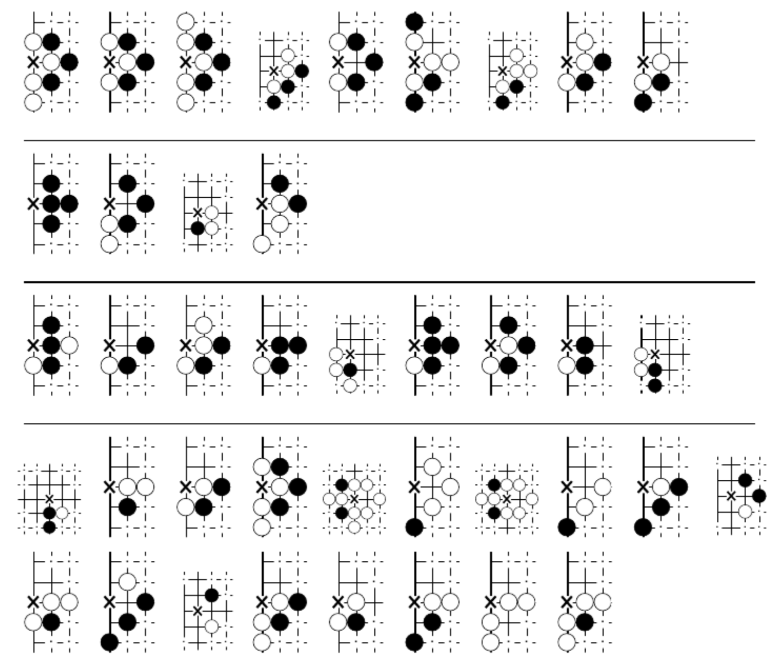
1) first natural step : remove most important moves (top PageRank/CheiRank)

2) several communities might be mixed : regroup them by common ancestry method

Common ancestry : a community is made of members sharing more than a threshold number of common ancestors. The threshold is an arbitrary parameter that needs to be tuned depending on the network.



Results :
More homogeneous ("Ko", attempt to connect on the rim,...)



Perspective and Conclusion

Summary :

- Google matrix method is useful and easy to characterize topological features of various systems and compare them
- Possibility to define several rankings depending on the needs and use them as more than just a rank listing

Limitations :

- Neurons : dynamics and neuron rewiring are not taken into account
- Game of Go : need for a more systematic way of extracting a specific community
  need for a deeper understanding of move community and a bridge to implement the ideas presented for it to be really useful

Perspective :

- Personalization of teleportation matrix
- Time variation of rank index K for dynamical networks