

# network *visualization*

introduction to *network science in Python* (*NetPy*)

Lovro Šubelj  
University of Ljubljana  
4th July 2024

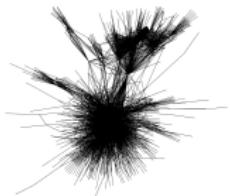
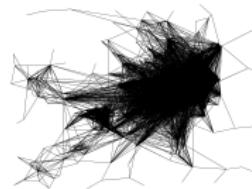
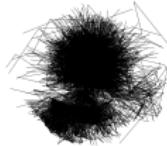
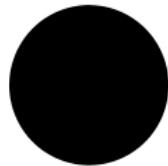
# visualization *overview*

network *visualization* with *wiring diagram*\*

1st compute *network layout* as *coordinates* in Euclidean plane etc.

2nd *representation* of *network links*? strength, pattern, shape, color etc.

3rd *representation* of *network nodes*? size, shape, color, label etc.



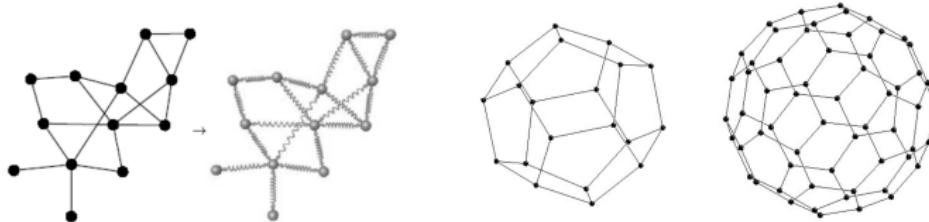
similar link lengths, no crossings, displays symmetry, even node distribution etc.

---

\* small/dense graphs better visualized with block models

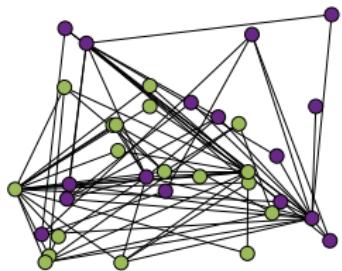
## visualization *framework*

- Eades *spring embedded layout* [Ead84]
- Fruchterman-Reingold *force-directed layout* [FR91]
- Kamada-Kawai *graph theoretic layout* [KK89]
- *attractive forces* between *neighbors* using Hooke's law
- *repulsive force* between *nodes* using Coulomb's law
- iteratively move nodes thus to *minimize layout energy*

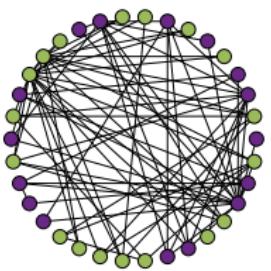


pleasing with similar link lengths, symmetry & even distribution

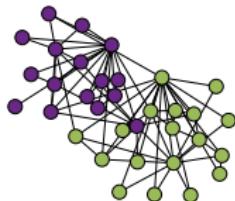
# visualization *karate*



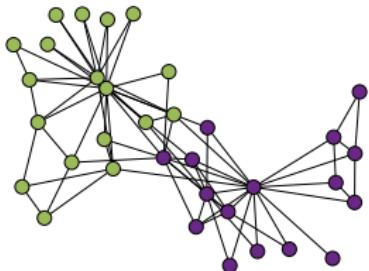
*random layout*



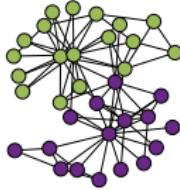
*circular layout*



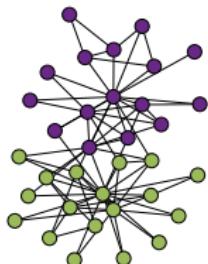
*spring embedded layout* [Ead84]



*Fruchterman-Reingold layout* [FR91]

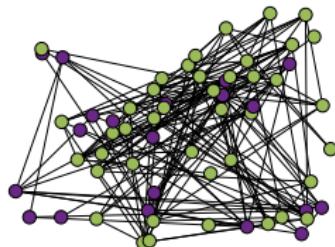


*Kamada-Kawai layout* [KK89]

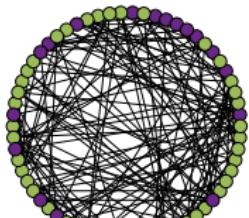


*LGL layout* [ADWM04]

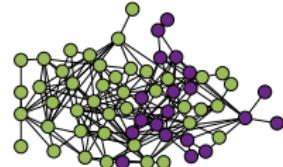
# visualization *dolphins*



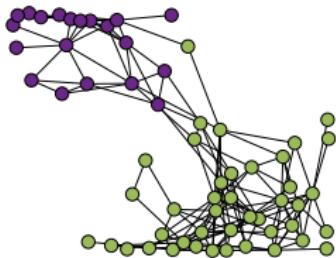
*random layout*



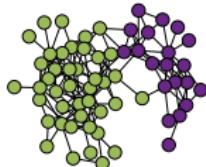
*circular layout*



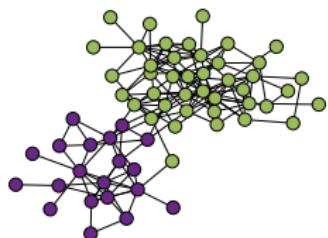
*spring embedded layout* [Ead84]



*Fruchterman-Reingold layout* [FR91]

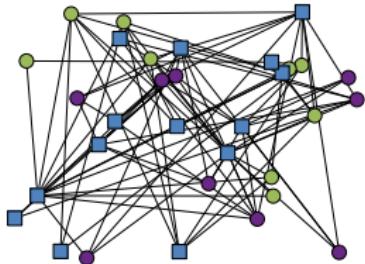


*Kamada-Kawai layout* [KK89]

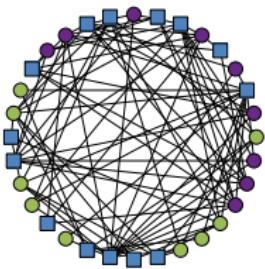


*LGL layout* [ADWM04]

# visualization *women*



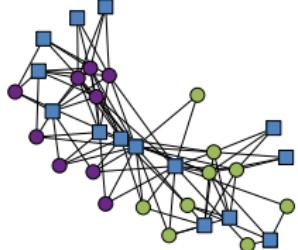
*random layout*



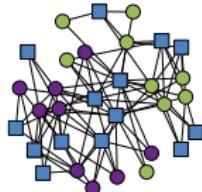
*circular layout*



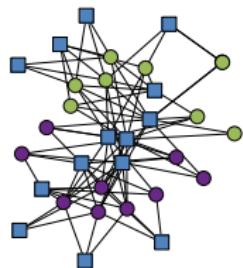
*spring embedded layout [Ead84]*



*Fruchterman-Reingold layout [FR91]*

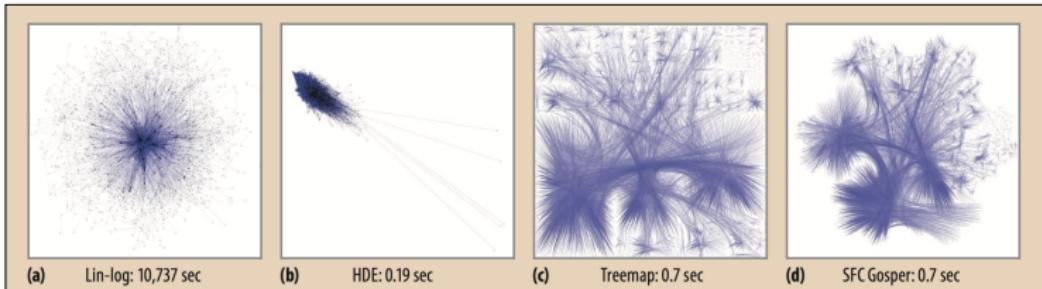


*Kamada-Kawai layout [KK89]*

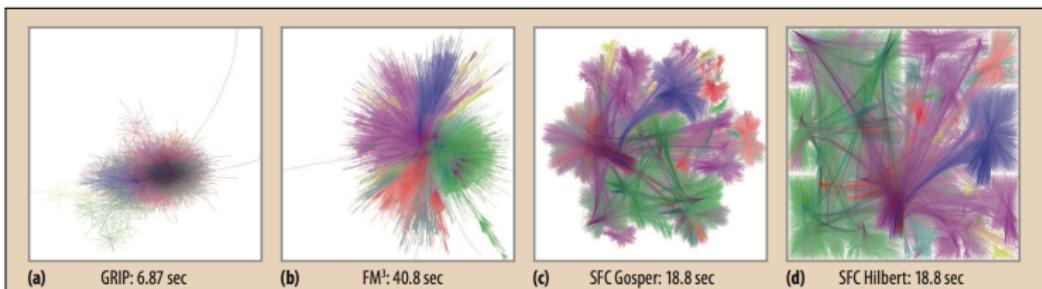


*LGL layout [ADWM04]*

# visualization *static*

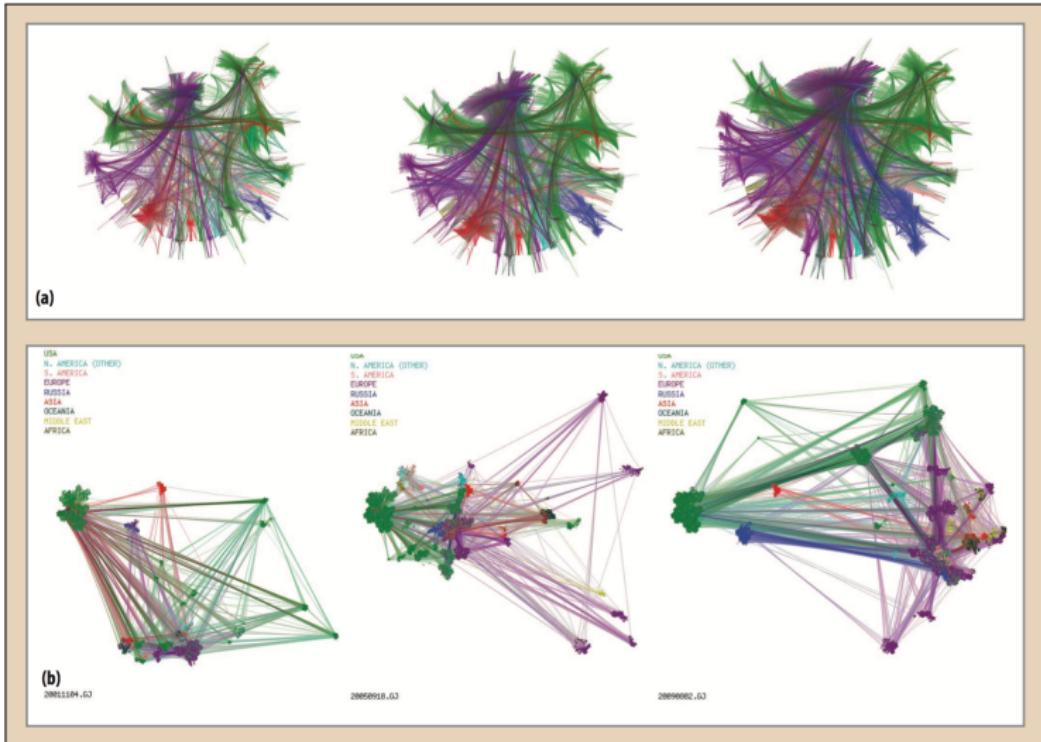


(a) *traditional* (b) *algebraic* (c) *hierarchical* and (d) *clustering-based force-directed layouts* of web graph



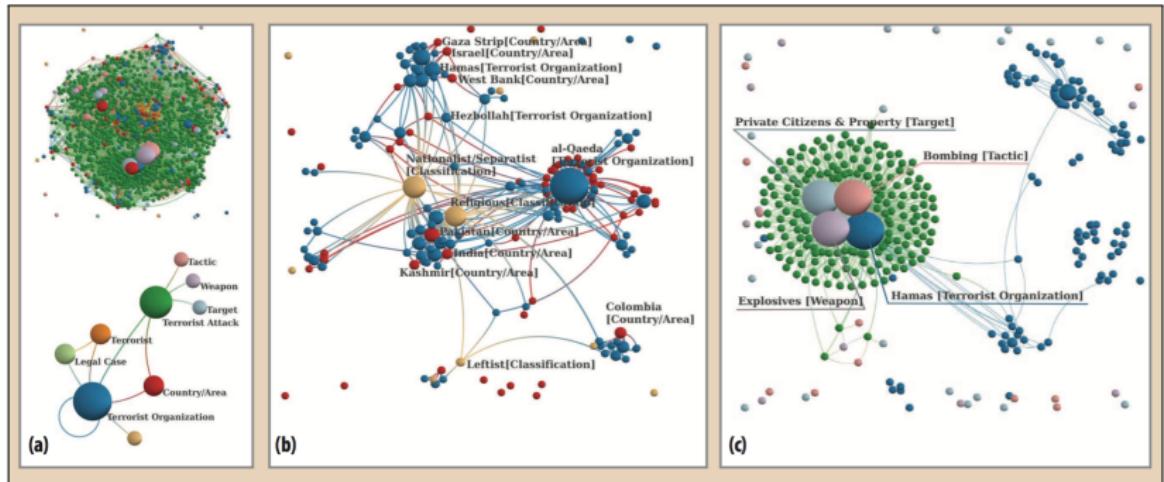
(a,b) *multilevel* and (c,d) *clustering-based force-directed layouts* of autonomous systems by continent

# visualization *dynamic*



(a) *incremental* and (b) *global* clustering-based *force-directed layouts* of Internet by continent

# visualization *heterogeneous*



terrorist network (a) *force-directed layout* with *semantic ontology* and (b) *active organizations* (c) *attack behaviour*

# visualization *references*

-  Alex T. Adai, Shailesh V. Date, Shannon Wieland, and Edward M. Marcotte.  
LGL: Creating a map of protein function with an algorithm for visualizing very large biological networks.  
*J. Mol. Biol.*, 340(1):179–190, 2004.
-  Peter Eades.  
A heuristic for graph drawing.  
*Congressus Numerantium*, 42:149–160, 1984.
-  Thomas M. J. Fruchterman and Edward M. Reingold.  
Graph drawing by force-directed placement.  
*Softw: Pract. Exper.*, 21(11):1129–1164, 1991.
-  Helen Gibson, Joe Faith, and Paul Vickers.  
A survey of two-dimensional graph layout techniques for information visualisation.  
*Infor. Visual.*, 12(3-4):324–357, 2013.
-  Tomihisa Kamada and Satoru Kawai.  
An algorithm for drawing general undirected graphs.  
*Inform. Process. Lett.*, 31(1):7–15, 1989.
-  Stephen G. Kobourov.  
Force-directed drawing algorithms.  
In Roberto Tamassia, editor, *Handbook of Graph Drawing and Visualization*, pages 383–408. CRC Press, 2013.
-  Kwan-Liu Ma and Chris W. Muelder.  
Large-scale graph visualization and analytics.  
*Computer*, 46(7):39–46, 2013.