Pajek

Bhaben Deori 03 CS 3020

April 22, 2007

INTRODUCTION: Pajek is the most traditional of the visualization and analysis tools used for large networks having some thousand vertices.Other network tools like Visant,Graph Vis are visualizaion tools only.

We may however represent/visualize a large graph also by studying its degree distribution. But Pajek is a very powerful visualization tool and very efficient.

Pajek:Goals– The main goal of pajek is to provide its users with a powerful network visualization tool.But apart from this it also

1.supports abstraction by (recursive) decomposition of a large network into several smaller networks.

2.implements a selection of efficient (sub quadratic) algorithms for analysis of large networks.Infact Pajek allowes the user to plug his/her own algorithms.

Data Structures: In Pajek six types of objects are used:

1. Networks main objects (vertices and lines). Default extension: .net. Network can be presented on input file in different ways:(a)using arcs/edges (e.g. 1 2 line from 1 to 2) .(b) using arcslists/edgeslists (e.g. 1 2 3 line from 1 to 2 and from 1 to 3) (c) matrix format.Additional information for network drawing can be included in input file as well.

2. Partitions, they tell for each vertex to which class the vertex belong. Default extension: .clu

3. Permutations, reordering of vertices.This can be achieved because Pajek has in memory data structures to implement the reordering of vertices. Default extension: .per

4. Clusters, subset of vertices (e.g. one class from partition). Default extension: .cls.

(111111)- cluster 1, (222222)-cluster 2.

5. Hierarchies, hierarchically ordered vertices. Default extension: .hie.

6. Vectors, they tell for each vertex some numerical property (real number) like degree,clustering coefficient. Default extension: .vec.

In Pajek, sparse representation of networks is obtained by implementing adjacency list while dense is represented by adjacency matrix.

With Pajek we can : find clusters (components, neighborhoods of 'important' vertices, cores, etc.) in a network, extract vertices that belong to the same clusters and show them separately, possibly with the parts of the context (detailed local view), shrink vertices in clusters and show relations among clusters (global view)

Specifying a network in Pajek:

In this part we see how to specify a network.

1.Creating the network topology.

We start off by typing "*Vertices " followed by the number of vertices we wish to have in a textfile(like notepad). Say for example 10 vertices. -> *Vertices 10.Then number 1 is entered and its labelled with a name in parenthesis, followed by the X and Y coordinates within a range from 0 to 1.eg. 1 "node1" 0.1 0.2.This specifies node 1's x and y coordinate.We may also wish to add a Z coordinate if you wish and also a shape for the node. Some shaped available are: ellipse, diamond, box, triangle, box. This way we continue for as many vertices as defined at the beginning of the file. After we finish with vertices initialization, we may begin the next line with "*Matrix".

Now we may specify node relationships seperately with either fancy arcs or basic vertices, but for simplicity's sake, we will use an adjancey matrix to define node relationships. Using this method will create simple directed vertices between associated nodes. To create a connection between nodes, we place a 1 in the matrix, else put a 0 for no relationship. In the end, it should look something like this:

***Vertices 4**

1 "Andrej" 0.1201 0.2849 0.5000 ellipse

2 "Vlado" 0.8188 0.2458 0.5000 box

3 "Pajek" 0.3688 0.7792 0.5000 diamond

4 "Book" 0.8359 0.8333 0.5000 triangle

***Edges**

1 2

1 3

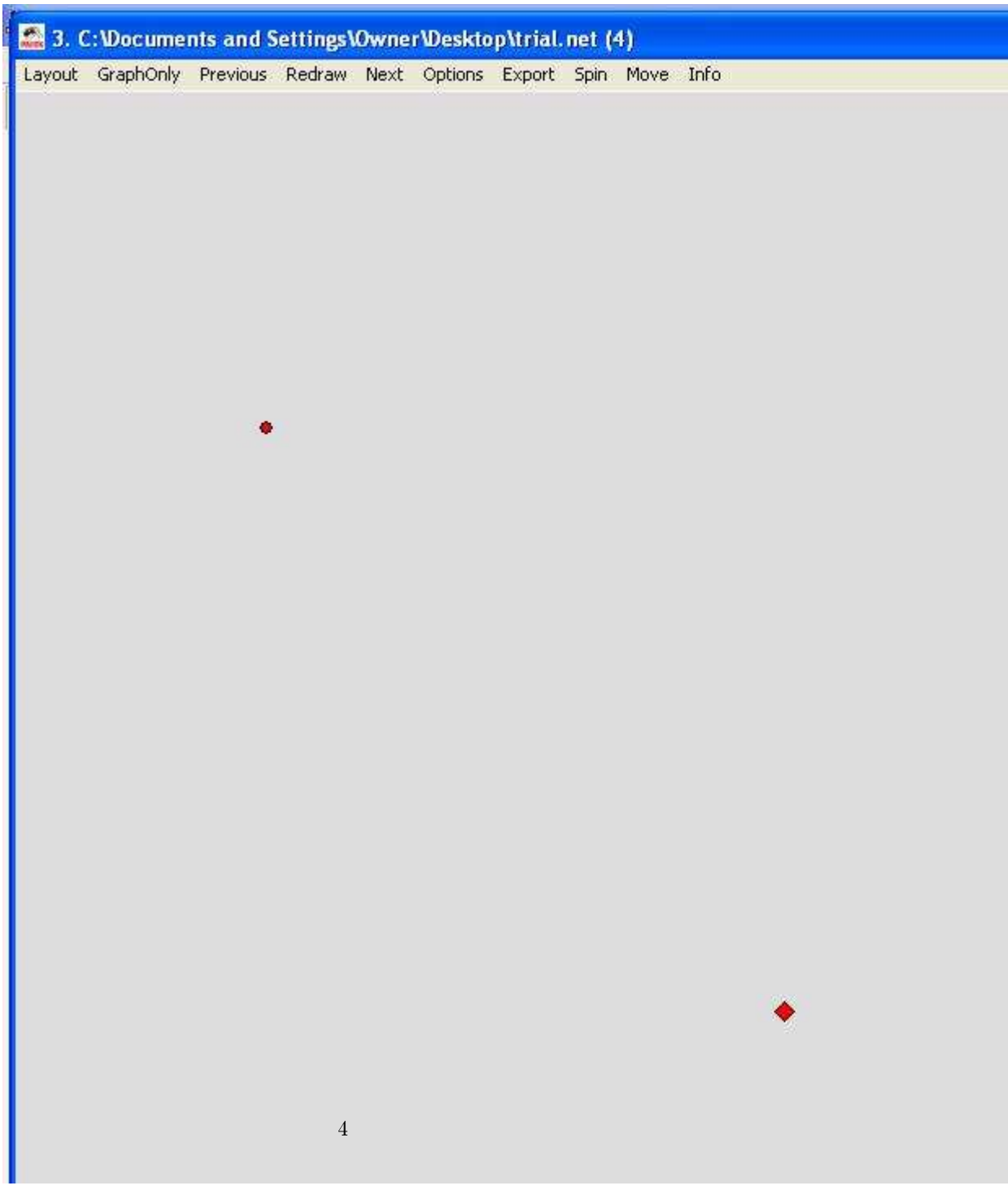
1 4

2 3

2 4

3 4

The graph appears as:


3. C:\Documents and Settings\Owner\Desktop\trial.net (4)
Layout GraphOnly Previous Redraw Next Options Export Spin Move Info

Main Window Tools:

Network:N.net

Read network from Ascii file.

Edit network. Choose vertex, show its neighbors and then:

add new lines to/from selected vertex (by left mouse double clicking on Newline);

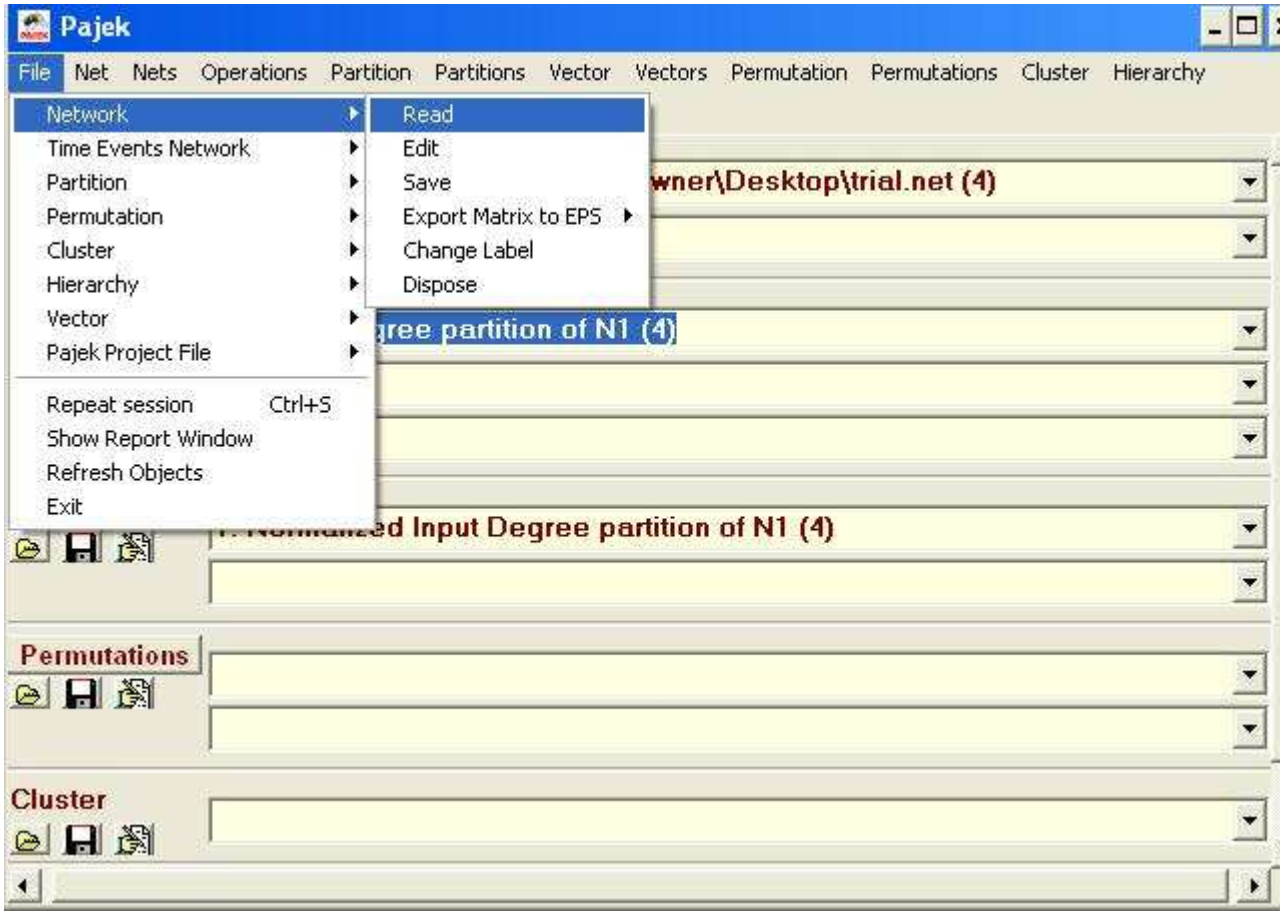
delete lines (by left mouse double clicking);

change value of line (by single right mouse clicking);

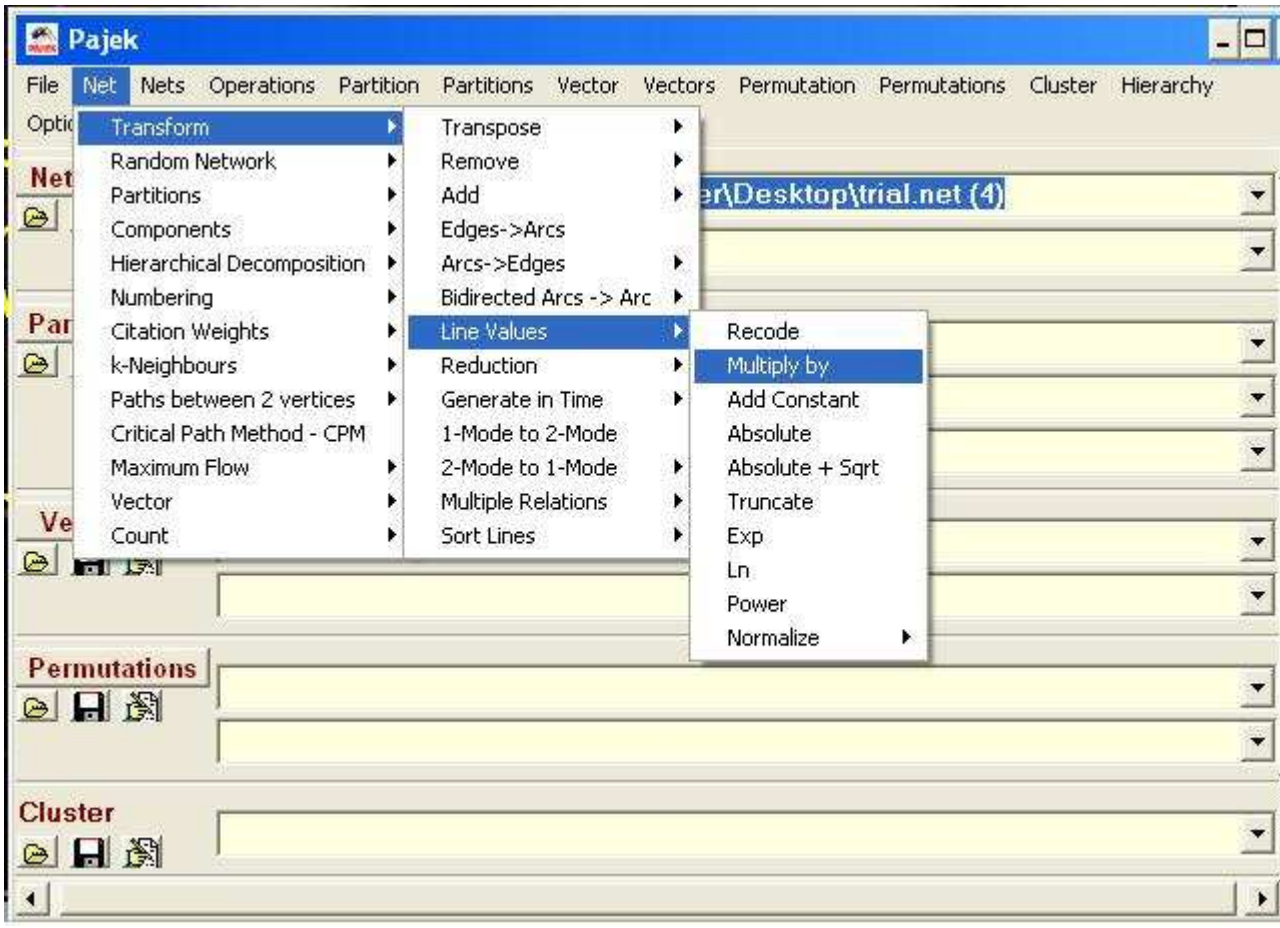
Save selected network to Ascii file.

Export Matrix to EPS > write matrix in EPS format

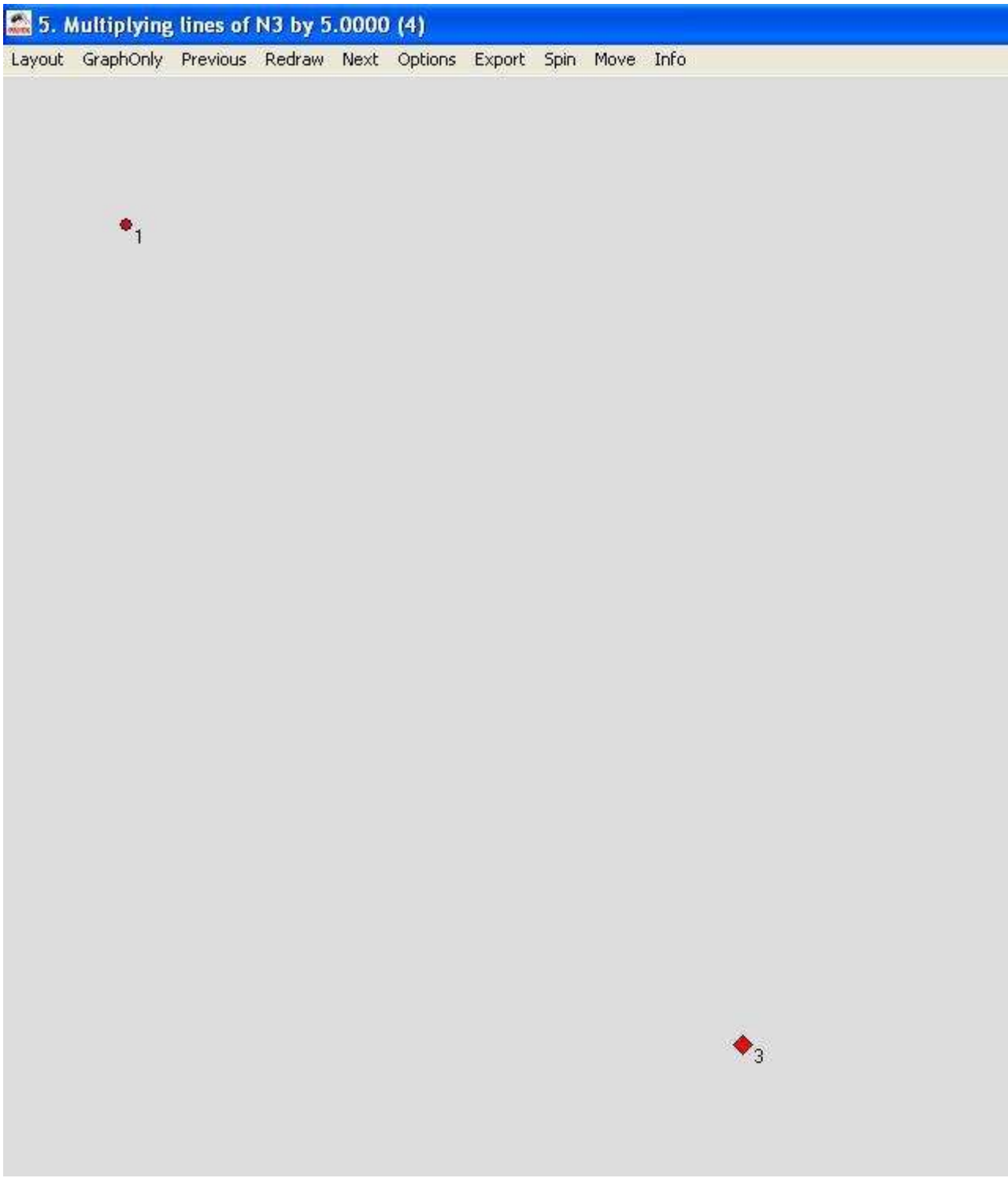
Dispose selected network from memory.The GUI for reading a network would appear as:



Line values can be transformed by multiplying by a constant, adding constant to line values,absolute line values,absolute + Sqrt etc.For eg. the GUI of Pajek for the transformation of line value by multiplying by a constant would appear as:



The new network graph after multiplying by 5 would be:


5. Multiplying lines of N3 by 5.0000 (4)
Layout GraphOnly Previous Redraw Next Options Export Spin Move Info
1
3

These are only a few of some of the functions we can use PAJEK for operating on networks.The scope of Pajek is very large.

References:

[1] Pajek Program for Analysis and Visualization of Large Networks Reference Manual

List of commands with short explanation version 1.13 Vladimir Batagelj and Andrej

Mrvar Ljubljana, March27, 2006.

[2] V. Batagelj, A. Mrvar: Pajek Program for Large Network Analysis. Home page:

http://vlado.fmf.uni-lj.si/pub/networks/pajek/.

[3] V. Batagelj, A. Mrvar: Pajek Analysis and Visualization of Large Networks. In

Junger, M., Mutzel, P. (Eds.): Graph Drawing Software. Springer (series Mathematics

and Visualization), Berlin 2003. 77-103. ISBN 3-540- 00881-0. Springer, Amazon,

preprint.