

Netwulf: Interactive visualization of networks in Python

Ulf Aslak^{1, 2} and Benjamin F. Maier^{3, 4}

1 Center for Social Data Science, University of Copenhagen, DK-1353 København K **2** DTU Compute, Technical University of Denmark, DK-2800 Kgs. Lyngby **3** Robert Koch Institute, Nordufer 20, D-13353 Berlin **4** Department of Physics, Humboldt-University of Berlin, Newtonstr. 15, D-12489 Berlin

DOI: [10.21105/joss.01425](https://doi.org/10.21105/joss.01425)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Submitted: 25 April 2019

Published: 01 October 2019

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Summary

Network visualization is an effective way to illustrate properties of a complex system. It is an important tool for exploring and finding patterns, and is used by researchers and practitioners across many fields and industries. Currently, there exist a number of tools for visualizing networks. *Networkx* (Hagberg, Schult, & Swart, 2008) is a popular Python package for network analysis which provides limited functionality for computing layouts and plotting networks statically. Layout computations are done in Python or using the php-based software *Graphviz* (Ellson, Gansner, Koutsofios, North, & Woodhull, 2002), which is slow. Another Python package for network analysis and visualization is *graph-tool* (Peixoto, 2014), which relies on a high number of external C++-libraries for installation/compilation which can be overwhelming for beginners. Furthermore, its visualization functions have limited interactive features. *Gephi* (Bastian, Heymann, & Jacomy, 2009) and *Cytoscape* (Shannon et al., 2003) are dedicated interactive visualization and analysis software programs. They are both Java-based and run desktop clients with a GUI where users save and load networks as separate files. *Webweb* (Wapman & Larremore, 2019) enables interactive visualization for Python and Matlab networks using the d3.js (Bostock, Ogievetsky, & Heer, 2011) force layout. Its main purpose, however, is exploration of network features and exporting one-time visualizations as SVG or HTML.

For many users, these tools offer the necessary functionality to visualize networks in most desired ways. However, since a growing population of network researchers and practitioners are relying on Python for doing network science (*Stack overflow developer survey*, 2019), it is increasingly pressing that a fast and intuitive Python tool for reproducible network visualization exists.

Netwulf is a light-weight Python library that provides a simple API for interactively visualizing a network and returning the computed layout and style. It is build around the philosophy that network manipulation and preprocessing should be done programmatically, but that the efficient generation of a visually appealing network is best done interactively, without code. Therefore, it offers no analysis functionality and only few exploration features, but instead focuses almost entirely on fast and intuitive layout manipulation and node/link styling. Interaction with *Netwulf* typically works as follows:

1. Users have a network object, *G*, in either dictionary or *networkx.Graph* format. They then launch a *Netwulf* visualization by calling `netwulf.visualize(G)`.
2. The command opens a new browser window containing *G* as an interactive, manipulable, stylable network. Here, the user can, for instance, explore how different configurations of physics parameters like *node charge* and *gravity* influence the layout, they can change properties like node color and link opacity, and even threshold the network data for weak or strong links. When the user has finalized the layouting process, they may either:

1. Save the image directly from the interactive visualization as a PNG file.
2. Post the style and computed node positions back to Python in a dictionary format, which allows for further manipulation in the Python backend. Moreover, using the function `netwulf.draw_netwulf`, the network can be redrawn using the common Python drawing library *matplotlib* (Hunter, 2007), which further enables saving the visualization in any format.

The interactive visualization is implemented in JavaScript, relies on d3.js (Bostock et al., 2011) for computing layouts, and uses the HTML5-object canvas for rendering. This makes it, to our knowledge, the most performant tool for interactive network visualization in Python to date.

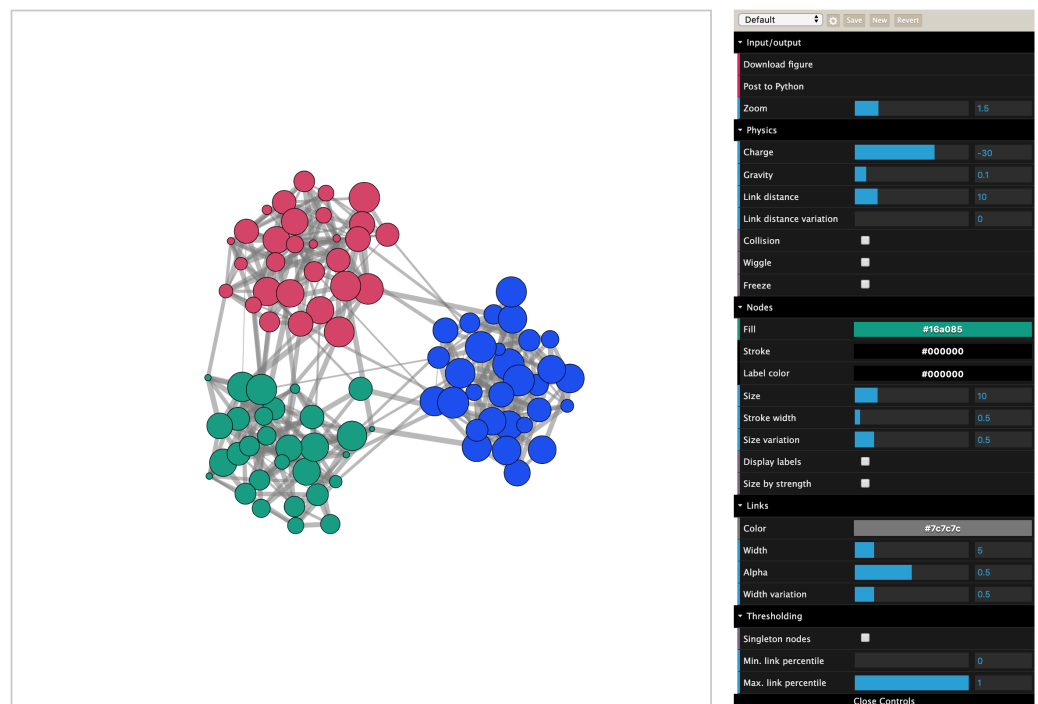


Figure 1: Interactive visualization of a modular network in Netwulf.

Acknowledgements

Both authors contributed equally to the software, documentation, and manuscript. B. F. M. is financially supported as an *Add-On Fellow for Interdisciplinary Life Science* by the Joachim Herz Stiftung.

References

- Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: An Open Source Software for Exploring and Manipulating Networks. In. doi:[10.13140/2.1.1341.1520](https://doi.org/10.13140/2.1.1341.1520)
- Bostock, M., Ogievetsky, V., & Heer, J. (2011). D3: Data - Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12). doi:[10.1109/TVCG.2011.185](https://doi.org/10.1109/TVCG.2011.185)

Ellson, J., Gansner, E., Koutsofios, L., North, S. C., & Woodhull, G. (2002). Graphviz— open source graph drawing tools. In P. Mutzel, M. Jünger, & S. Leipert (Eds.), *Graph drawing* (pp. 483–484). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:[10.1007/3-540-45848-4_57](https://doi.org/10.1007/3-540-45848-4_57)

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring network structure, dynamics, and function using NetworkX. In Gäel Varoquaux Travis Vaught & J. Millman (Eds.), *Proceedings of the 7th Python in Science Conference (SciPy 2008)* (Vol. 7, pp. 11–15). Pasadena, CA USA.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3), 90–95. doi:[10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55)

Peixoto, T. P. (2014). The graph-tool python library. *figshare*. doi:[10.6084/m9.figshare.1164194](https://doi.org/10.6084/m9.figshare.1164194)

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., et al. (2003). Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome research*, 13(11), 2498–2504. doi:[10.1101/gr.1239303](https://doi.org/10.1101/gr.1239303)

Stack overflow developer survey. (2019). Retrieved from <https://insights.stackoverflow.com/survey/2019>

Wapman, H. K., & Larremore, D. B. (2019). Webweb: A tool for creating, displaying, and sharing interactive network visualizations on the web. *Journal of Open Source Software*. doi:[10.21105/joss.01458](https://doi.org/10.21105/joss.01458)