A network for the game of go

Vivek Kandiah, Bertrand Georgeot
Laboratoire de Physique Théorique, IRSAMC
CNRS/ University Paul Sabatier, Toulouse
Olivier Giraud
LPTMS CNRS/University Paris-Sud

Supported by EC FET Open project NADINE
CNRS and Région Midi-Pyrénées
B.G. and O. Giraud, Europhysics Letters 97 68002 (2012)
V. Kandiah, B.G. and O. Giraud, arxiv 1405.6077

Networks

->Recent field: study of complex networks.Tools and models have been created;

->Many networks are scale-free with power-law distribution of links difference between directed and non directed networks

->Important examples from recent technological developments: internet, World Wide Web, social networks...

->Can be applied also to less recent objects
in particular, study of human behavior: languages, friendships…

Networks for games

-> Network theory never applied to games

-> Games are nevertheless a very ancient activity, with a mathematical theory attached to the more complex ones

-> Games represent a privileged approach to human decision-making

->Can be very difficult to modelize or simulate

The game of go

→Game of go: very ancient Asian game, probably originated in China in Antiquity (image on the left from VIIIth century)

-> Go is the Japanese name; Weiqi in Chinese, Baduk in Korean

The game of go

-> Go is a very popular game played by many parts of the population (ex. right) on a board called Goban (see below)

Rules of go

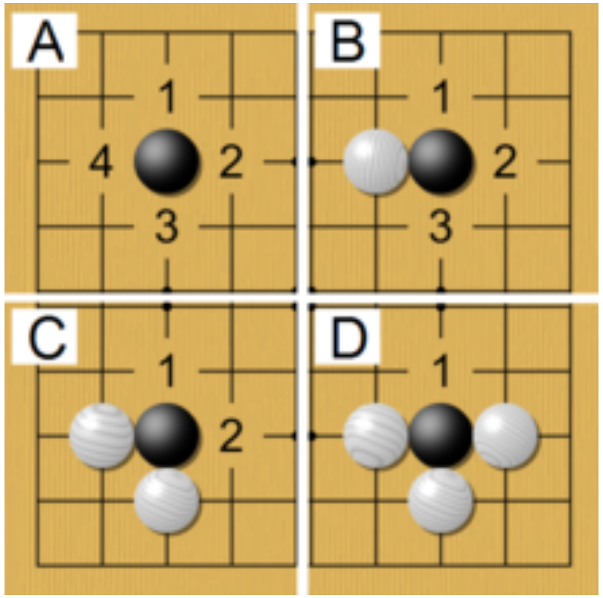
->White and black stones alternatively put at intersections of 19 x19 lines

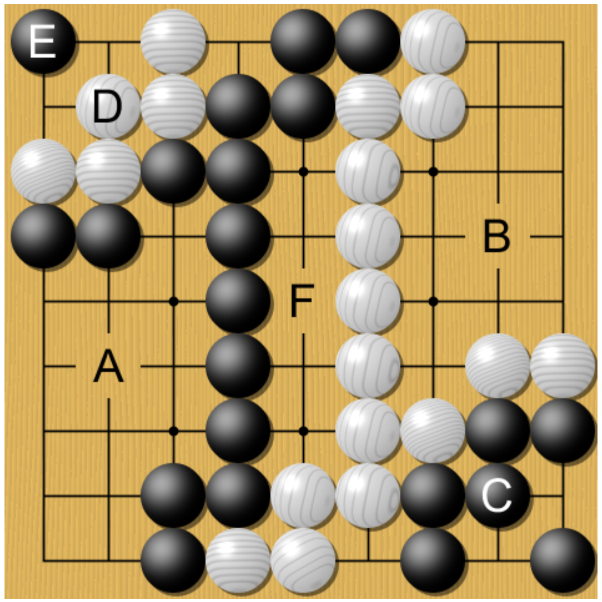
->Stones without liberties are removed

->A chain with only one liberty is said in atari

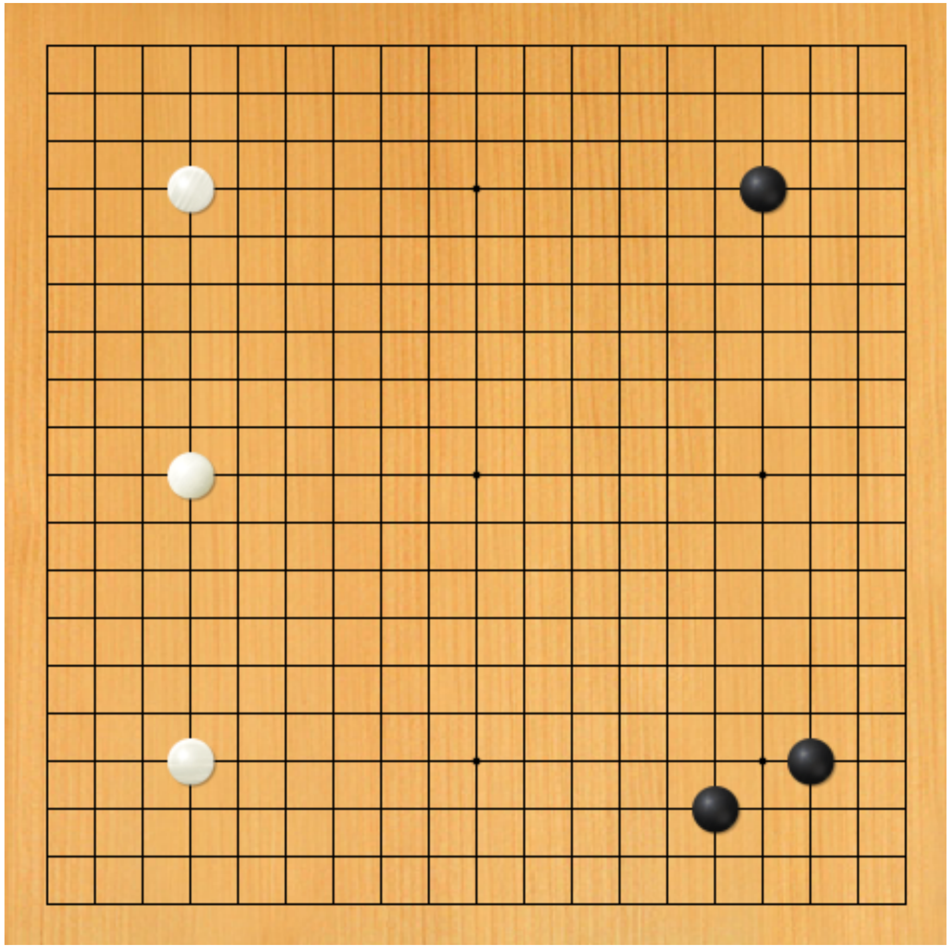
->Handicap stones can be placed

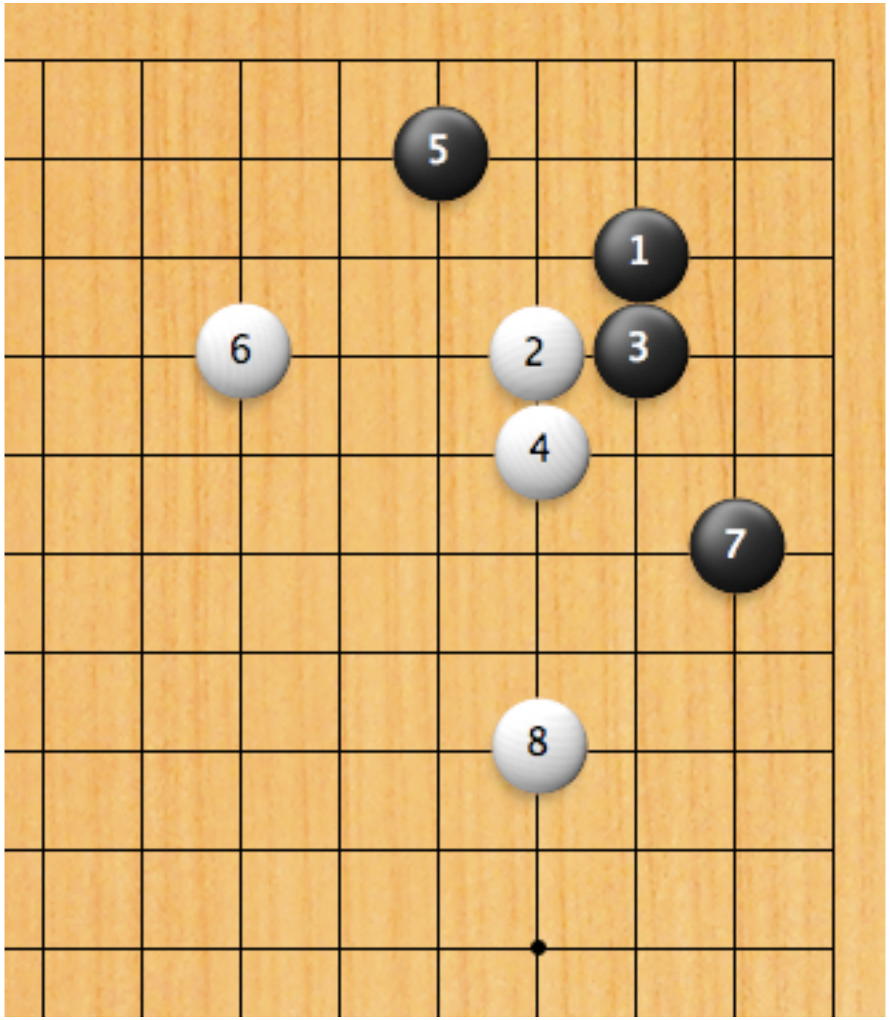
->Aim of the game: construct protected territories



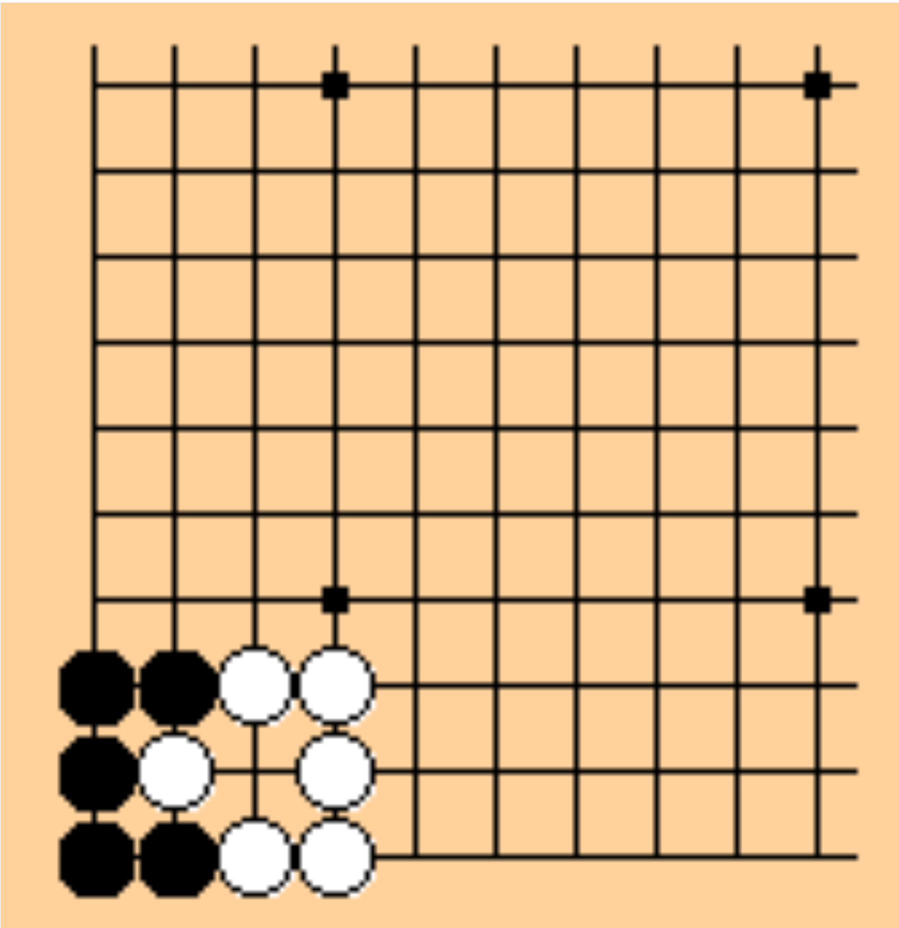


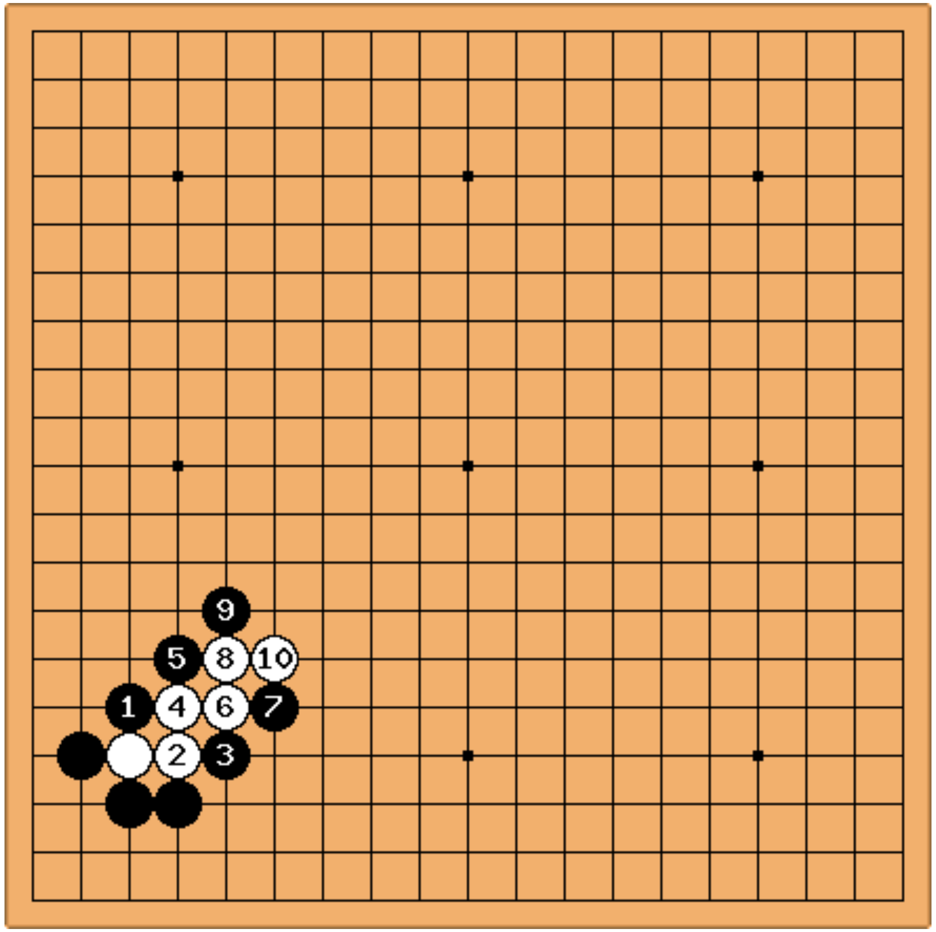
Beginnings: Fuseki and Joseki



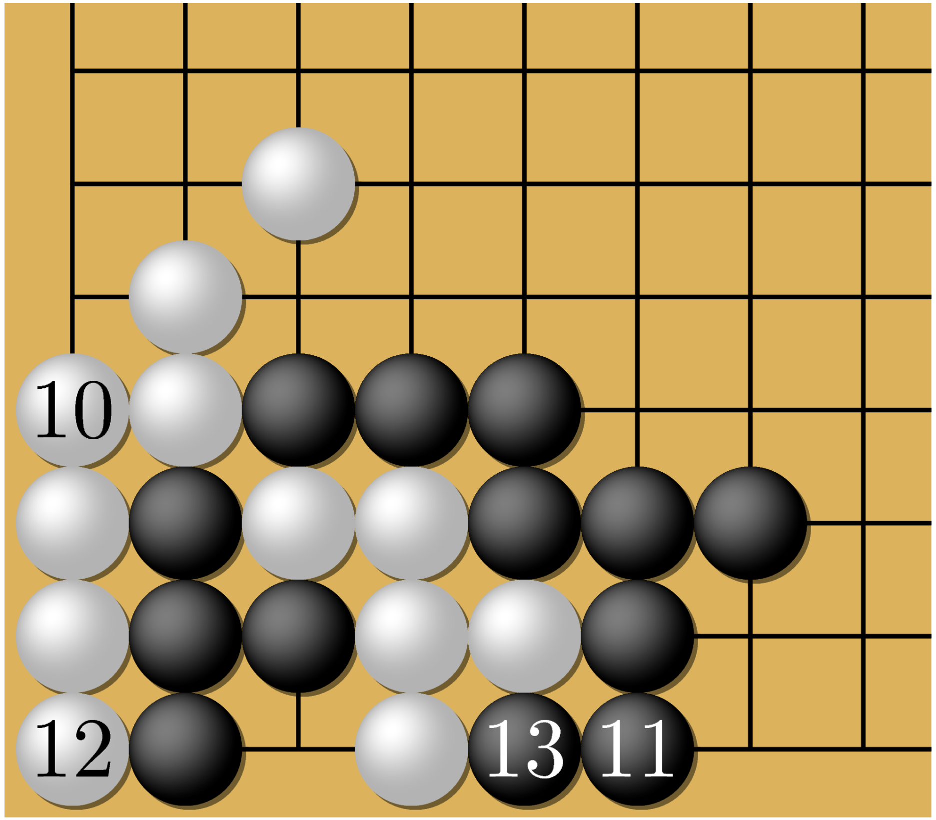


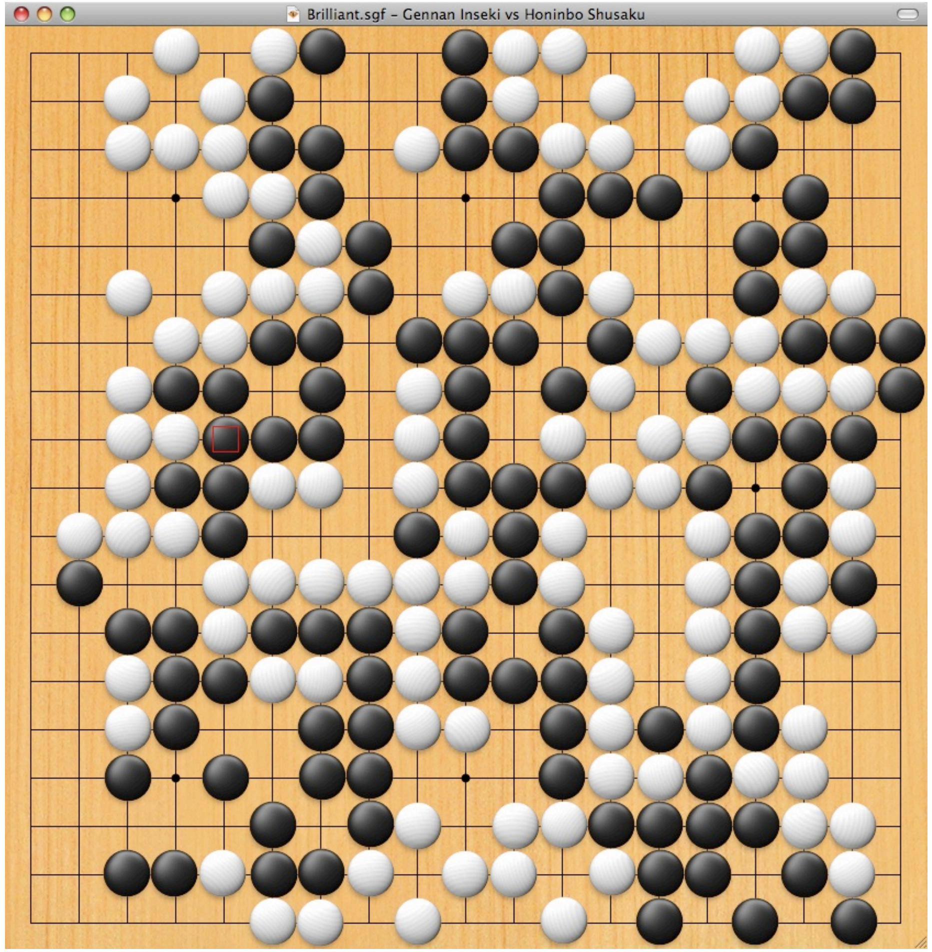
During the game-Ko and ladders





Endgames-life and death





Player rankings

→There are nine levels (dans) of professionals (top players) followed by nine levels of amateurs
->A handicap stone can compensate for roughly one dan: like in golfing, players of different levels can play evenly thanks to handicaps
->There are regular tournaments of go since very long times

Computer simulations

-->While Deep Blue famously beat the world chess champion Kasparov in 1997, no computer program has beaten a very good go player even in recent times. Why?

->total number of legal positions 10^{171}, vs "only" 10^{50} for chess

-> Not easy to assign positional advantage to a move

-> Best programs use Monte Carlo Go: play random games starting from one move and see the outcome until a value can be assigned to the move

->Monte Carlo Go has beaten all other programs, can beat professional players on 9x9 gobans, and with handicaps on 19x19

Databases

->We use databases of expert and amateur games in order to construct networks from the different sequences of moves, and study the properties of these networks

->Databases available at http://www.u-go.net/

->Whole available record, from 1941 onwards, of the most important historical professional Japanese go tournaments: Kisei (143 games), Meijin (259 games), Honinbo (305 games), Judan (158 games)

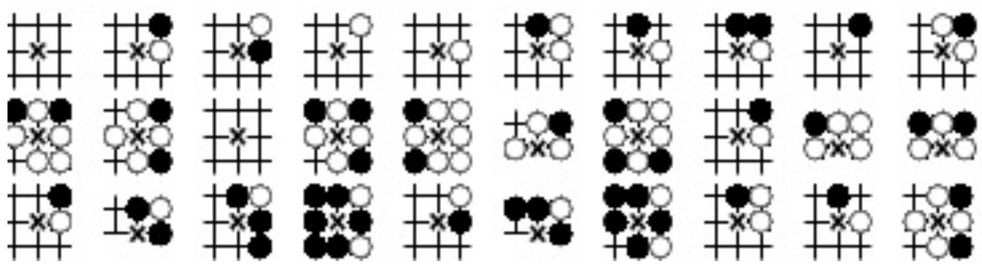
-Contains also 135 000 amateur games played online

->Level of players is known, mutually assessed according to games played

Vertices of the network I

->"plaquette" : square of 3 x3 intersections

->We identify plaquettes related by symmetry
->We identify plaquettes with colors swapped
->1107 nonequivalent plaquettes with empty centers
->vertices of our network

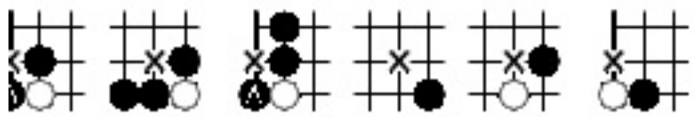


Vertices of the networks II

->"plaquette" : square of 3 x3 intersections + atari status of nearest-neighbors

->We still identify plaquettes related by symmetry

->Because of rules restrictions, only 2051 legal nonequivalent plaquettes with empty centers

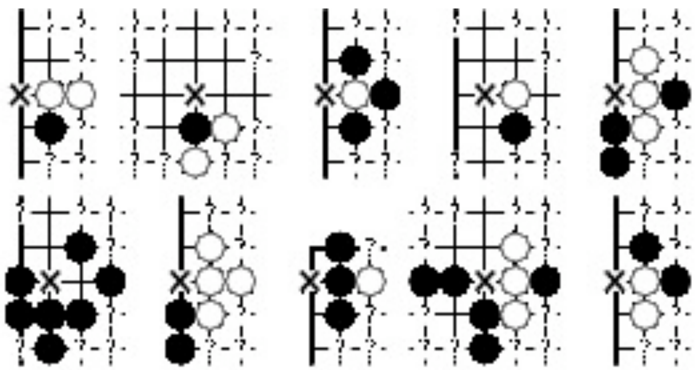


Vertices of the networks III

->"plaquette" : diamond of 3 x3 +4 intersections

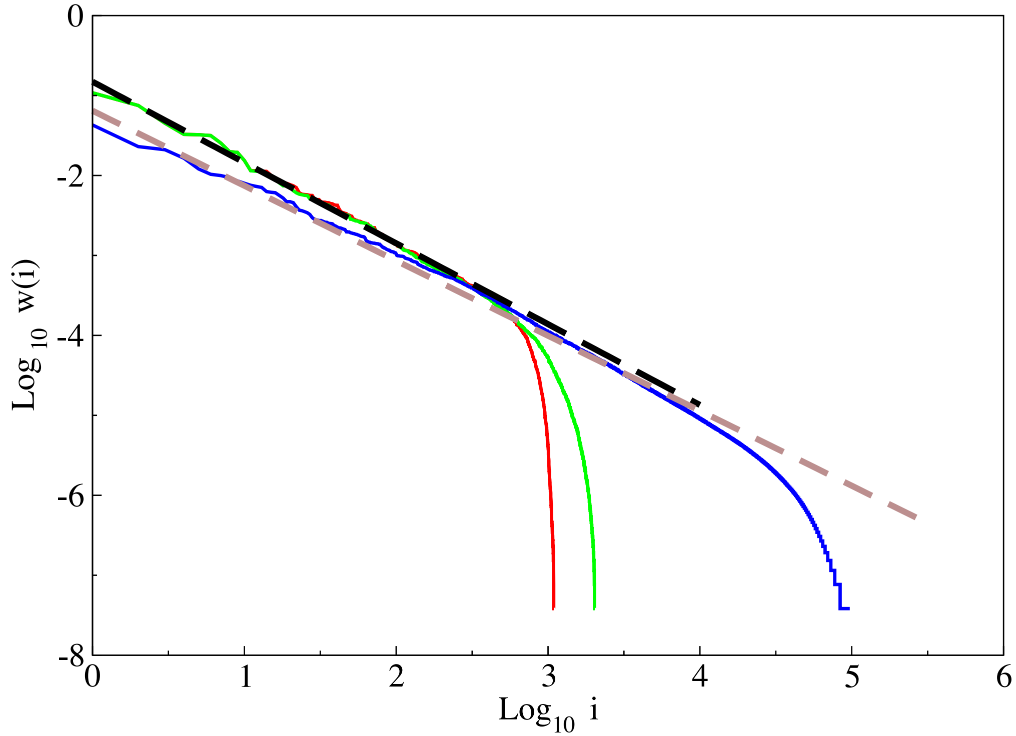
->We still identify plaquettes related by symmetry

->193995 nonequivalent plaquettes with empty centers
(96771 actually never used in the database)



Zipf's law

->Zipf's law: empirical law observed in many natural distributions (word frequency, city sizes...)
->If items are ranked according to their frequency, predicts a power-law decay of the frequency vs the rank.
->integrated distribution of three network nodes clearly follows a Zipf's law,
with exponent close to 1



Normalized integrated frequency distribution of three types of nodes. Thick dashed line is y=-x.

Links of the network

->we connect vertices corresponding to moves a and b if b follows a in a game at a distance < d.
->Each choice of d defines a different network. The choice of d determines the distance beyond which two moves are considered nonrelated.
->Sequences of moves follow Zipf's law (cf languages) Exponent decreases as longer sequences reflect individual strategies
->move sequences are well hierarchized by d=5
->amateur database departs from all professional ones, playing more often at shorter distances

Sizes of the three networks

-> Total number of links including degeneracies is 26 116 006, the same for all networks

->Network I: 1107 nodes, 558190 links without degeneracies

->Network II: 2051 nodes, 852578 links without degeneracies

->Network III: 193995 nodes, 7405395 links without degeneracies

->Very dense networks, especially the smallest ones

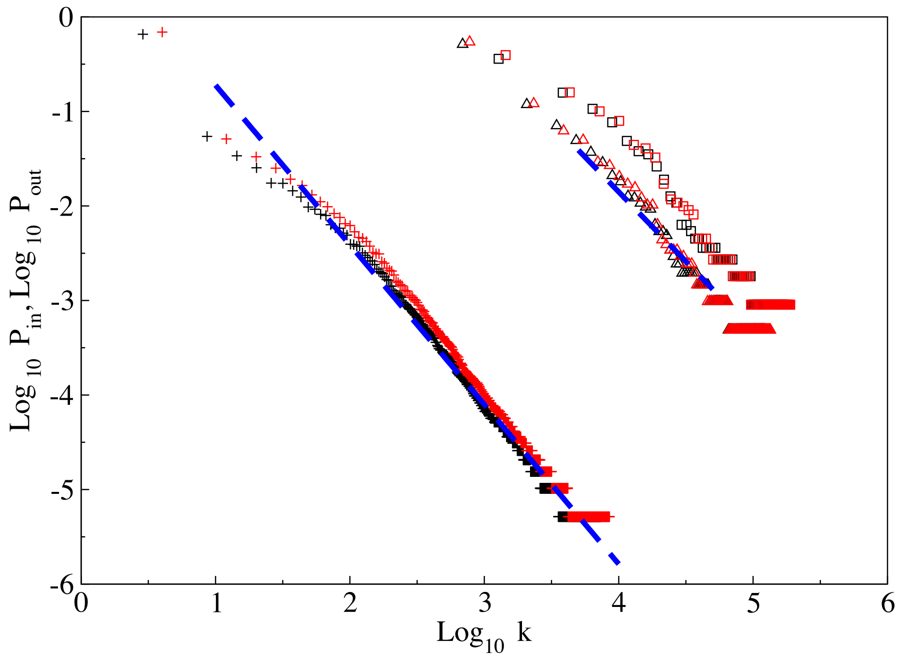
-> Very different from e.g. the World Wide Web

Link distribution

->Tails of link distributions very close to power-law for all three networks

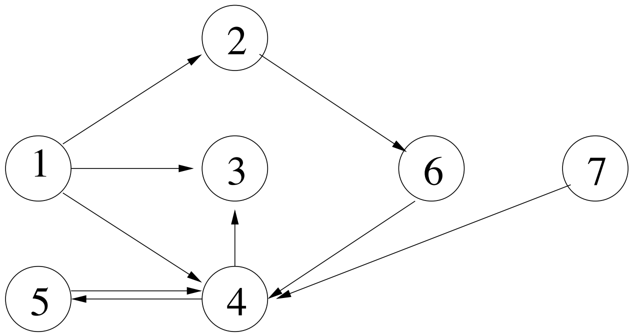
->network displays the scale-free property

->symmetry between ingoing and outgoing links is a peculiarity of this network



Normalized integrated distribution of links for the three networks

Matrix for directed networks



Weighted adjacency matrix

$$H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Google algorithm

Ranking pages $\{1, \ldots, N\}$ according to their importance.
Idea:

- The importance of a page i depends on the importance of the pages j pointing on it
- If a page has many outgoing links the importance it transmits is inversely proportional to the number of pages it points to.

PageRank p_i should thus verify

$$p_i = \sum_{j \to i} \frac{p_j}{n_j},$$

n_j= number of outgoing links of page j.
With the (stochastic) matrix H introduced above,

$$\mathbf{p} = H\mathbf{p}$$

.

Computation of PageRank

$\mathbf{p} = H\mathbf{p} \;\Rightarrow$ **p=** stationary vector of H:
can be computed by iteration of H.

To remove convergence problems:

Replace columns of 0 (dangling nodes) by $\frac{1}{N}$: $H \to$ matrix S

In our example, $H = \begin{pmatrix} 0 & 0 & \frac{1}{7} & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{7} & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{7} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{7} & 0 & 1 & 1 & 1 \\ 0 & 0 & \frac{1}{7} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{7} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{7} & 0 & 0 & 0 & 0 \end{pmatrix}$.

To remove degeneracies of the eigenvalue 1, replace S by

$$G = \alpha S + (1 - \alpha)\frac{1}{N}$$

PageRank and CheiRank

- The PageRank algorithm gives the PageRank vector, with amplitudes p_i, with $0 \leq p_i \leq 1$
- All webpages can then be ordered according to their PageRank value
- The PageRank value of a webpage can be understood as the average time a random surfer will spend there
- It ranks websites according to the number of links pointing to them which come from high-PageRank sites.

->PageRank is associated to the largest eigenvalue of the matrix G. It is based on ingoing links

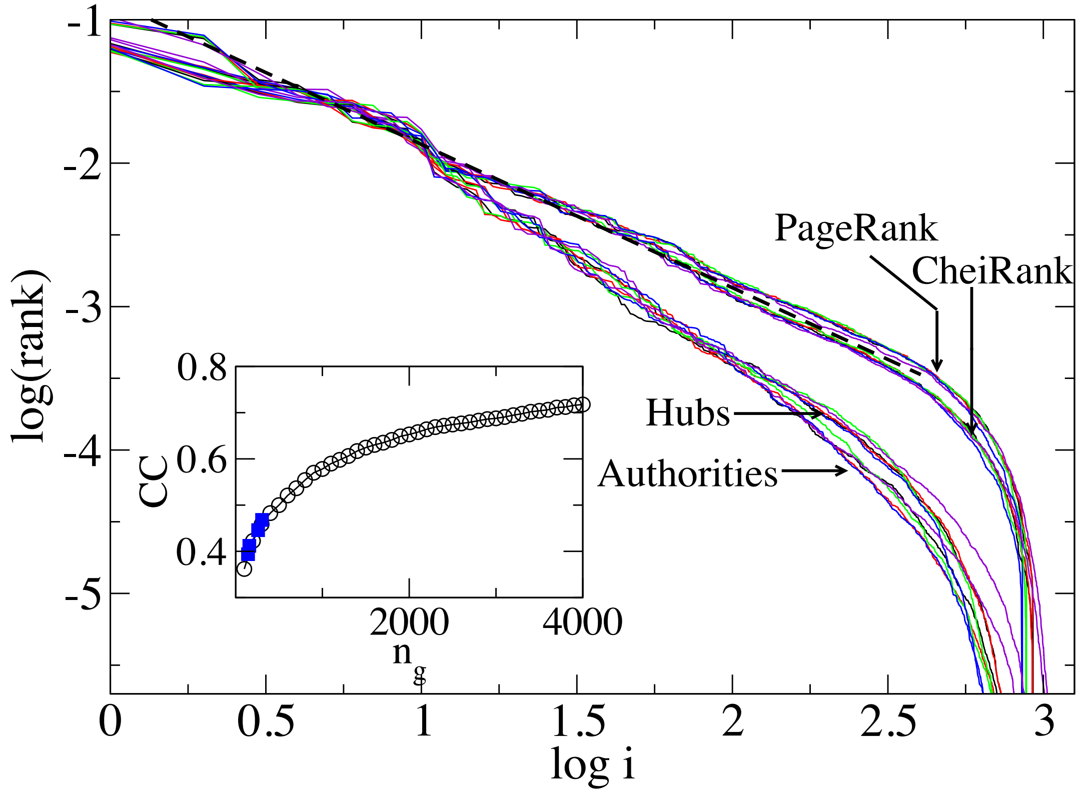
->CheiRank corresponds to the PageRank of the network obtained by inverting all links. It can be associated to a new matrix G*, and is based on outgoing links

Ranking vectors: network I

->PageRank: ingoing links
->CheiRank: outgoing links
->HITS algorithm:Authorities (ingoing links) and Hubs (outgoing links)

->Ranking vectors follow an algebraic law

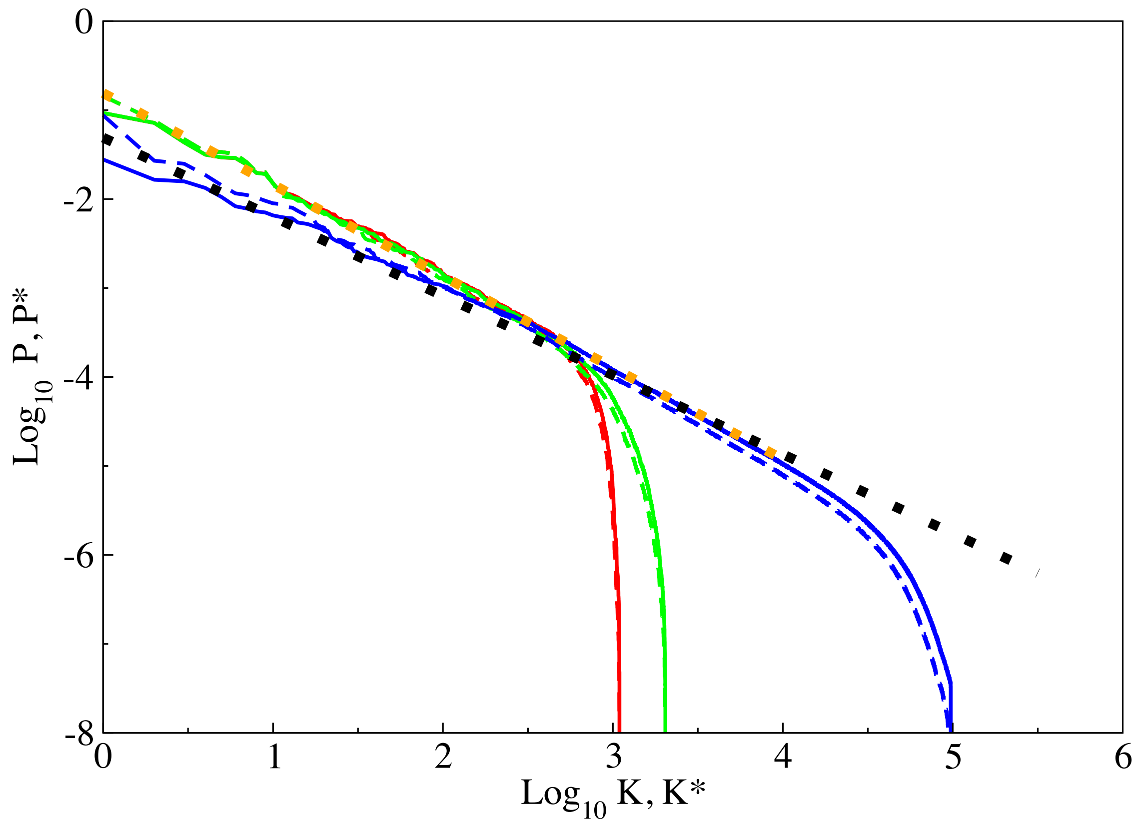
->Symmetry between distributions of ranking vectors based on ingoing links and outgoing links.



Ranking vectors: other networks

->Still symmetry between distributions of ranking vectors based on ingoing links and outgoing links.

->Power law different for the largest network



->Ranking vectors of G and G* for the three networks red: size 1107, green: size 2051, blue: size 193995.

Ranking vectors: correlations

->Strong correlations between PageRank and CheiRank

->Strong correlation between moves which open many possibilities of new moves and moves that can follow many other moves.

->However, the symmetry is far from exact.

->Correlation less strong for largest network

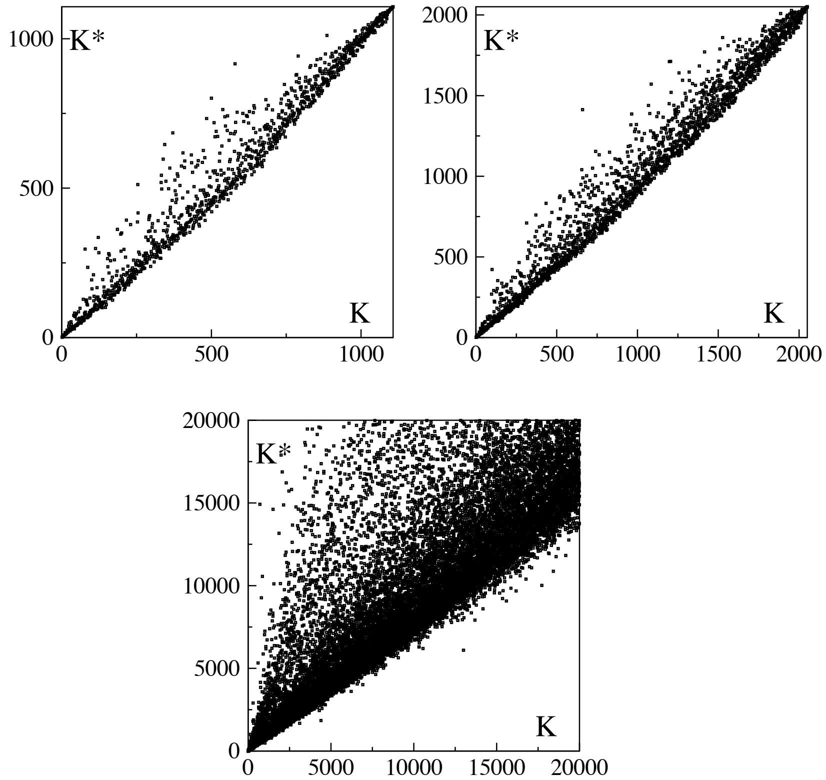


Figure: K* vs K where K (resp. K*) is the rank of a vertex when ordered according to PageRank vector (resp CheiRank) for the three networks (sizes 1107, 2051, 193995)

Ranking vectors vs most common moves

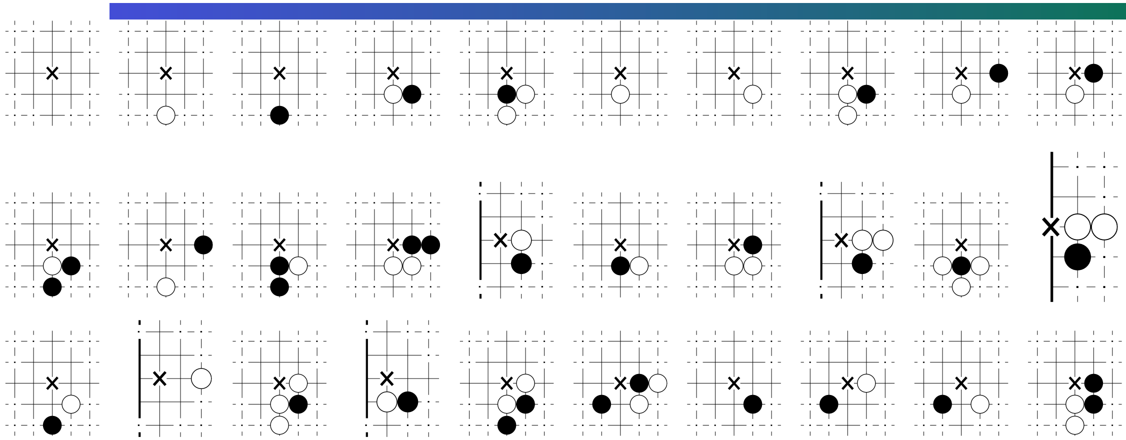
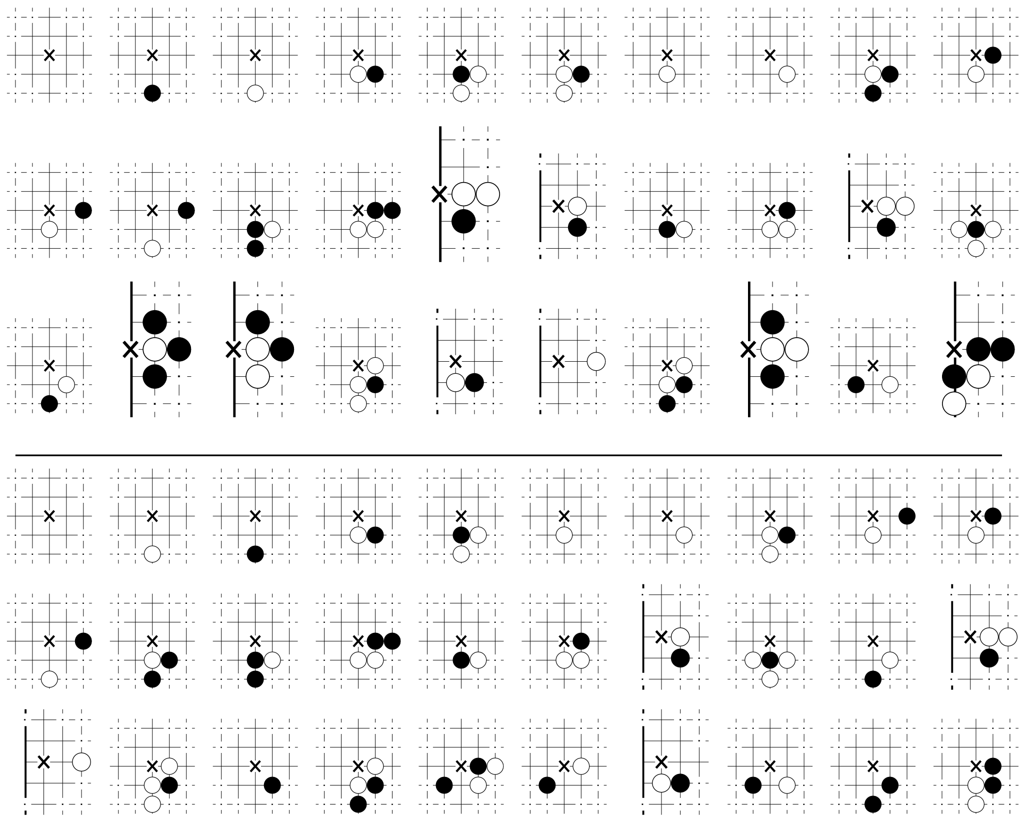


Figure: Top 30 most common moves of network III; right: top 30 PageRank and CheiRank for same network



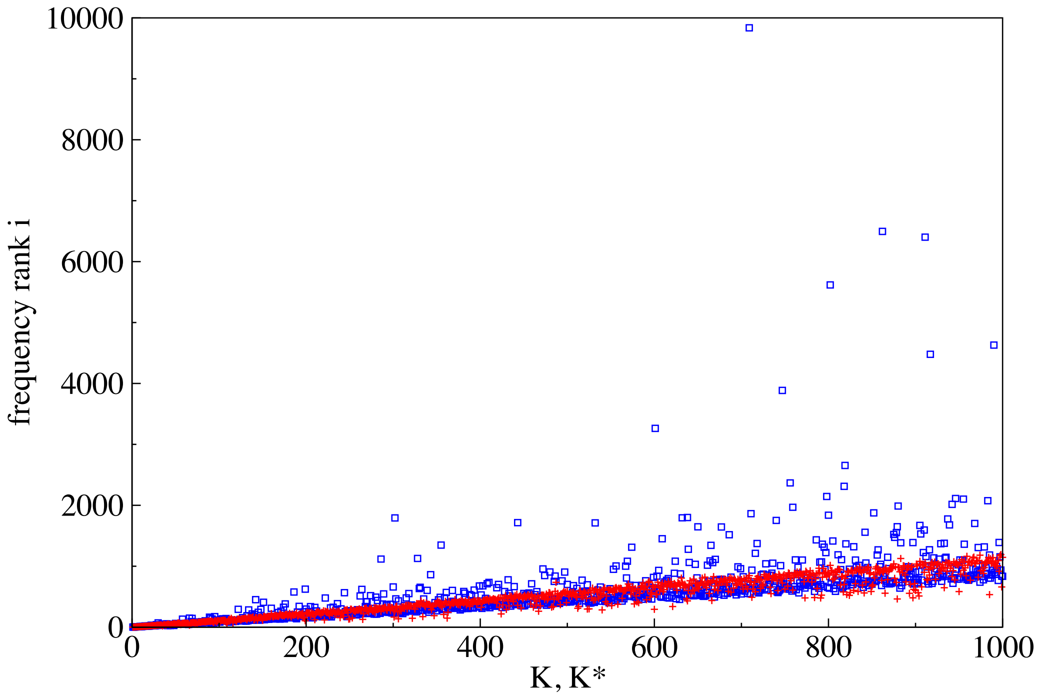
Ranking vectors vs most common moves

->There are correlations between PageRank, CheiRank, and most common moves

->However, there are also many differences, which mark the importance of specific moves in the network even if they are not that common

->Genuinely new information, which can be obtained only from the network approach

Figure: frequency rank vs PageRank (blue) and CheiRank (red) for network III



Ranking vectors vs most common moves

-> In the World Wide Web, frequency count corresponds to ranking by e. g. indegree

->PageRank takes into account indegree but weighted by importance of nodes from where the links are coming

-> Here PageRank underlines moves to which converge many well-trodden paths in the database

->CheiRank does the same in the reverse direction, highlighting moves which open many such paths

-> Could be used to bias or calibrate the Monte Carlo Go

Spectrum of the Google matrix

->For second and third networks, still gap between the first eigenvalue and next ones
->Radius of the bulk of eigenvalues changes with size of network
->More structure in the networks with larger plaquettes which disambiguate the different game paths and should make more visible the communities of moves

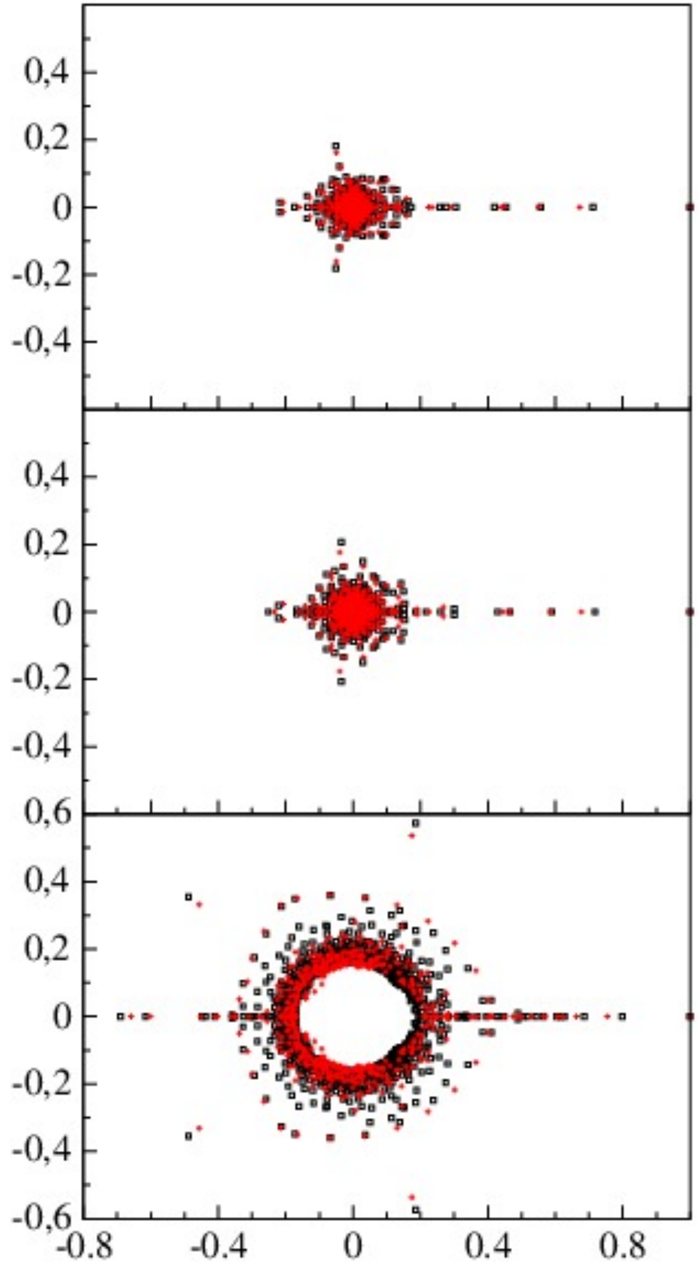


Figure: Eigenvalues of G in the complex plane for the networks with 1107, 2051 and 193995 nodes

What is the meaning of eigenvectors of the Google matrix ?

->Next to leading eigenvalues are important, may indicate the presence of communities of moves with common features.

->Indeed, eigenvectors of G for large eigenvalues correspond to parts of the network where the random surfer gets stopped for some time before going elsewhere

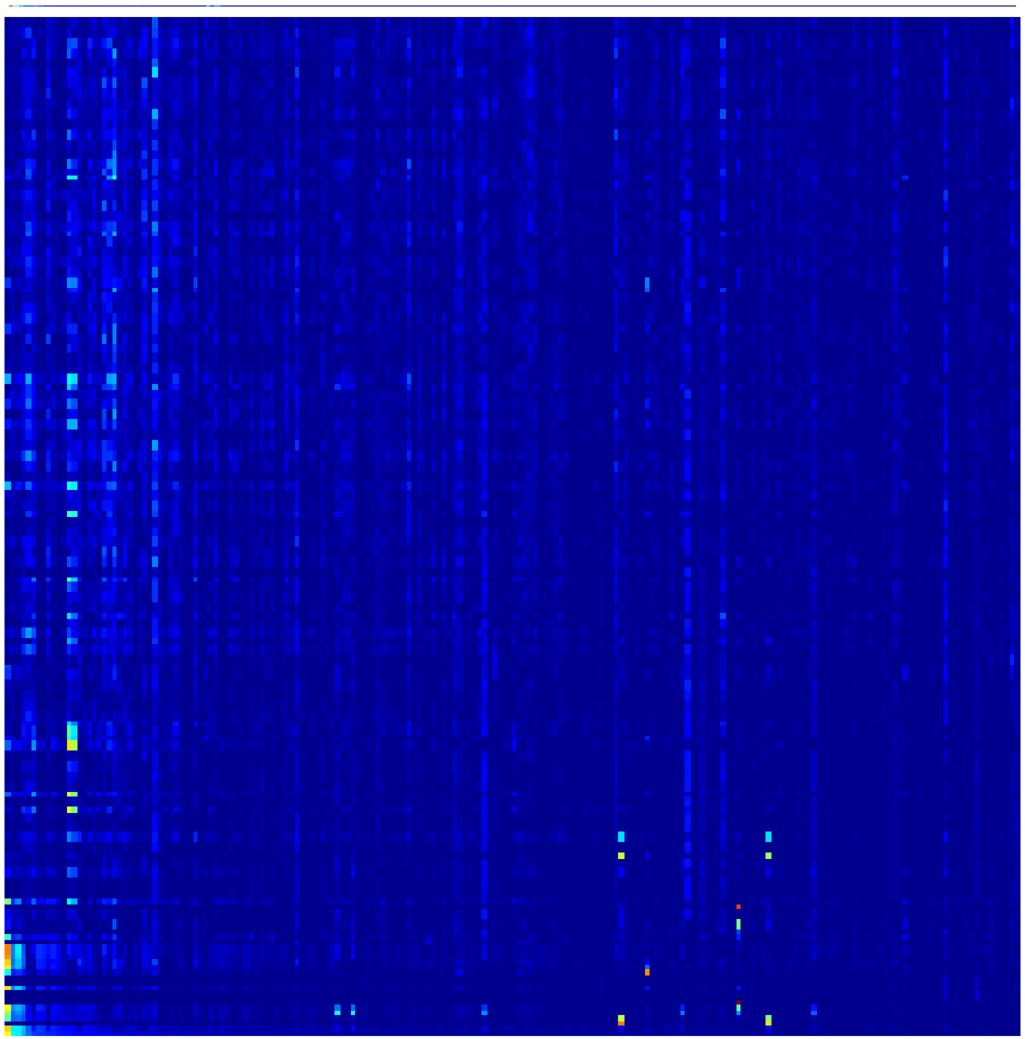
-> Correspond to sets of moves which are more linked together than with the rest of the network

-> Should indicate communities of moves which tend to be played together

Eigenvectors correlations

->Top 200 eigenvectors of diamond network in order of decreasing modulus of eigenvalue from bottom to top

->One line: one eigenvector in the order of PageRank



->Correlations visible, not necessarily related to high PageRank

Eigenvectors localization

->Inverse participation ratio: measures the spreading of eigenvectors $(\Sigma_i |\backslash P_i|^2/\Sigma_i |\backslash P_i|^4)$

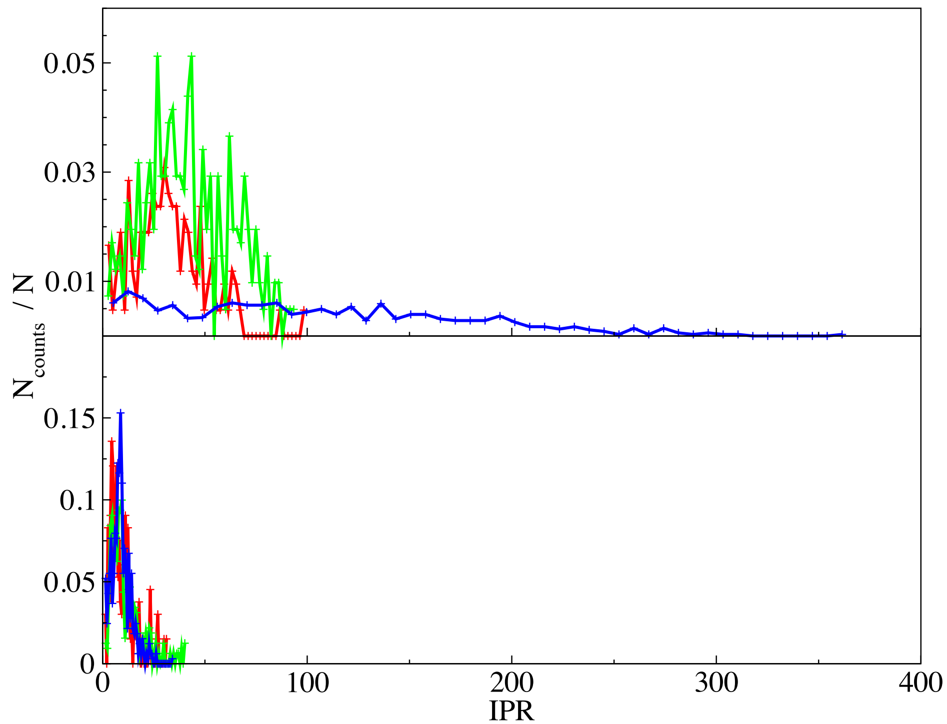
->Large dispersion for G (top)

-> Lower dispersion for G with links inverted (bottom)

->Average value quite low compared to network size

Vectors are concentrated on small parts of the network (communities)

Network of size 193995 (diamonds)

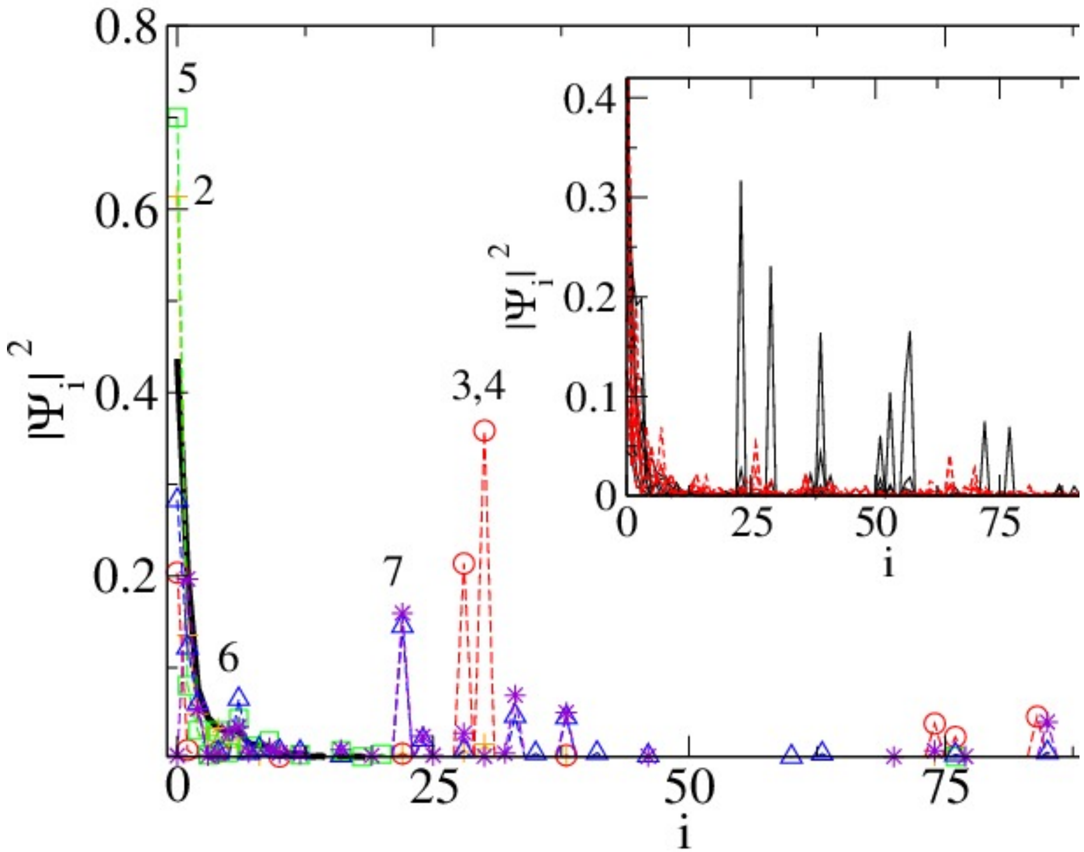


Eigenvectors for network I

->Network I: the distribution of the first 7 eigenvectors (Left) shows that they are concentrated on particular sets of moves different for each vector.

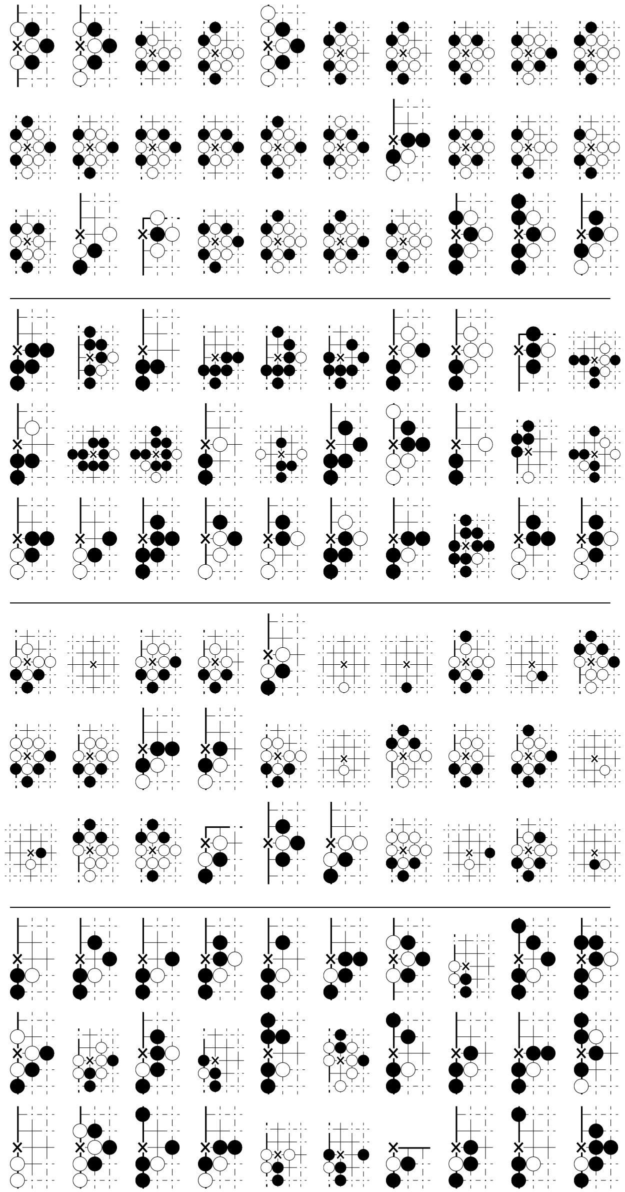
->eigenvectors are different for different tournaments and from professional to amateur

->much less peaked for randomized network



Moduli squared of the right eigenvectors of the 7 largest eigenvalues of G (network with 1107 vertices). Inset: real games (black) vs random network (red)

Eigenvectors for network III

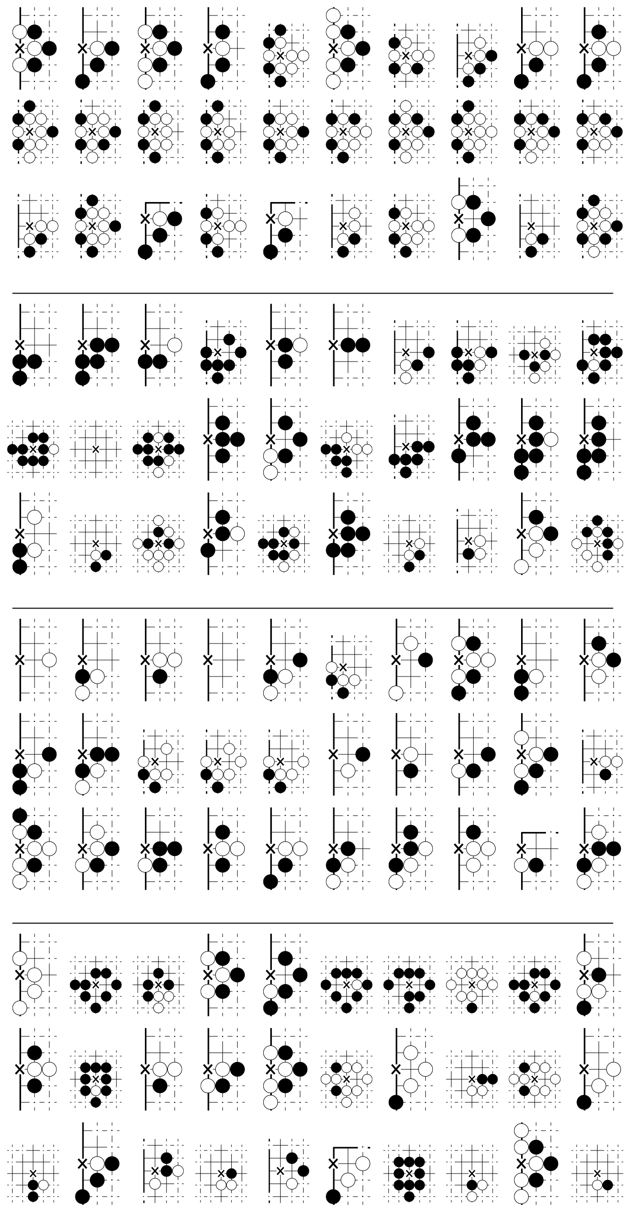


Top 30 moves

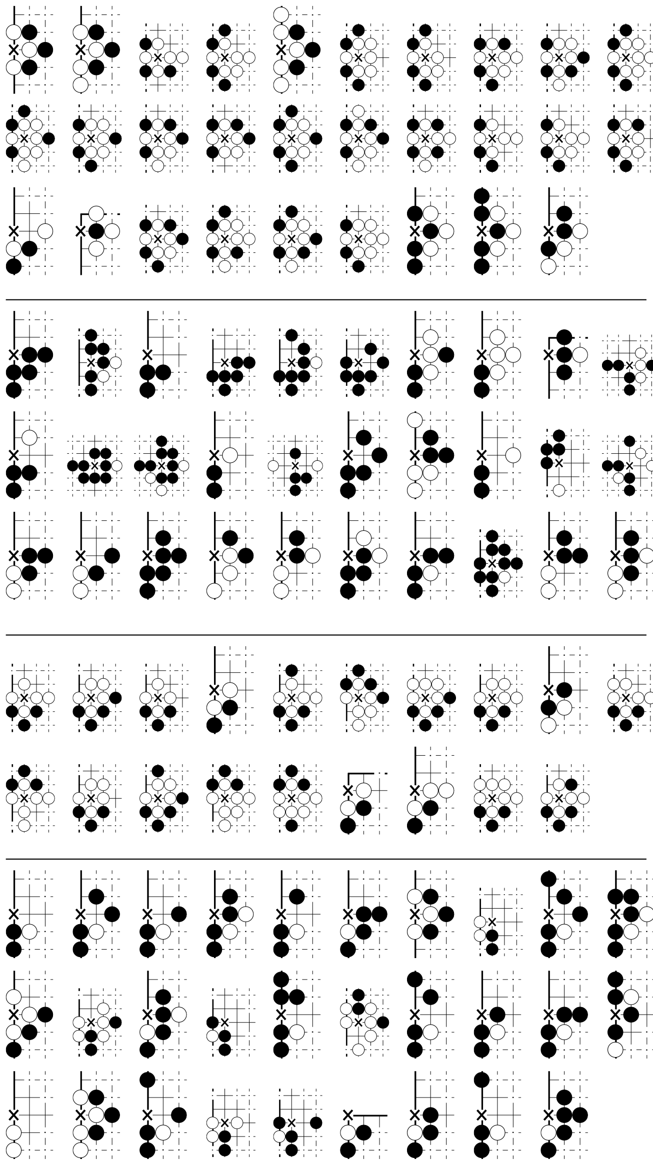
7th, 11th, 13th and 21th eigenvectors of G (left)

7th, 11th, 18th and 21th eigenvectors of G* (right)

Impression: different groups mixed in the same eigenvector



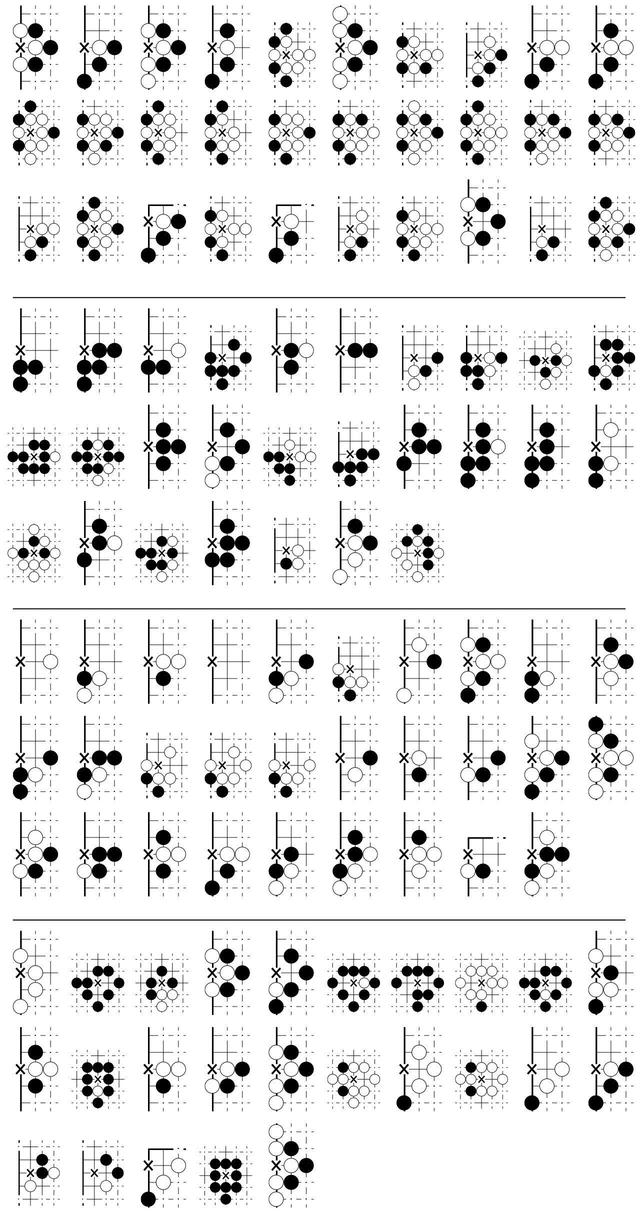
Eigenvectors-first treatment



First idea: most important moves from ranking vectorsmay appear independently of communities

->Filter out top 30 PageRank (left) or CheiRank (Right)

->More homogeneous groups, but still may be improved



Eigenvectors-second treatment

->Second idea: in the same eigenvector, several communities may coexist

-> To disentangle them, regroup moves by common ancestry: we fix a theshold of common ancestors, and add moves to the community if they share enough ancestors with one member of the community

->The threshold is a parameter which should be tuned depending on the network and the type of community searched for

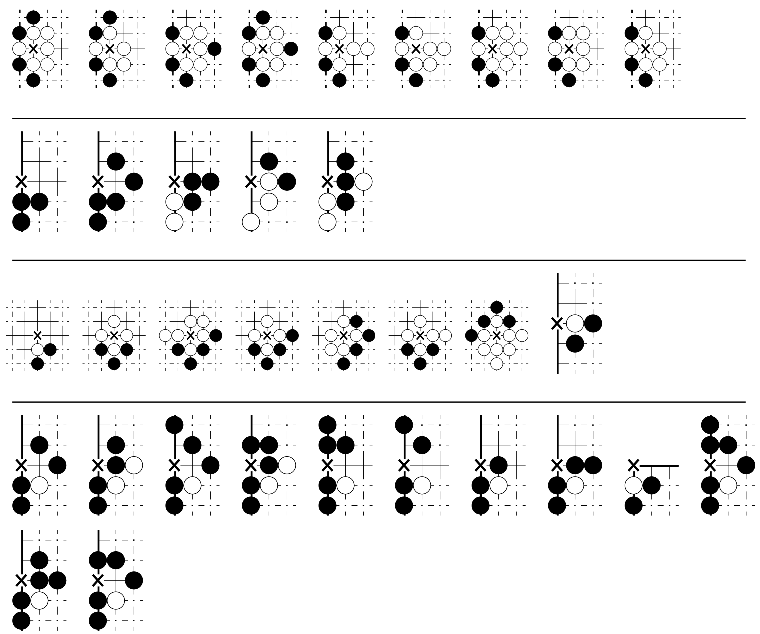
-> Such communities could be used to improve the Monte Carlo go: e.g. initialize the value of moves according to neighbours in the community, or bias the Monte Carlo towards the community

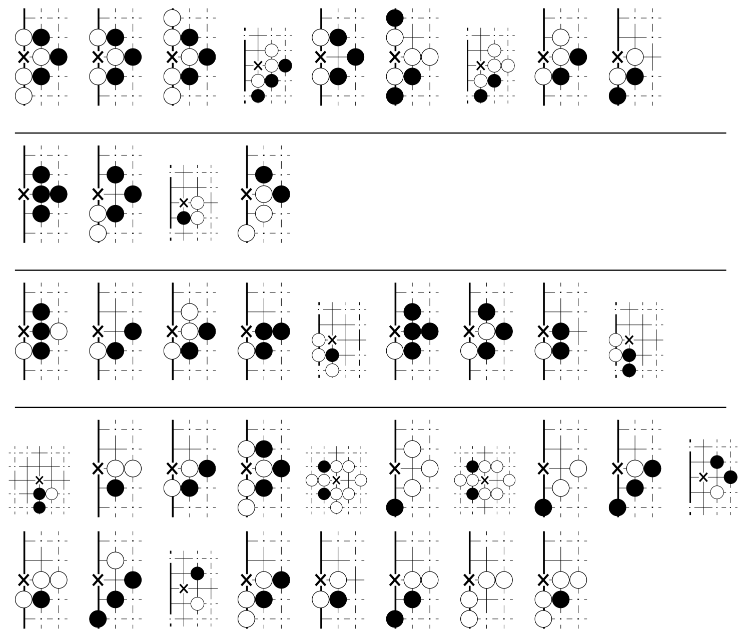
Eigenvectors-second treatment

->This method enables to extract groups of moves with common features

->Examples below for G (left) and G* (right)

->Ko (« eternity ») situations (alternate captures of opponent's stone) visible (first and third left), black connecting on side of the board (fourth left), attempts by black to takeover an opponent's chain on the rim of the board (first right)





Networks for different levels of play

->The presence of handicaps means that the winner may not be the best player

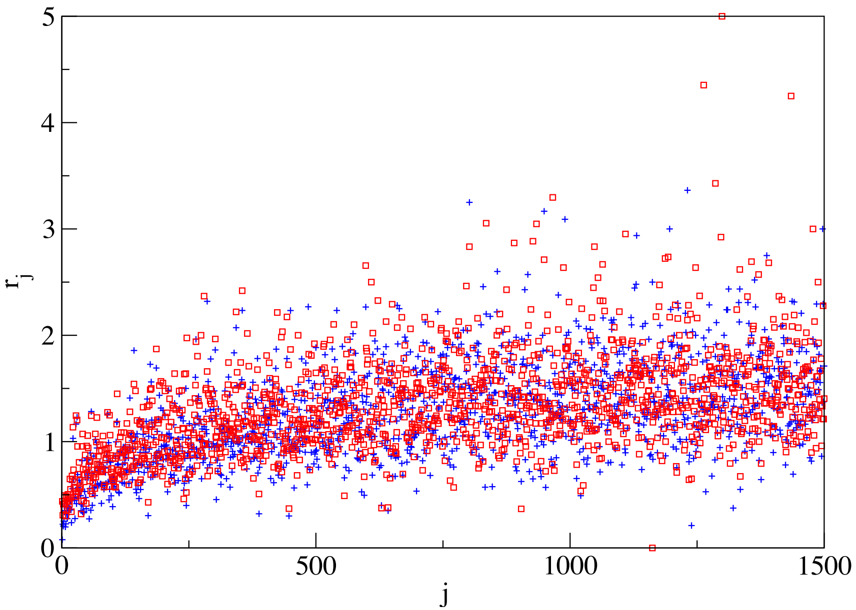
-> However, the level of players is known (number of dans)

-> One can construct networks for 1d vs 1d and compare with 9d vs 9d. We look at

$$r_j = \sum_{i \leftarrow j} |k_i - k_i'| / \sum_i k_i$$

which quantifies the difference in outgoing links between two networks

Figure: red is for 1d/1d vs 9d/9d, blue for 6d/6d Network with 193995 vertices.



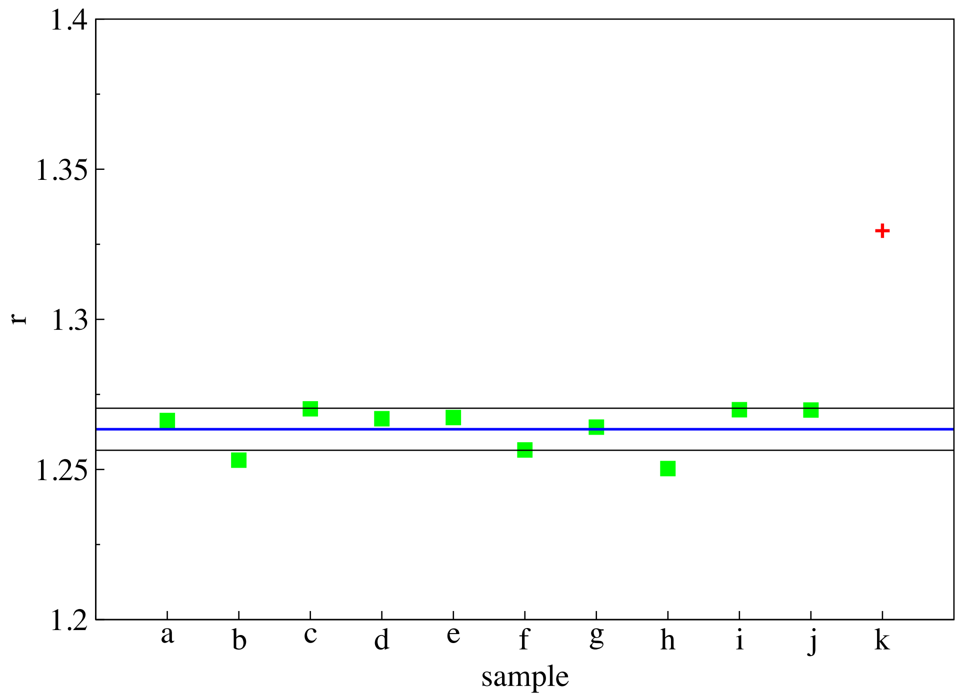
Is this difference significant?

Networks for different levels of play

-> We compared different samples of 6d/6d to the 1d/9d and computed $r = \langle r_j \rangle$ in each case

-> Result: statistically significant difference between 1d/9d and the 6d/6d samples

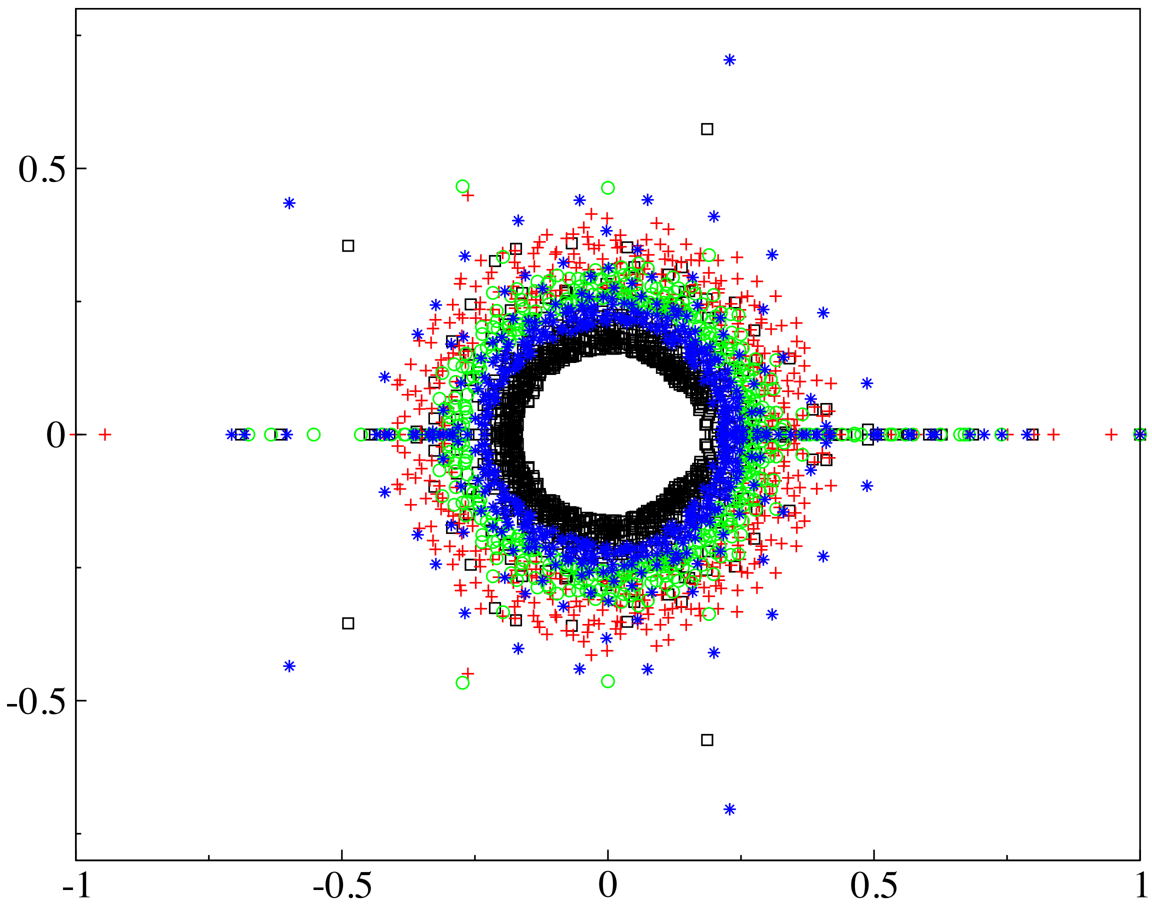
->Differences can be seen between the networks built from moves of players of different levels



Networks for different game phases

->One can separate the games into beginning, middle, and end
->The three networks are different, with markedly different spectra and eigenvectors

Figure: spectrum for all moves (black), 50 first moves (red), middle 50 (green) and last 50 (blue), Network with 193995 vertices.



Networks for different game phases

->Eigenvectors are different from those of full game network, showing specific communities

->Bias toward more empty plaquettes for beginnings, more filled plaquettes towards the end

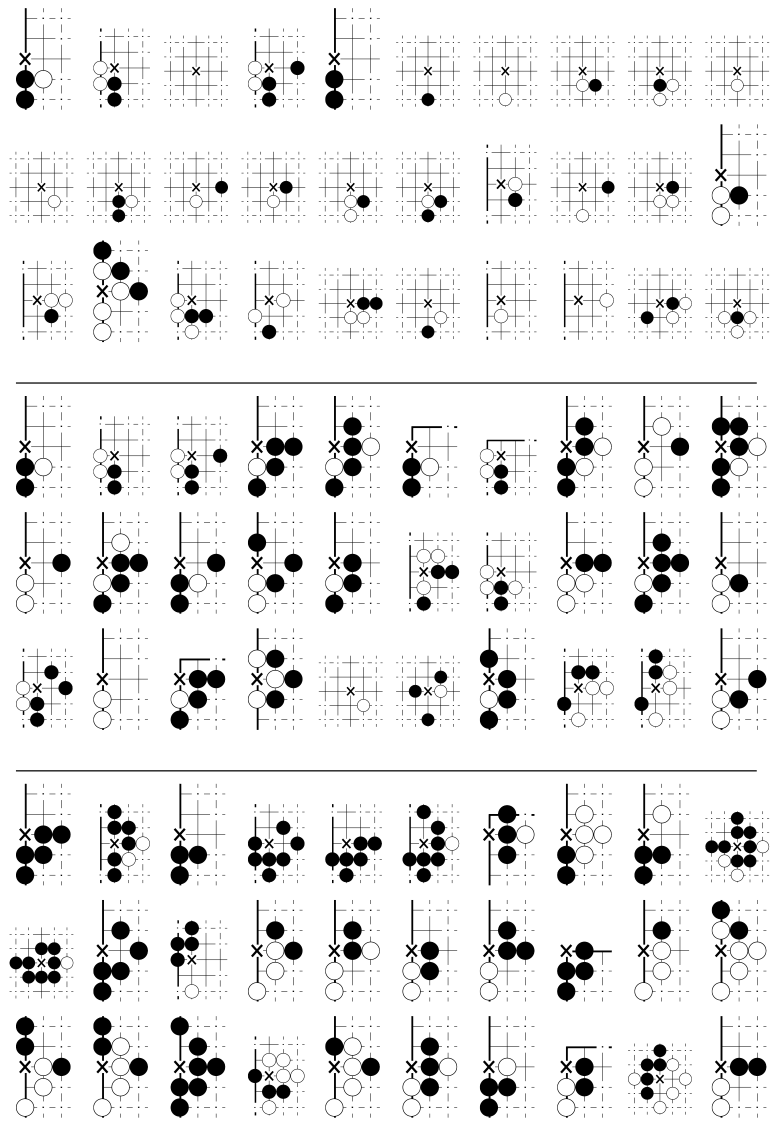


Figure: fourth eigenvector of G for 50 first moves (top), middle 50 (middle) and last 50 (bottom)

Conclusion

->we have studied the game of go, one of the most ancient and complex board games, from a complex network perspective.

->Ranking vectors highlight specific moves which are pivotal but may not be the most common

->Eigenvectors of G and G* are localized on specific groups of moves which correspond to communities of related moves

->One can construct networks for specific phases of the game or specific levels of players

-> Ranking vectors and communities could be used to improve the Monte Carlo go, currently the best go simulators

->Our approach could be used for other types of games, and in parallel shed light on the human decision making process.