

Pajek

versione dalla 3 in poi

Download:

<http://mrvar.fdv.uni-lj.si/pajek/>

Fernanda Strozzi

L'Università Carlo Cattaneo



Pajek significa «ragno» in Sloveno.

Pajek è un software per la visualizzazione e l'analisi delle reti complesse.

E' continuamente esteso e aggiornato.

Gli autori sono: Vladimir Batagelj e Andrej Mrvar dell'università di Ljubijana (Slovenia)

manuale



Pajek and Pajek-XXL

Programs for Analysis
and Visualization
of Very Large Networks

Reference Manual

List of commands with short explanation

version 3.13

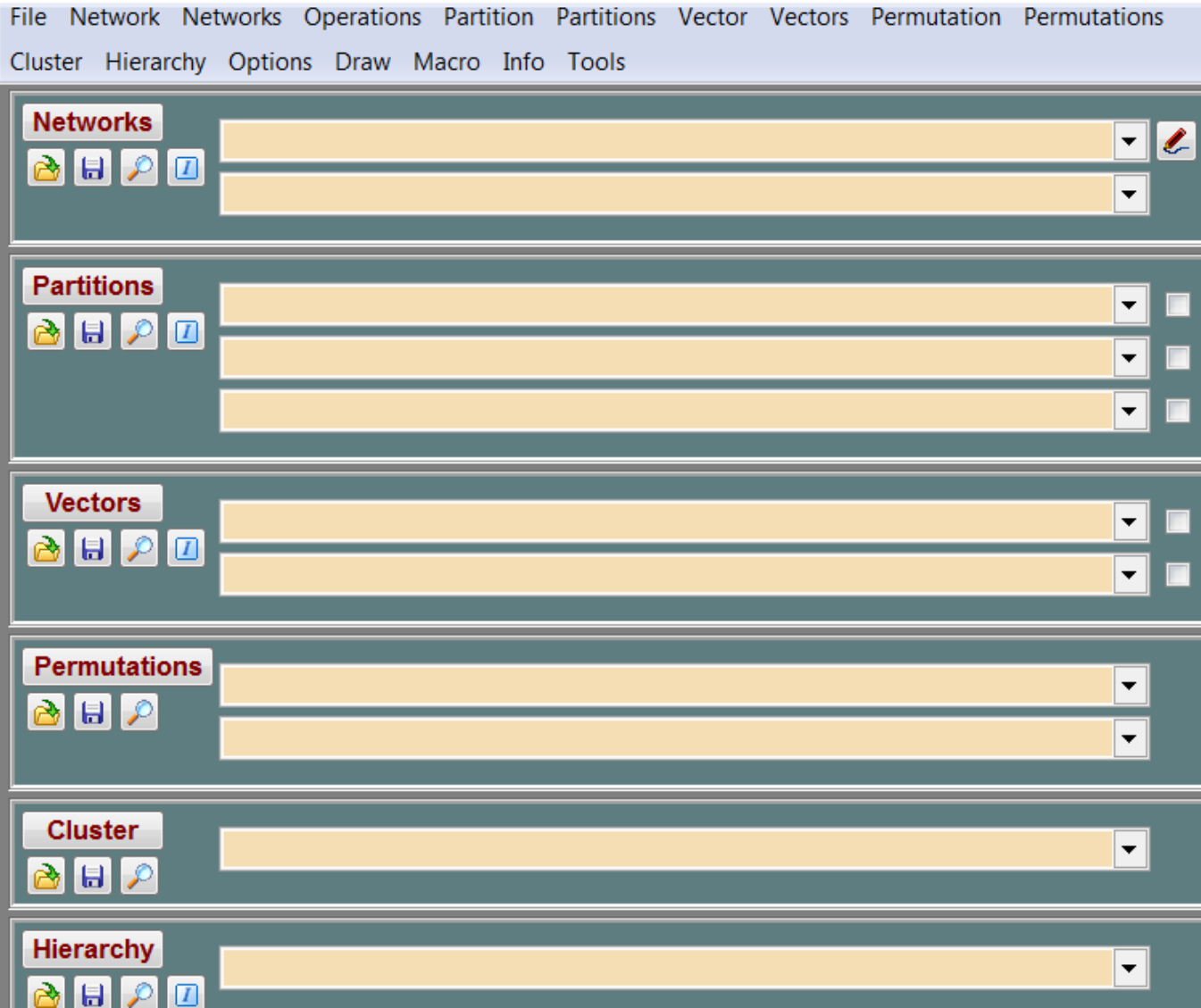


Meglio la più recente disponibile

Andrej Mrvar and Vladimir Batagelj
Ljubljana, September 1, 2013

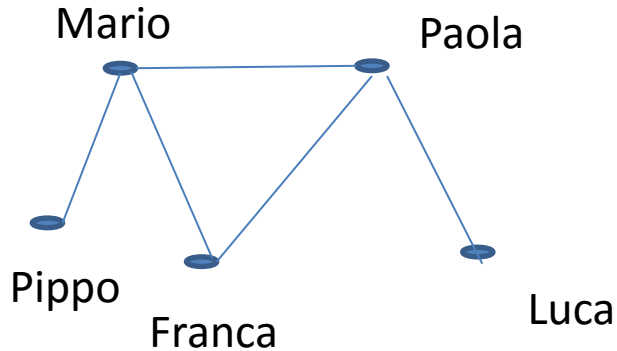
Vedremo come fare con Pajek a:

- introdurre reti
- rappresentare graficamente le reti
- Statistica di base



**Finestra Principale
Pajek64 (o 32)
non XXL**

Esercizio 0: inserire una rete



Network/CreateRandomNetwork/
Total n° of Arcs. → [file.net](#)

Salvarlo ; aprirlo; modificare il
file manualmente con un word
processor; risalvarlo; caricarlo

```
*Vertices 5  
1 «Pippo»  
2 «Franca»  
3 «Luca»  
4 «Paola»  
5 «Mario»  
*matrix  
0 0 0 0 1  
0 0 0 1 1  
0 0 0 1 0  
0 1 1 0 1  
1 1 0 1 0
```

```
*Vertices 5  
1 «Pippo»  
2 «Franca»  
3 «Luca»  
4 «Paola»  
5 «Mario»  
*Edgeslist  
1 5  
2 5 4  
3 4  
4 5
```

```
*Vertices 5  
1 «Pippo»  
2 «Franca»  
3 «Luca»  
4 «Paola»  
5 «Mario»  
*Edges  
1 5 1  
2 5 1  
2 4 1  
3 4 1  
4 5 1  
*arcs
```

1. Lista di vertici seguita da
una lista di arcs/edges

2. Lista di vertici seguita da una
matrice.

Limitazioni: non è possibile
indicare linee multiple
Linee di valore 0 = non linee

3. Lista di arcs/edges (peso 1)

Esercizio 0: Costruire una partizione o un vettore

Sex.clu

*vertices 5

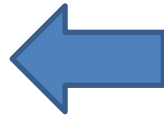
1

2

1

2

1



Create partition (male/female)

Partition/Create Constant Partition → **file.clu**

Salvarlo ; aprirlo; modificare il file
manualmente con un word processor;
risalvarlo; caricarlo

age.vec

*vertices 5

15

10

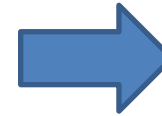
10

8

7

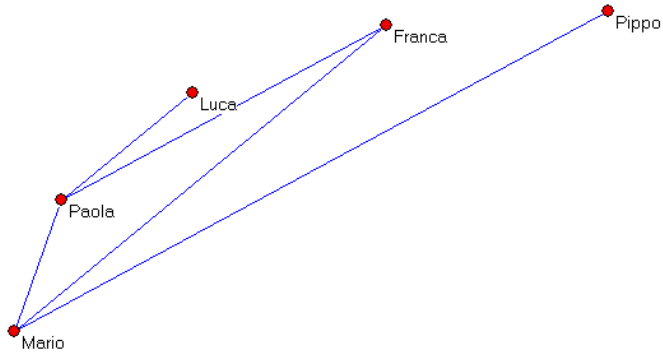
Vector/Create Constant Vector → **file.vec**

Salvarlo ; aprirlo; modificare il file manualmente
con un word processor; risalvarlo; caricarlo

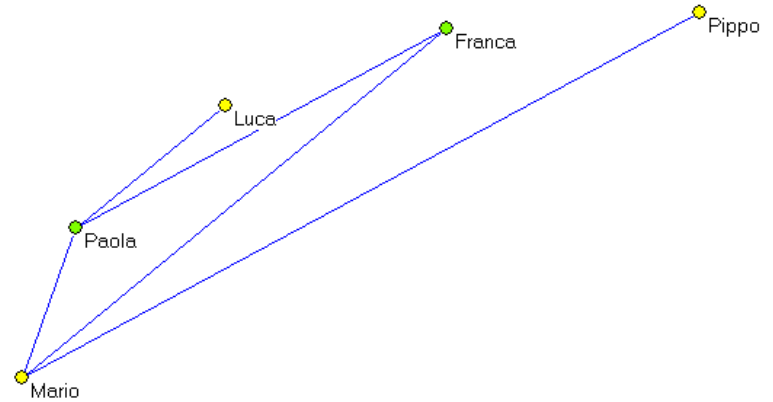


Esercizio 0: disegnare la rete le partizioni e i vettori

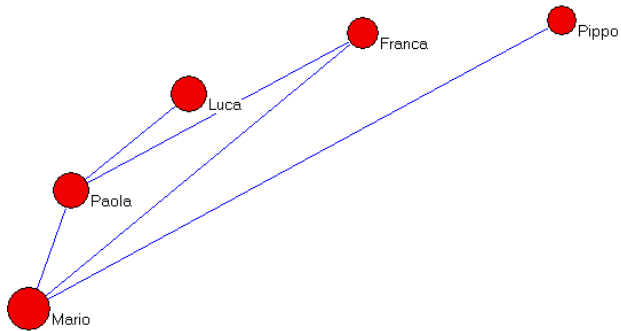
Draw/network



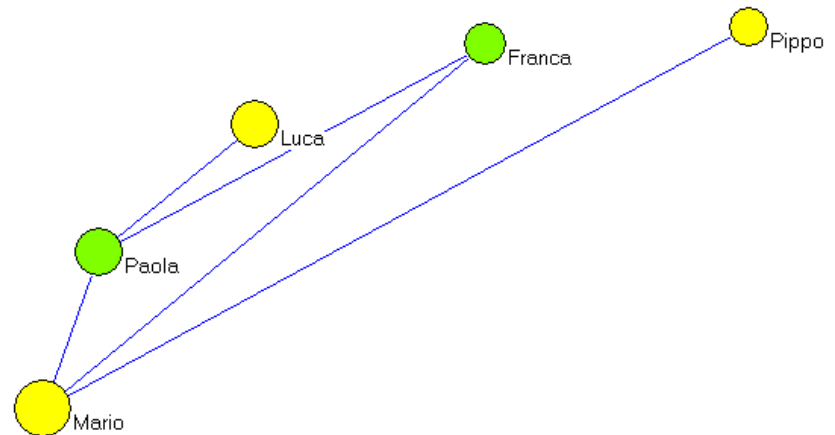
Draw/Network + First Partition



Draw/Network + First Vector

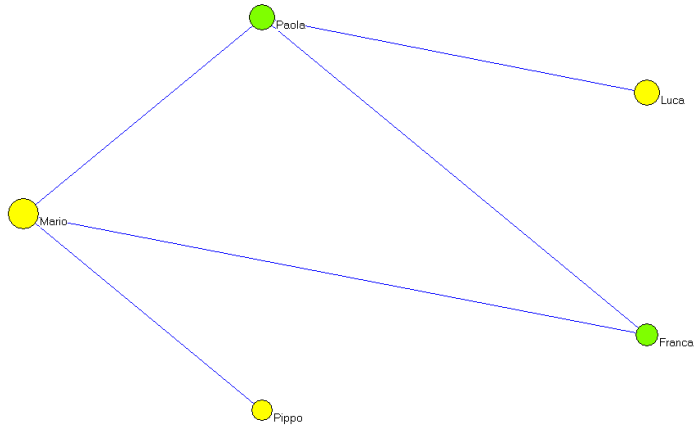


Draw/Network + First Partition + First Vector

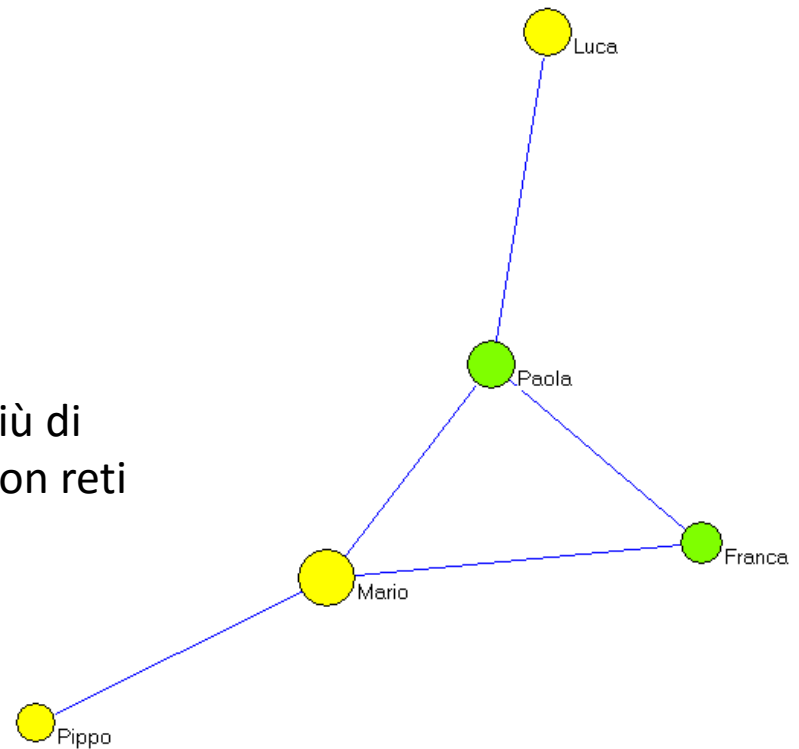


Esercizio 0: disegnare la rete con diversi layout

Layout/Circular/Original



Layout/Energy/Kamada-Kaway/Free

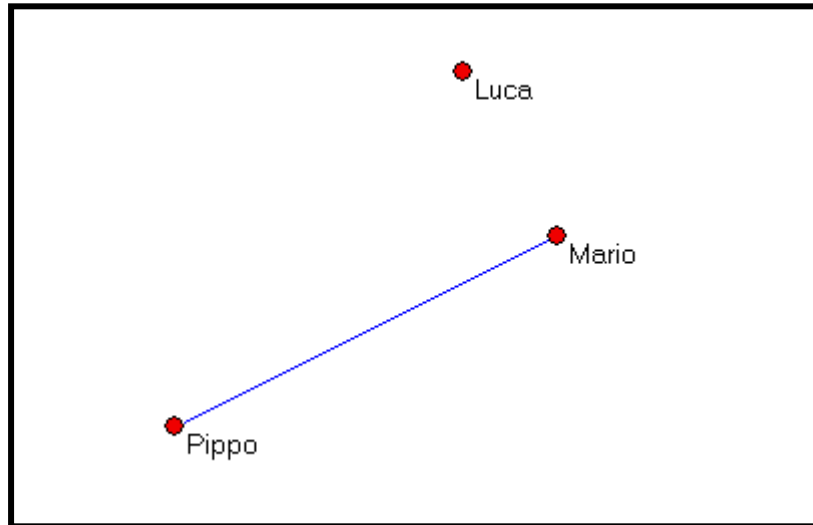


Per default Pajek non disegna reti con più di 5000 vertici anche se può fare i calcoli con reti con più di 9.999.997 vertici.

Esercizio 0: Estrarre una partizione

Partition/Info

Operations/Network + Partition/Extract Subnetwork → 1



La posizione dei vertici può essere cambiata usando il mouse.

Si possono usare le *Option* dalla finestra grafica per visualizzare diversamente i nodi e gli archi.

Statistica di base

Nodi: 5

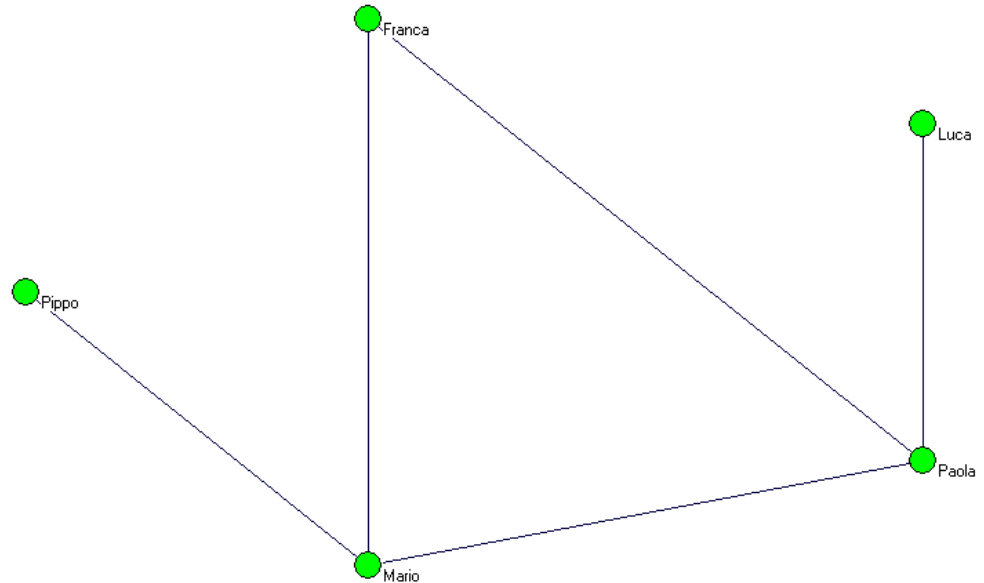
Edges: 5

Grado medio: 2 ($2 * m / n$)

diametro: 3 (Pippo-Luca)

Distanza media: 1.6

Clustering: 0.55



- Network/Info/general
- Network/create vector/distribution of distances
- Network/create vector/Clustering coefficients

Pajek

Esercizio Partizioni

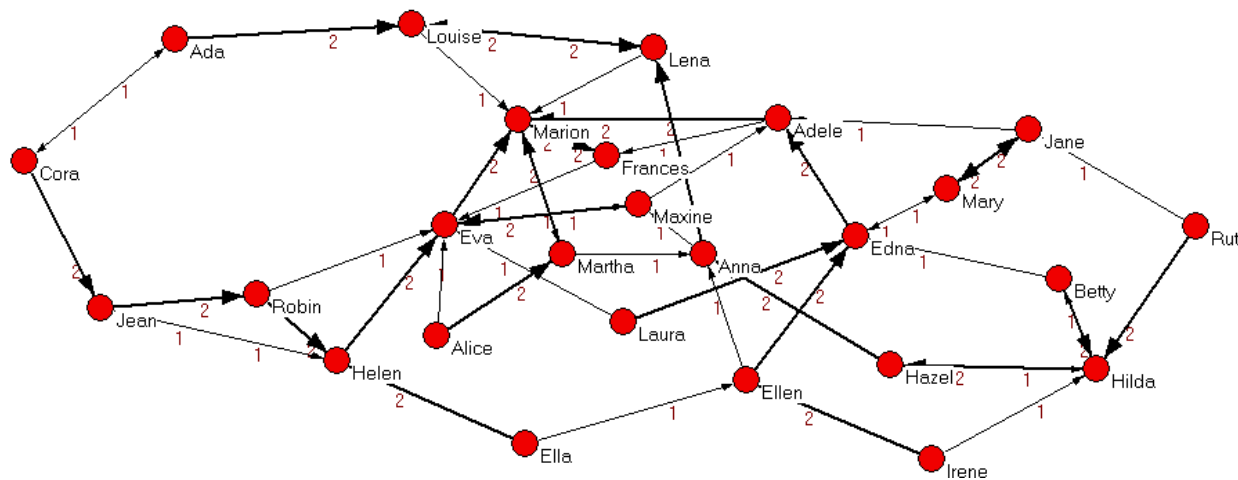
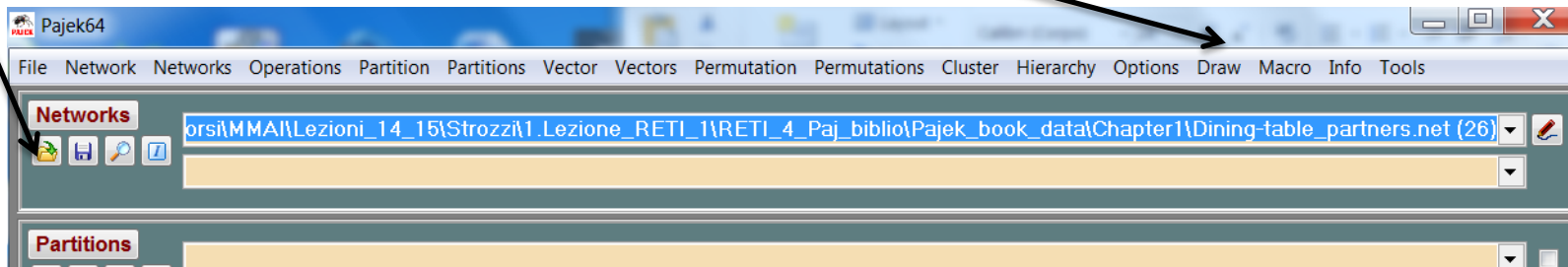
Dining-table_partners.net:

Aprire la rete e disegnarla

Dining-table_partners.net:
Supponiamo che ogni ragazza condivida il suo piatto (e ogni piatto che è stato condiviso con lei) solo con la sua 1 e 2 scelta

Draw network

Load network



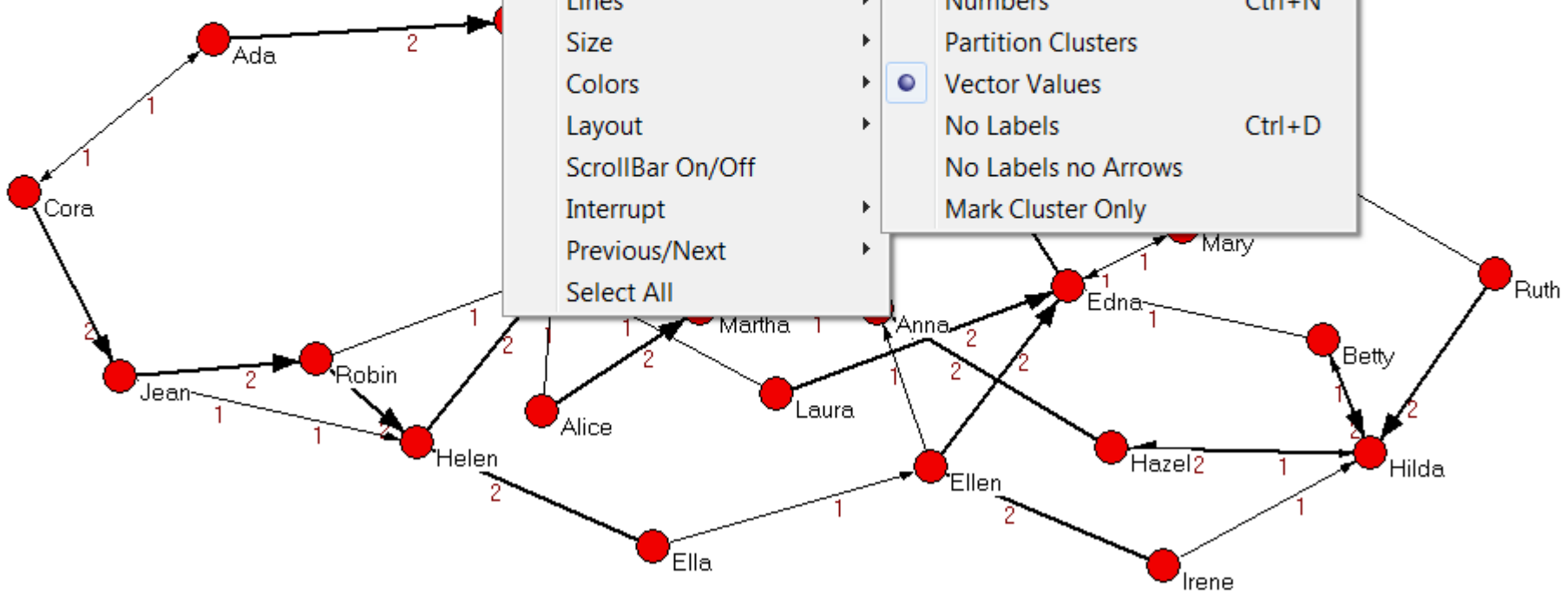
Dining-table_partners.net: 26 vertices, 52 valued arcs (1 = first choice, 2 = second choice), no edges, no loops.

Drawing Window: options

1. C:\Users\fstrozzi\BU_25_7_2014\corsi\MMAI\Lezioni_14_15\Strozzi\1.Lezione_RETI_1\RETI_4_Paj_biblio\Pajek_book...

Layout GraphOnly Previous Redraw Next Options Export Spin Move Info

- Transform
 - Values of Lines
 - Mark Vertices Using
 - Lines
 - Size
 - Colors
 - Layout
 - ScrollBar On/Off
 - Interrupt
 - Previous/Next
 - Select All
- Labels Ctrl+L
 - Numbers Ctrl+N
 - Partition Clusters
 - Vector Values
 - No Labels Ctrl+D
 - No Labels no Arrows
 - Mark Cluster Only



Network/Info/General fornisce informazioni generali sulla rete

```

Report
File
=====
3. C:\Users\fstrozzi\BU_25_7_2014\corsi\MMAI\Lezioni_14_15\Strozzi\1.Lezione_RETI
=====
Number of vertices (n): 26
-----
                Arcs      Edges
-----
Number of lines with value=1      26      0
Number of lines with value#1      26      0
-----
Total number of lines              52      0
-----
Number of loops                    0      0
Number of multiple lines           0      0
-----
Density1 [loops allowed]    = 0.07692308
Density2 [no loops allowed] = 0.08000000
Average Degree = 4.00000000
    
```

← $2m/(n*n)$ link esistenti/link possibili (con loops)

← $2m/(n*(n-1))$ link esistenti/link possibili (no loops)

← $2m/n$

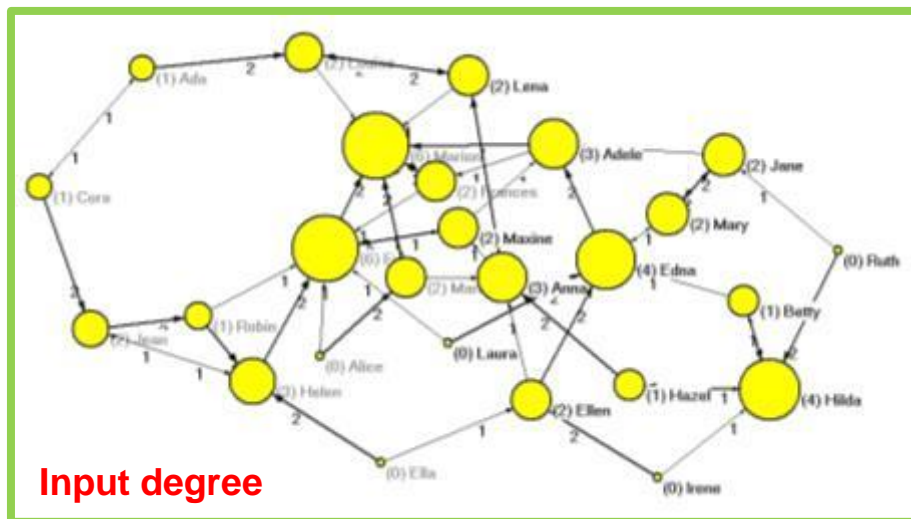
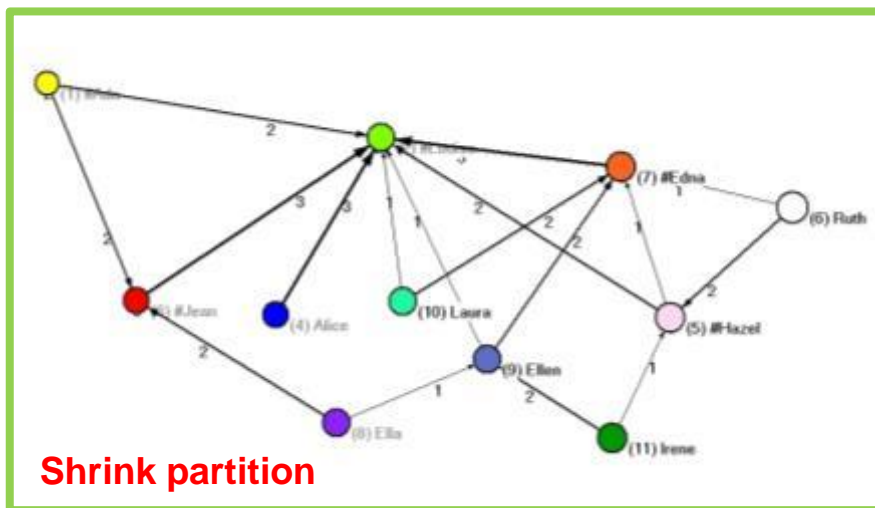
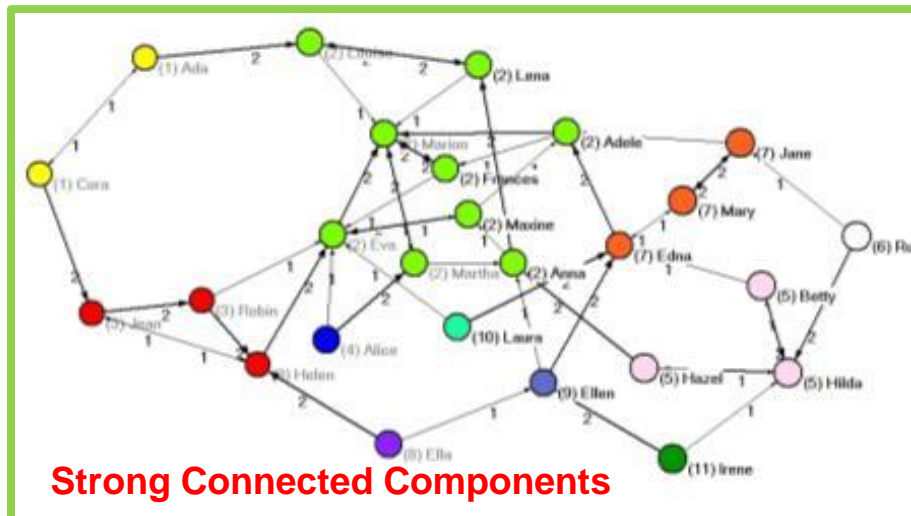
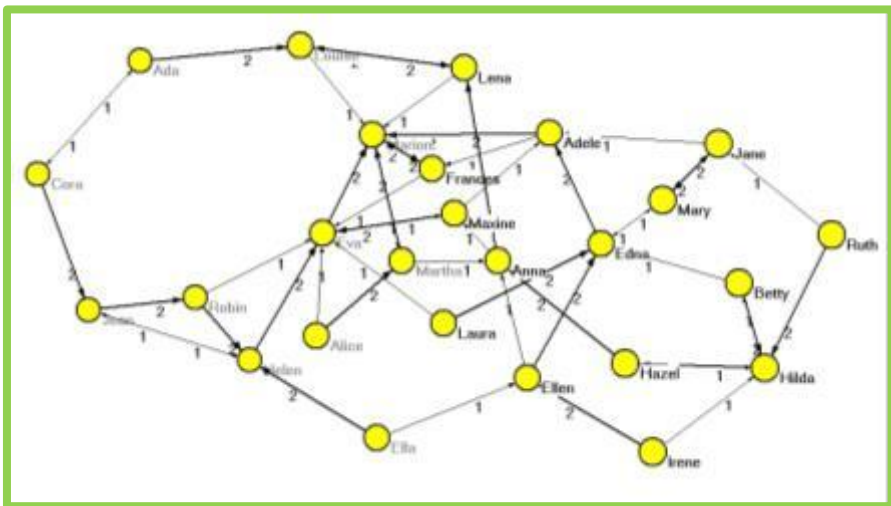
m=numero link con peso 1=26
n=numero nodi=26

Partizioni e Difusione

Datasets:

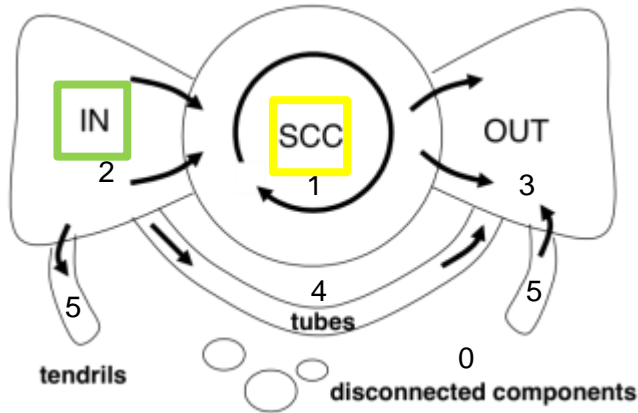
Dining-table_partners.net

RETE DIRETTA



Partizioni e Difussione

The bow-tie model of the Web
Broder et al. (1999)



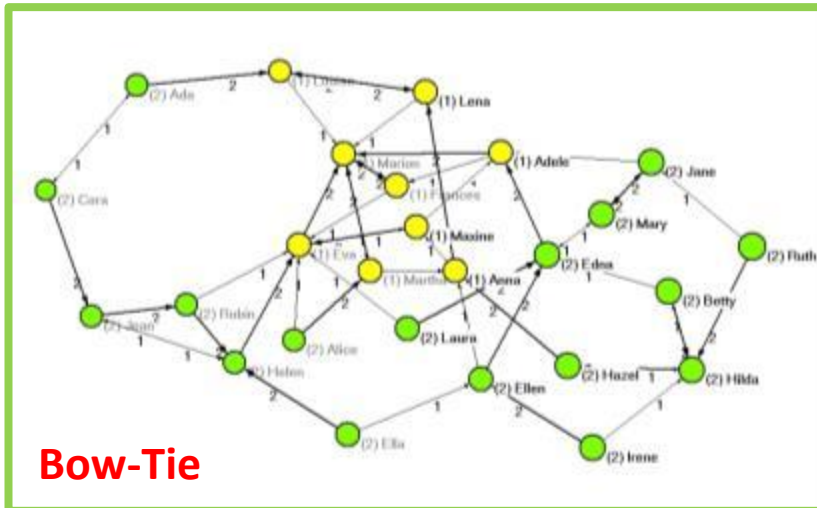
In Pajek



Network>Create Partition>Bow-Tie

```
Coding of classes in parititon
-----
1 - LSSC: Largest Strongly Connected Component
2 - IN
3 - OUT
4 - TUBES
5 - TENDRILS
0 - OTHERS
-----
```

- **SCC** (strongly connected component):
 - can reach all nodes from any other by following *directed* edges
- **IN**
 - can reach SCC from any node in 'IN' component by following directed edges
- **OUT**
 - can reach any node in 'OUT' component from SCC
- **Tendrils and tubes**
 - connect to IN and/or OUT components but not SCC
- **Disconnected**
 - isolated components

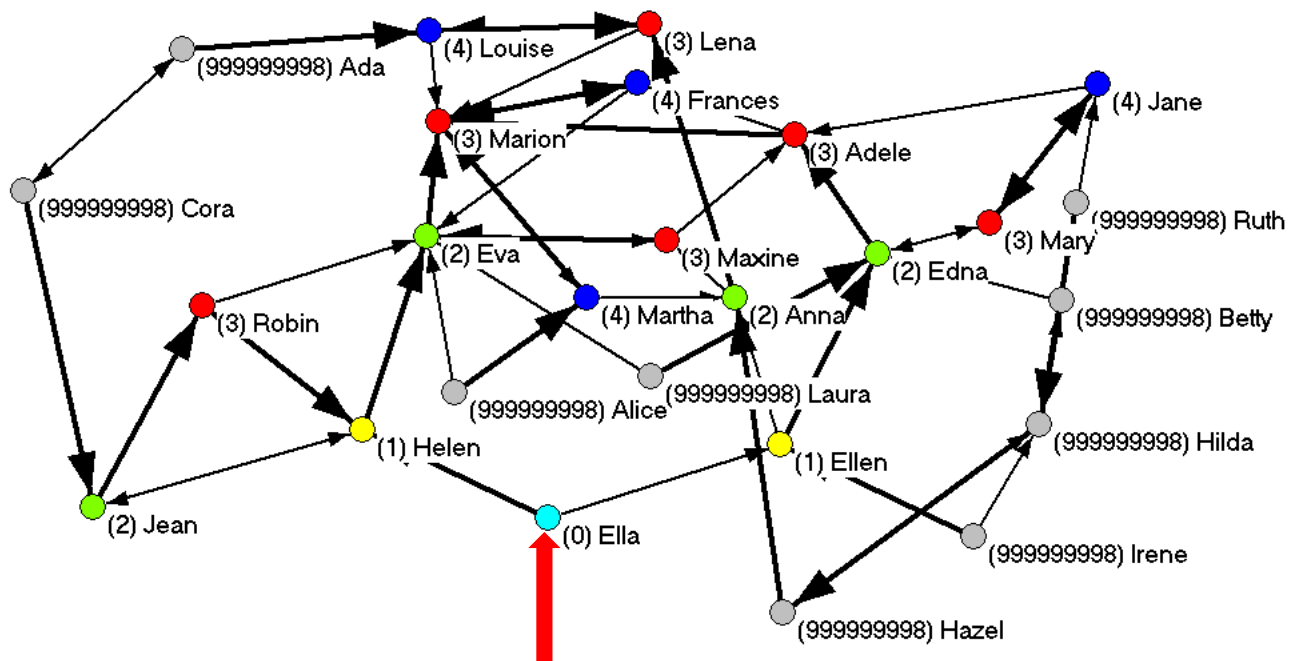


Bow-Tie

Partizioni e Difusione

Flusso di Informazioni= Diffusione di malattie= tecnica campionamento snowball

K – neighborhood - output



SEED (23)

Pajek

Misure di centralità e centralizzazione

Strike.net, Strike_Groups.clu

Esercizio 3: Rete di comunicazione informale

Datasets: Strike.net, Strike_Groups.clu

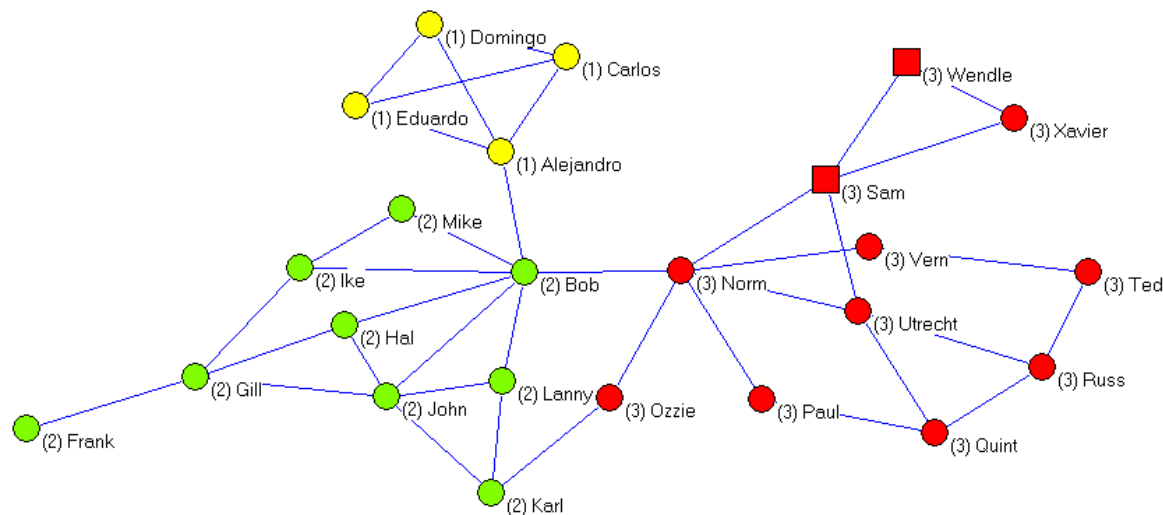
Trovare le persone centrali e quelle periferiche in una rete sociale, dal punto di vista della comunicazione.

La rete Strike.net rappresenta le rete di comunicazione in una piccola impresa : una segheria.

Nella segheria la nuova gestione ha introdotto una modifica nei compensi dei lavoratori che non è stata accettata dai lavoratori stessi e che quindi hanno incominciato a scioperare arrivando ad una situazione di stallo negoziale. I nuovi gestori hanno chiesto quindi ad un consulente esterno di analizzare la struttura della comunicazione tra i dipendenti in quanto si sospettava che le reali modifiche contrattuali non fossero state comunicate correttamente dai vari negoziatori sindacali.

Il consulente ha costruito la rete di comunicazione informale come segue: a tutti i lavoratori è stato chiesto di indicare la frequenza con cui discutevano di problemi di lavoro con i loro colleghi usando una scala da 1 a 5 che rappresentava la frequenza di comunicazione da meno di una volta a settimana a diverse volte al giorno. Sono state scelte le persone che comunicavano tra loro con un valore di scala almeno uguale a 3.

La rete mostrava demarcazioni abbastanza stringenti tra gruppi definiti per età e lingua. I giovani lavoratori spagnoli (al più hanno 30 anni) comunicavano tra loro e con un giovane inglese. I giovani inglesi con solo due inglesi anziani. Tutti i legami tra i gruppi avevano delle giustificazioni sociali. Tra gli ispanici, Alejandro era il più competente in inglese e Bob parlava un po' di spagnolo, il che spiegava il loro legame. Bob doveva a Norm il suo posto di lavoro e, probabilmente, a causa di questo, avevano sviluppato un legame di amicizia. Infine, Ozzie era il padre di Karl.



Spagnoli < 30 anni

Ingleses < 30 anni

Ingleses > 30 anni

Informazioni generali sulla rete

Number of vertices (n): 24

Number of edges: 38

No loops, no multiple lines

Average degree=3.17

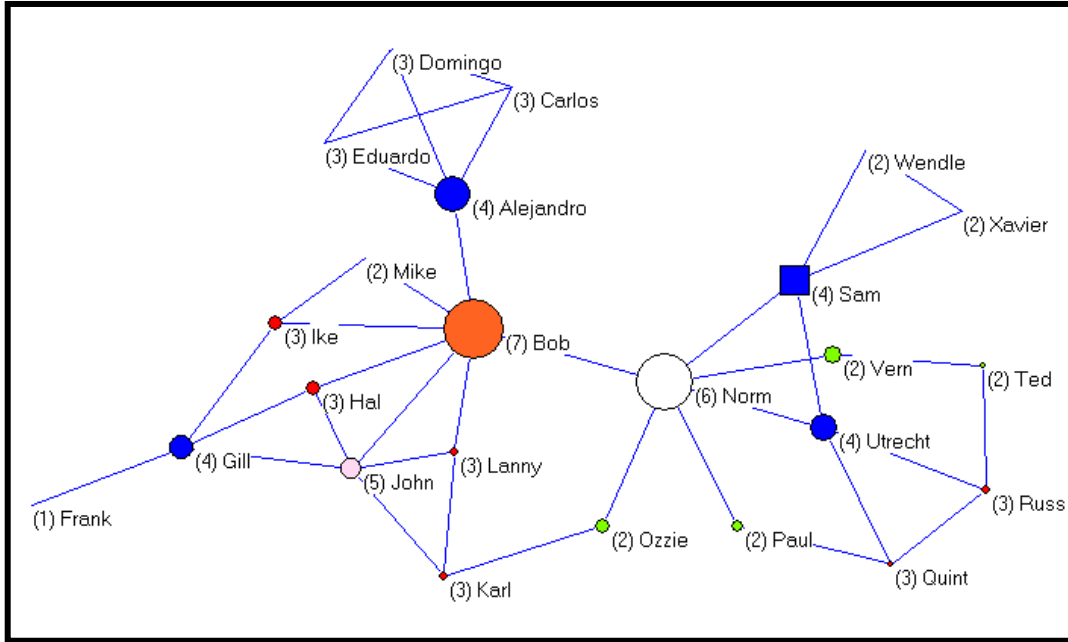
Density=1.38

Diameter=6 (From Frank(1) to Russ (20))

Average Distance=2.99

N.B. Se dovessero essere presenti loop o linee multiple, le si deve togliere prima di calcolare le misure di centralità dei nodi

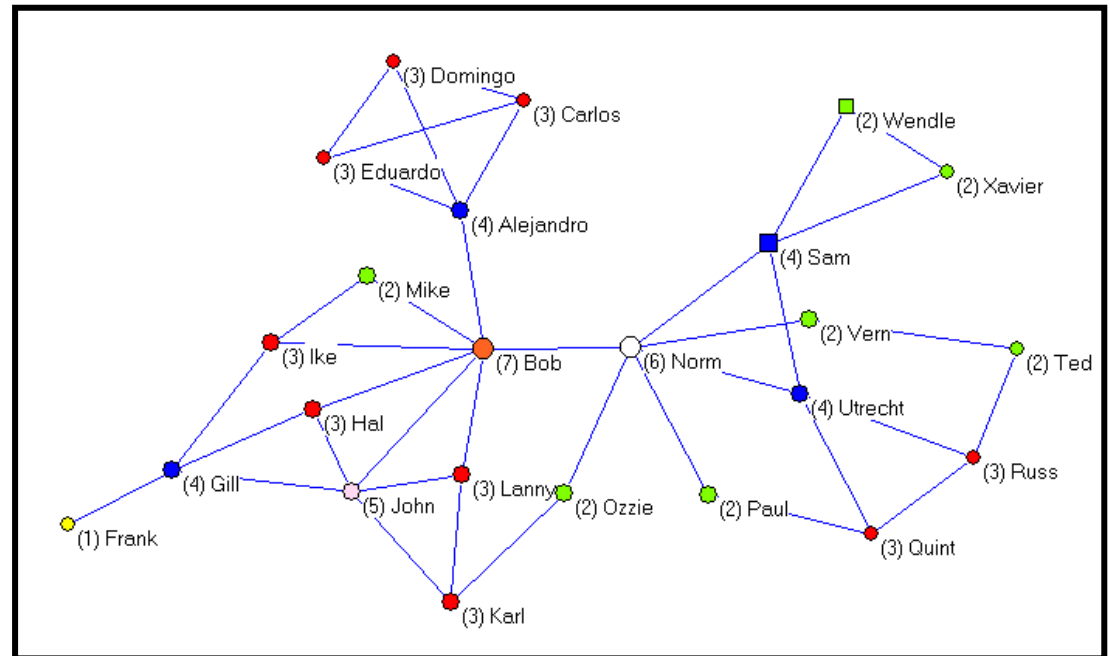
Degree /betweenness centrality



Betweenness centralization
0.55

Degree /closeness centrality

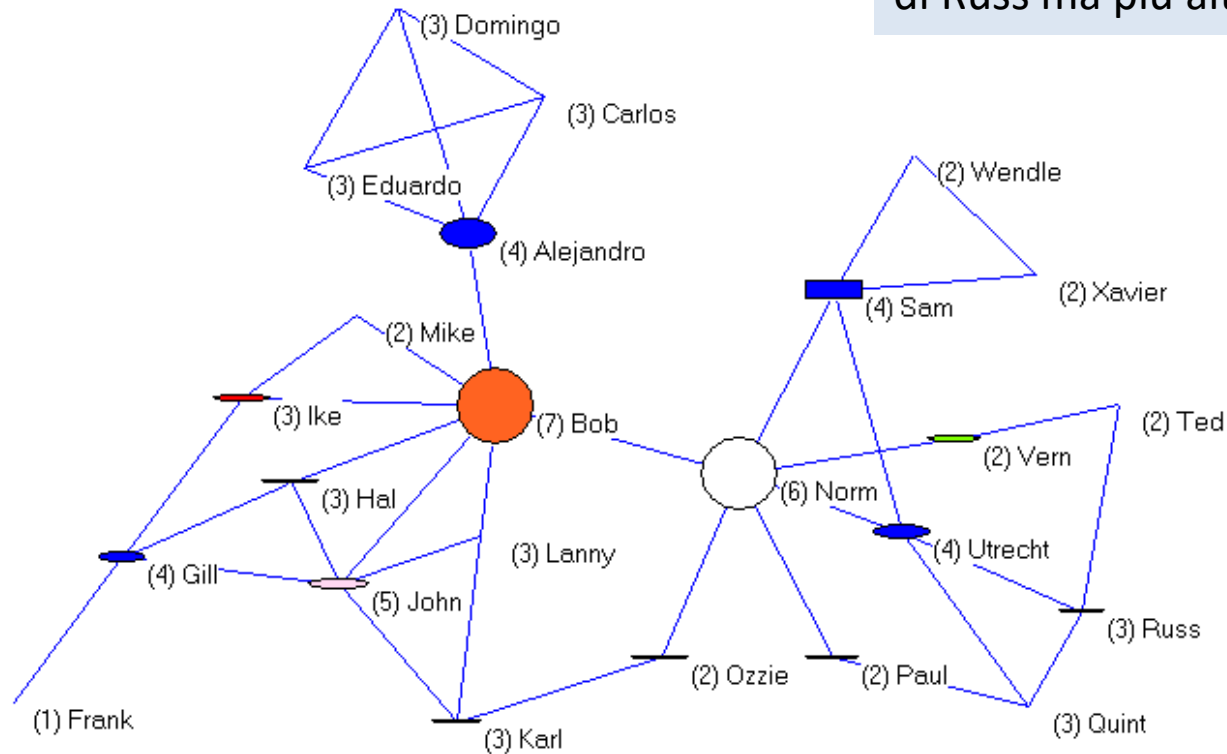
closeness centralization
0.35



Degree partition/Closeness/Betweenness

Alejandro e Utrecht hanno grado uguale ma Alejandro ha maggiore Betweenness.
Ozzie ha la stessa Betweenness di Russ ma più alta Closeness

Second vector (Betweenness)

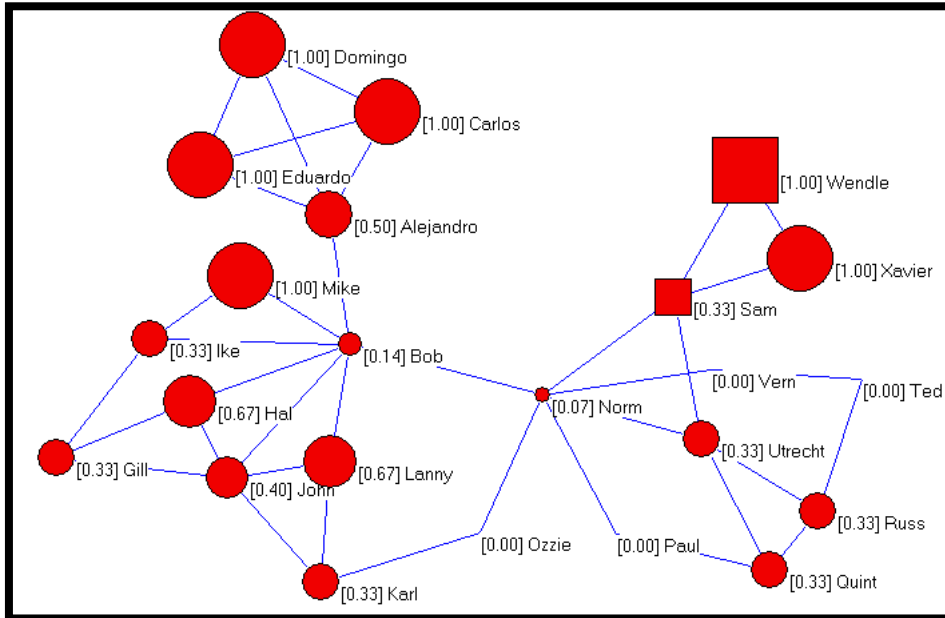


First Vector (Closeness)

Clustering

**Watts-Strogatz Clustering Coefficient:
0.46128364**

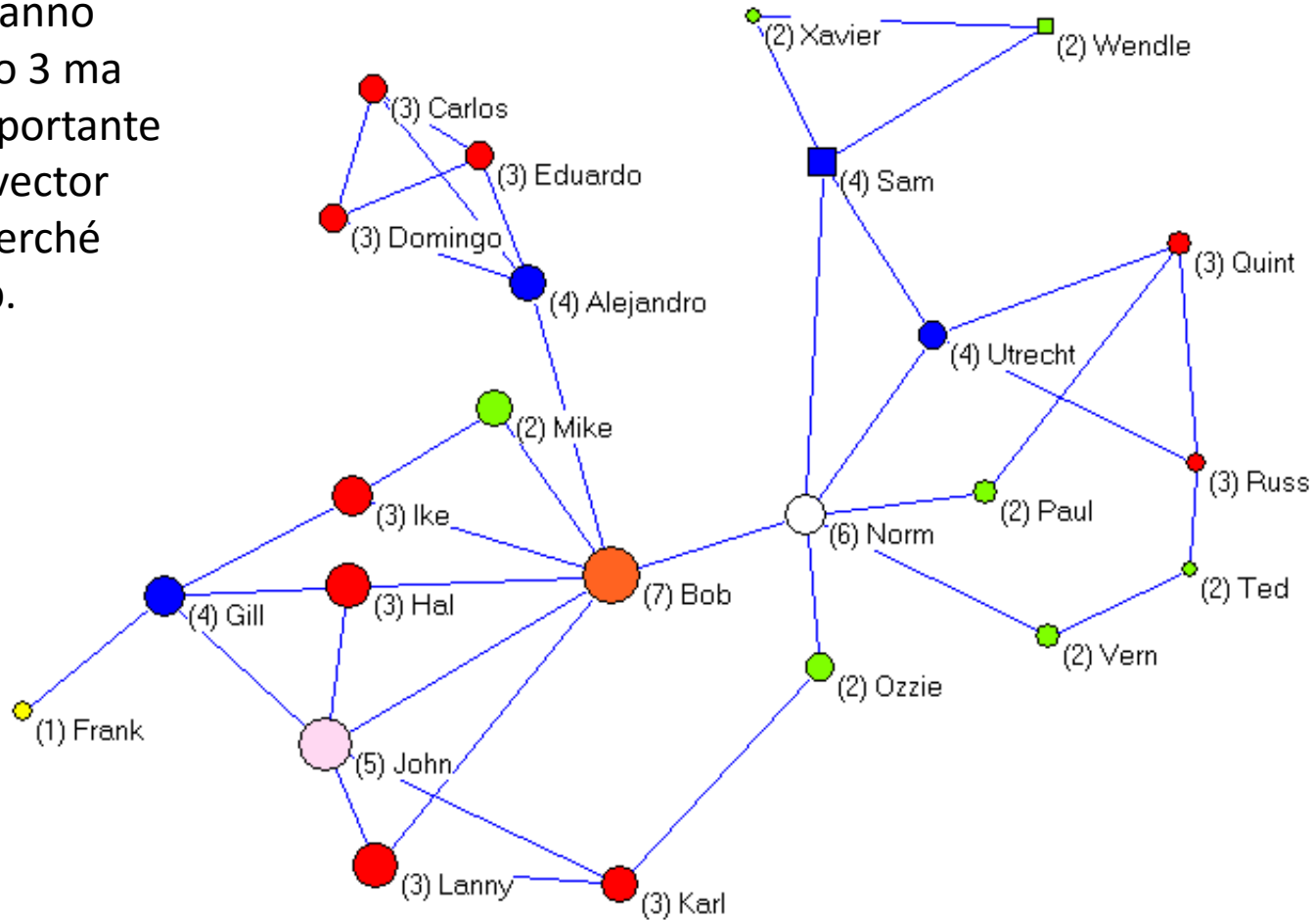
tolgo Frank altrimenti non riesco a
rappresentare graficamente il vettore di
clustering



1. 999999998.000000 - Frank
2. 0.166667 - Gill
3. 0.333333 - Ike
4. 1.000000 - Mike
5. 0.666667 - Hal
6. 0.400000 - John
7. 0.333333 - Karl
8. 0.666667 - Lanny
9. 0.142857 - Bob
10. 0.500000 - Alejandro
11. 1.000000 - Carlos
12. 1.000000 - Eduardo
13. 1.000000 - Domingo
14. 0.066667 - Norm
15. 0.000000 - Ozzie
16. 0.000000 - Vern
17. 0.000000 - Paul
18. 0.333333 - Quint
19. 0.333333 - Utrecht
20. 0.333333 - Russ
21. 0.000000 - Ted
22. 0.333333 - Sam
23. 1.000000 - Xavier
24. 1.000000 - Wendle

Degree partition/eigenvector centrality vector

Russ e Hal hanno uguale grado 3 ma Hal è più importante come eigenvector centrality perché legato a Bob.

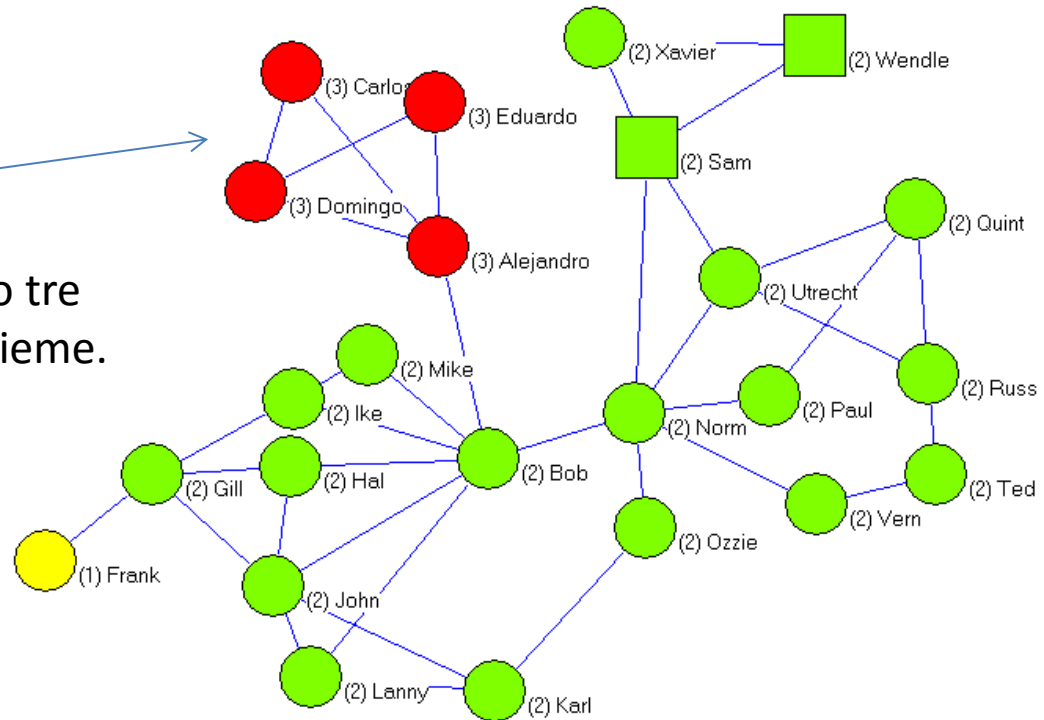


Partizioni: k-core

K-core: esistono sottoinsiemi coesi?

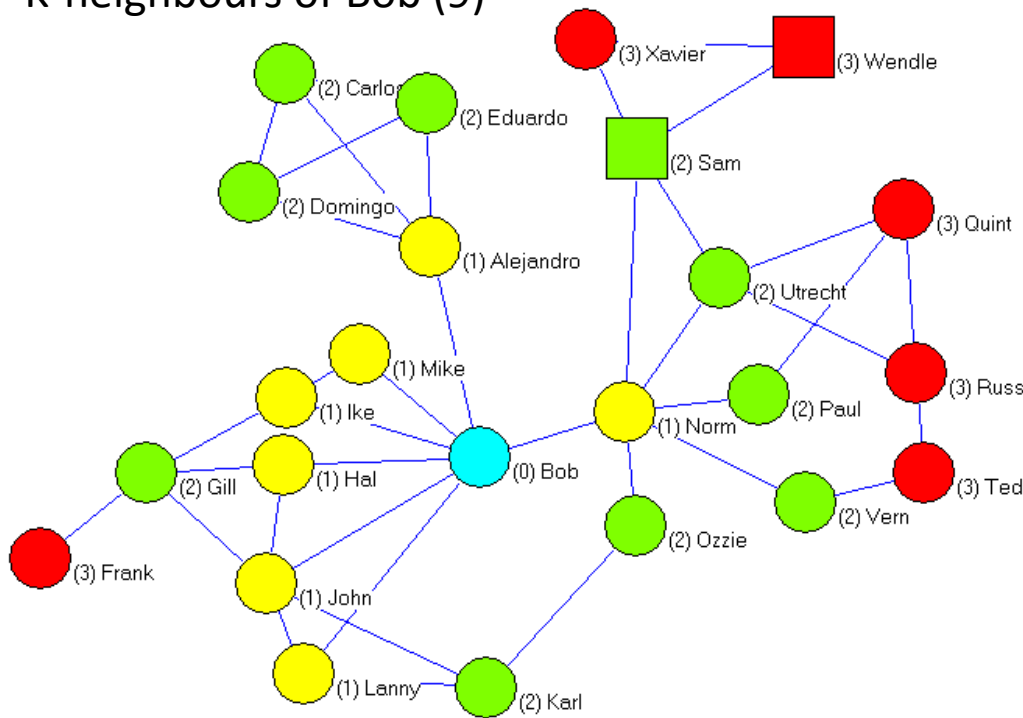
3-core

Sottoinsieme di nodi che hanno tre «amici» all'interno del sottoinsieme.



$4\text{-core} \subset 3\text{-core} \subset 2\text{-core} \dots$

K-neighbours of Bob (9)

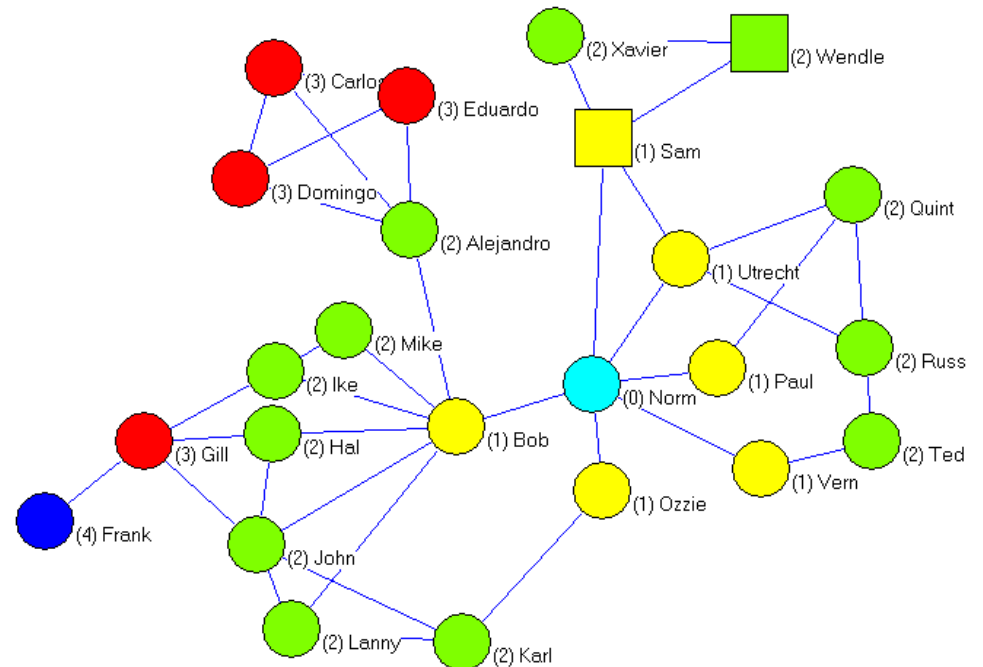


A partire da Norm una informazione può raggiungere tutti in 4 passi

Partizioni: k-neighbours

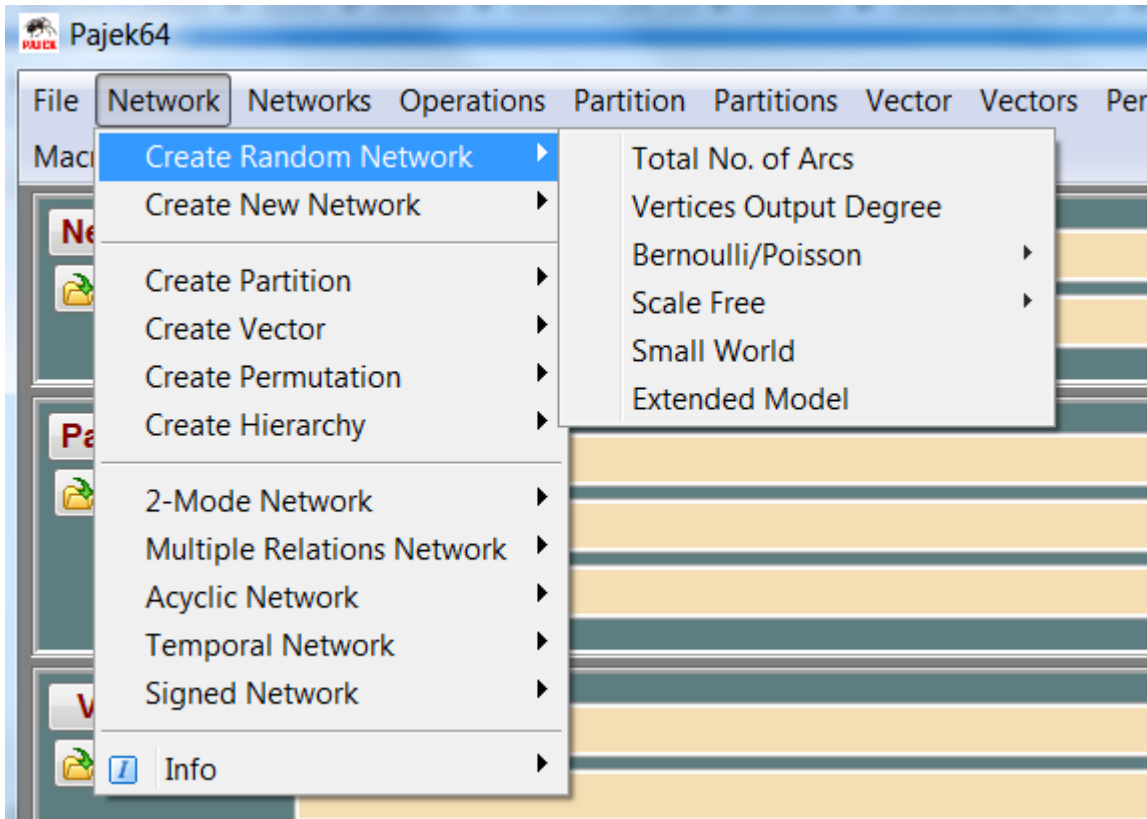
A partire da Bob una informazione può raggiungere tutti in 3 passi

K-neighbours of Norm (14)



Costruire una rete Random con Pajek

Reti Erdos Renyi
Reti Watts-Strogats
Reti Scale Free



Bernoulli/Poisson (Erdos-Renyi): genera una rete in accordo con il modello di Erdos-Renyi. Il comando Pajek chiede il numero di vertici e il grado medio.

Scale Free: genera una rete scale free in accordo con il modello di Pennock et al. 2002 (Winner don't take all), <http://modelingtheweb.com>.

Il comando Pajek chiede il numero di vertici massimo, numero massimo di linee, il grado medio, dimensione della E-R iniziale, Probabilità iniziale delle linee, α =peso da dare alla dinamica del preferential attachment. Più è grande α più il preferential attachment è importante.

Small World: genera una rete Small World e chiede: numero di nodi della rete, numero di vicini collegati ad ogni nodo (nell'anello di partenza). Probabilità di ricollegamento (rewiring). Più è alta più la rete risultante sarà vicina ad una rete random , più è bassa più la rete risultante sarà vicina ad una ad anello

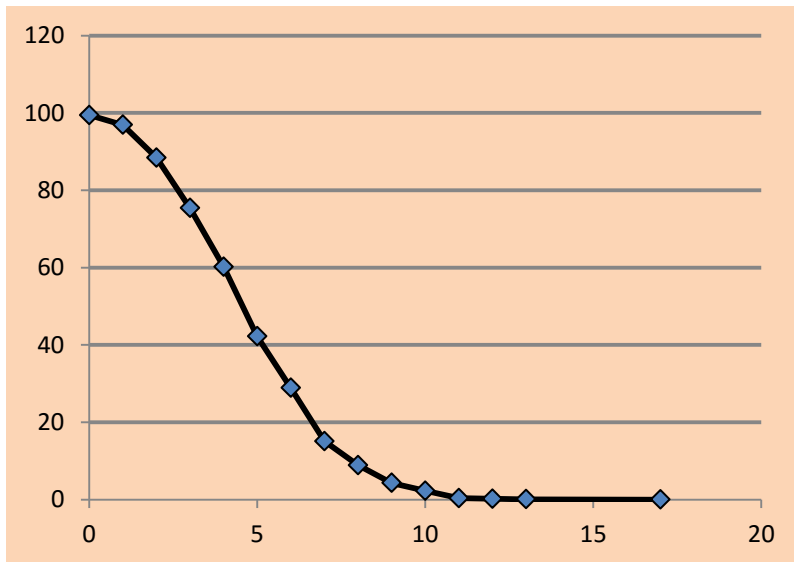
Bernoulli/Poisson Random Network



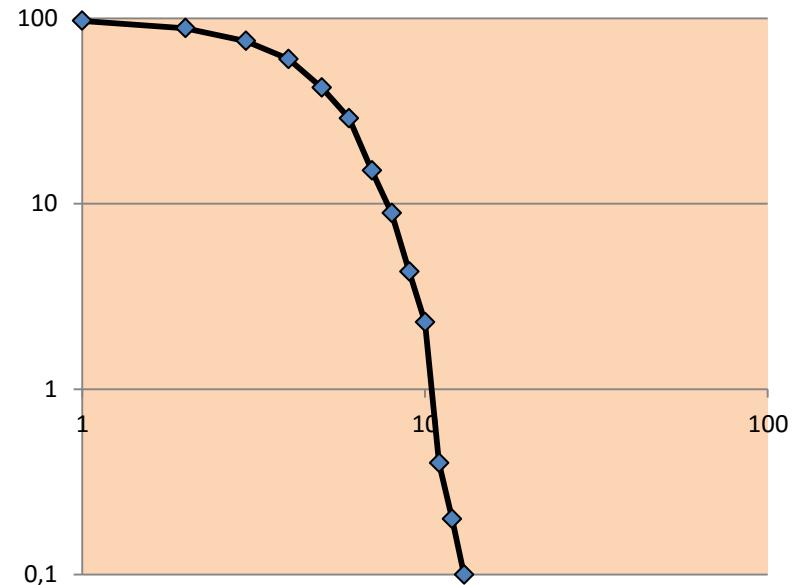
Random Network

Number of vertices=1000
Average Degree of vertices=5

Funzione di sopravvivenza:
 $P(x>k)$



k



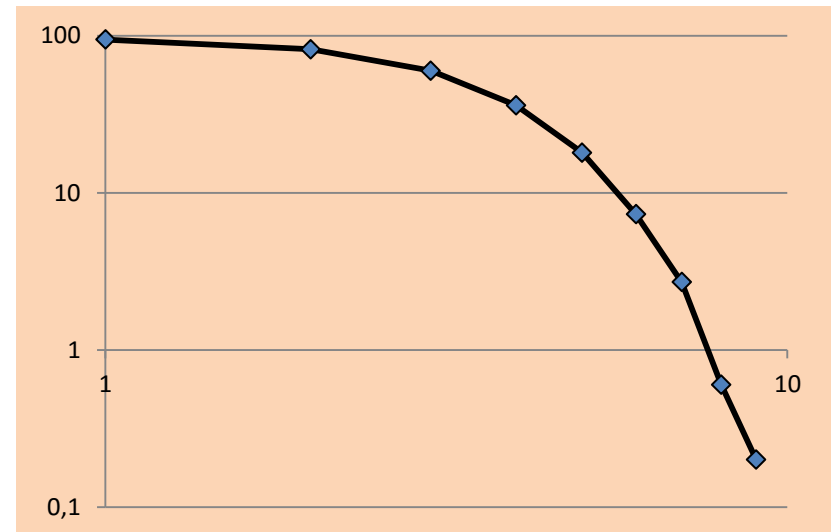
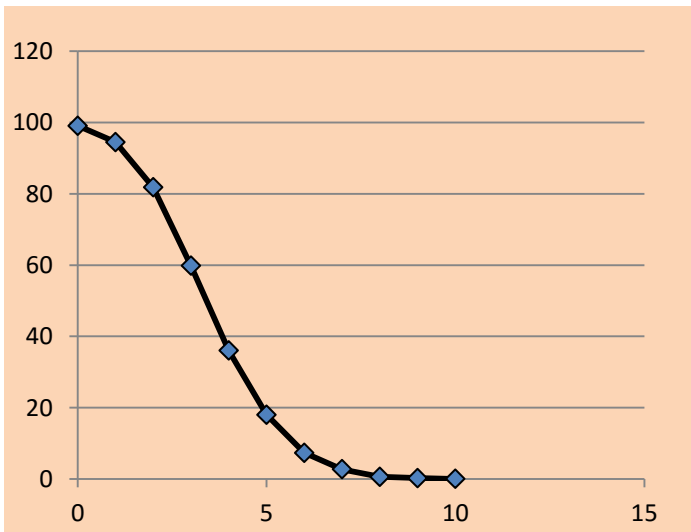
k

Small World Random Network

Number of vertices=1000

Number of linked neighbors on each side of a vertex=2

Replacement Probability=0.5



Funzione di sopravvivenza:
 $P(x > k)$

Scale Free Random Network

Number of vertices=1000

Number lines=0

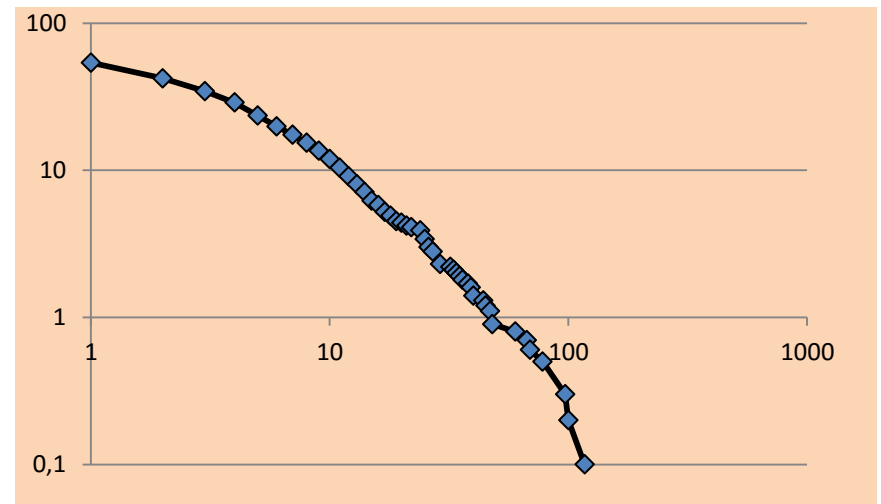
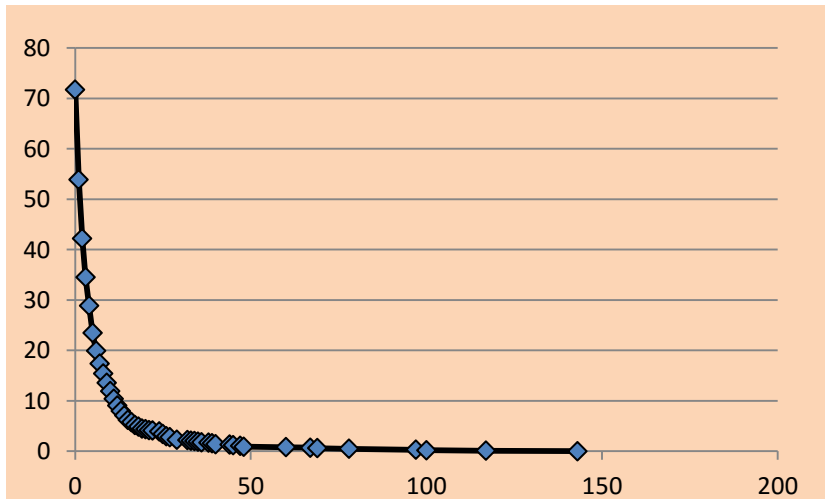
Average Degree of vertices=5

Number of vertices of initial Erdos-Renyi network=10

Initial Probability of lines=0.2

Alpha=0.25

Replacement Probability=0.5



Funzione di sopravvivenza:
 $P(x > k)$

	l (distanza media)	d (diametro)	C (Clustering)
ER	4.3	8	0.004
SW	5.4	12	0.07
SF	3.4	8	0.08