

Systems biology

FANMOD: a tool for fast network motif detection

Sebastian Wernicke* and Florian Rasche

Institut für Informatik, Friedrich-Schiller-Universität Jena, Ernst-Abbe-Platz 2, 07743 Jena, Germany

Received on January 3, 2006; revised and accepted on February 1, 2006

Advance Access publication February 2, 2006

Associate Editor: Jonathan Wren

ABSTRACT

Summary: Motifs are small connected subnetworks that a network displays in significantly higher frequencies than would be expected for a random network. They have recently gathered much attention as a concept to uncover structural design principles of complex biological networks. FANMOD is a tool for fast network motif detection; it relies on recently developed algorithms to improve the efficiency of network motif detection by some orders of magnitude over existing tools. This facilitates the detection of larger motifs in bigger networks than previously possible. Additional benefits of FANMOD are the ability to analyze colored networks, a graphical user interface and the ability to export results to a variety of machine- and human-readable file formats including comma-separated values and HTML.

Availability: The tool is freely available online at <http://www.minet.uni-jena.de/~wernicke/motifs/> and runs under Linux, MacOS and Windows.

Contact: wernicke@minet.uni-jena.de

1 INTRODUCTION

Many biological networks contain certain small subnetworks in significantly higher frequencies than random networks. Milo *et al.* (2002, 2004) proposed to use such overabundant ‘topological modules’ (Vespignani, 2003) for uncovering the structural design principles of biological networks, thereby coining the term network motifs for them. The analysis of network motifs has led to interesting results, e.g. in the areas of protein–protein interaction prediction (Albert and Albert, 2004), hierarchical network decomposition (Itzkovitz *et al.*, 2005) and the analysis of temporal gene expression patterns (Kalir *et al.*, 2001).

Finding network motifs consists of three computationally expensive subtasks:

- Finding which subgraphs occur in the input network and in what number.
- Determining which of these subgraphs are topologically equivalent (i.e. isomorphic) and grouping them into subgraph classes accordingly.
- Determining which subgraph classes are displayed at much higher frequencies than in random graphs (under a specified random graph model).

Some work has been spent on the second subtask but—until recently—considerably less on the other two. In order to speed up the first subtask, an algorithm for sampling subgraphs has

been proposed by Kashtan *et al.* (2004). However, among some other drawbacks, this algorithm provides only non-uniform sampling and scales poorly as motif size increases; a more detailed analysis of these problems is given by Wernicke (2005).

FANMOD is a tool for network motif detection that implements a novel algorithm called RAND-ESU (Wernicke, 2005) to enumerate and sample subgraphs. This algorithm is orders of magnitude faster than any other existing algorithm for this task, facilitating the detection of larger motifs in bigger networks than previously possible. Moreover, FANMOD allows for motif detection in colored networks, something not possible with other existing tools.

2 COMPARISON WITH EXISTING TOOLS

We are aware of two tools that perform somewhat similar tasks as FANMOD and allow for the detection and analysis of network motifs in directed and undirected networks, namely MFINDER (Kashtan *et al.*, 2002) and MAVISTO (Schreiber and Schwöbbermeyer, 2005). Some works also mention PAJEK (Batagelj and Mrvar, 2003) in this context, a multi-functional tool for network analysis. However, PAJEK is of limited use in network motif analysis; while it supports the search for all occurrences of a certain pattern in a network, the enumeration of subgraphs and statistical comparison with random graphs are not sufficiently supported.

Both MFINDER and MAVISTO support the detection of network motifs consisting of up to eight vertices, but otherwise these tools have a different focus: MFINDER is a command-line tool that is concerned solely with the detection of network motifs whereas MAVISTO visualizes occurrences of a motif in a network by a force-directed graph layout algorithm. MFINDER also incorporates a broad range of random graph models for determining the frequency of subgraphs in random graphs. A tool named MDRAW has recently been released in order to visualize the output of MFINDER, it is available from the same website as MFINDER.

The main disadvantage of using MFINDER and MAVISTO for network motif detection is that the employed algorithms for subgraph enumeration and sampling (the latter being only supported by MFINDER) are comparably slow and scale poorly as the subgraph size increases. As an example, on a laptop equipped with a 1.5 GHz Pentium M processor and 512 MB RAM, enumerating all 1.4×10^6 size-5 subgraphs in the transcriptional network of *Escherichia Coli* (Shen-Orr *et al.*, 2002) requires 620 s with MAVISTO whereas MFINDER takes 180 s. Our tool FANMOD performs this task in less than 10 s.

Other advantages of FANMOD over the two existing tools include the ability to handle networks with colored vertices and

*To whom correspondence should be addressed.

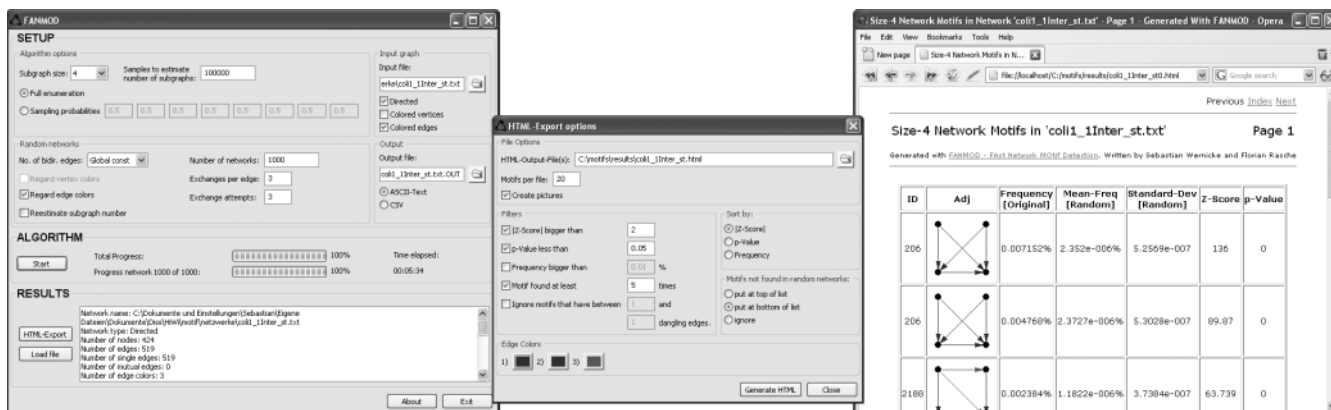


Fig. 1. Detecting size-4 network motifs with colored edges in the transcriptional network of *E. Coli* using the FANMOD interface (left). Via an export filter (middle), the obtained results can be exported to HTML (right).

edges in order to model different types of interactions between different kinds of entities (e.g. to find motifs in protein–gene interaction networks) and the ability to accurately predict the overall running time of its motif detection algorithm (contrary to previous algorithms, RAND-ESU allows for a quick and accurate estimation of the total number of size- k subgraphs in a given network). While FANMOD does not incorporate a module for visualizing concrete appearances of motifs in a network, various output formats (including comma-separated values and HTML) facilitate the analysis and further processing of results.

3 IMPLEMENTATION AND FEATURES

The FANMOD tool is written in the C++ programming language and consists of approximately 7000 lines of non-library code. The graphical user interface and other system-dependent features are implemented with the wxWIDGETS framework (Smart *et al.*, 2005), which is available for a broad range of platforms including Linux, Mac OS and Windows.

FANMOD can detect network motifs up to a size of eight vertices. For this, all subgraphs of the given size are either enumerated or uniformly sampled in the input network using the algorithm described by Wernicke (2005). The subgraphs are grouped into isomorphic subgraph classes using an implementation of the canonical graph-labeling algorithm NAUTY (McKay, 1981). Finally, FANMOD determines the frequency of subgraph classes in a user-specified number of random graphs. The random graphs are generated from the original network by switching edges between vertices; the user may choose between different switching schemes in order to preserve certain graph properties (such as the number of bidirectional edges in directed networks) during the randomization.

For colored networks, motifs of size up to seven vertices (depending on the number of colors that are used for edges and vertices) can be detected. The speed of the tool is not affected by colors, in general it is even a little faster because the canonical graph labeling is facilitated. The random networks can optionally preserve the number of edges between vertices of different colors.

The calculated significance of each subgraph in the network (expressed as P -Values and Z -Scores with respect to the generated random networks) can be exported to a variety of formats. An

HTML export function with various filters (e.g. a filter for avoiding so-called ‘dangling motifs’ that contain degree-one vertices) allows for the quick inspection and distribution of results (see Fig. 1 for an example).

ACKNOWLEDGEMENTS

This work was supported by the Deutsche Telekom Stiftung and the Deutsche Forschungsgemeinschaft (DFG) project PEAL (parameterized complexity and exact algorithms, NI 369/1).

Conflict of Interest: none declared.

REFERENCES

- Albert, I. and Albert, R. (2004) Conserved network motifs allow protein–protein interaction prediction. *Bioinformatics*, **20**, 3346–3352.
- Batagelj, V. and Mrvar, A. (2003) Pajek—analysis and visualization of large networks. In Jünger, M. and Mutzel, P. (eds), *Graph Drawing Software*. Springer-Verlag, pp. 77–103.
- Itzkovitz, S. *et al.* (2005) Coarse-graining and self-dissimilarity of complex networks. *Phys. Rev. E*, **71**, 016127.
- Kalir, S. *et al.* (2001) Ordering genes in a flagella pathway by analysis of expression kinetics from living bacteria. *Science*, **292**, 2080–2083.
- Kashtan, N., Itzkovitz, S., Milo, R. and Alon, U. (2002) Mfinder tool guide. *Technical report*, Department of Molecular Cell Biology and Computer Science and Applied Mathematics, Weizman Institute of Science, Israel.
- Kashtan, N. *et al.* (2004) Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. *Bioinformatics*, **20**, 1746–1758.
- McKay, B.D. (1981) Practical graph isomorphism. *Congr. Numer.*, **30**, 45–87.
- Milo, R. *et al.* (2004) Superfamilies of designed and evolved networks. *Science*, **303**, 1538–1542.
- Milo, R. *et al.* (2002) Network motifs: simple building blocks of complex networks. *Science*, **298**, 824–827.
- Schreiber, F. and Schwöbbermeyer, H. (2005) Mavisto: a tool for the exploration of network motifs. *Bioinformatics*, **21**, 3572–3574.
- Shen-Orr, S.S. *et al.* (2002) Network motifs in the transcriptional regulation network of *Escherichia Coli*. *Nat. Gen.*, **31**, 64–68.
- Smart, J., Hock, K. and Csomor, S. (2005) *Cross-Platform GUI Programming with wxWidgets*. Prentice Hall PTR.
- Vespignani, A. (2003) Evolution thinks modular. *Nat. Gen.*, **35**, 118–119.
- Wernicke, S. (2005) A faster algorithm for detecting network motifs. In *Proceedings of the 5th Workshop on Algorithms in Bioinformatics (WABI '05), Lecture Notes in Bioinformatics*. Vol. 3692, pp. 165–177. (Long version submitted).