FANMOD

FAST NETWORK MOTIF DETECTION

– MANUAL –

Florian Rasche and Sebastian Wernicke

Institut für Informatik, Friedrich-Schiller-Universität Jena,
Ernst-Abbe-Platz 2, D-07743 Jena, Germany
{m3rafl,wernicke}@minet.uni-jena.de.

June 12, 2006

1 Introduction

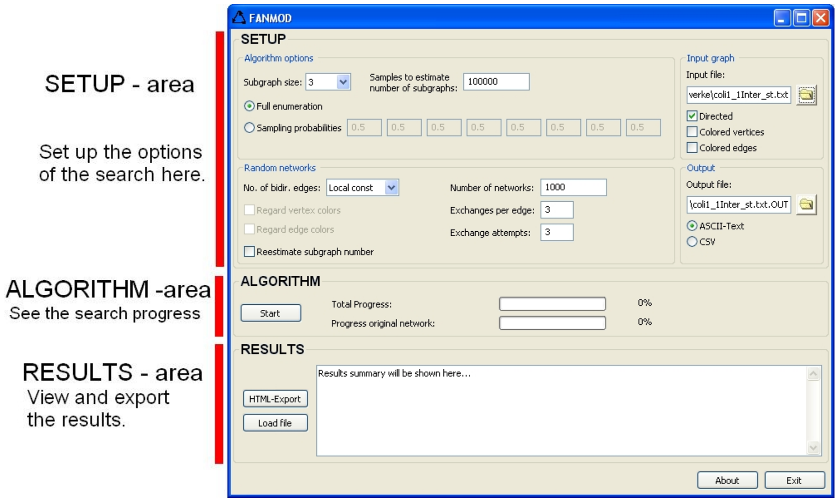
FANMOD [6] is a tool for finding so-called *networks motifs* in a network, that is, it finds small vertex-induced subgraphs that occur significantly more often than in random networks. For a general introduction to the concept of network motifs, see [3, 2].

FANMOD is able to search for network motifs of size between three and eight vertices. It can also analyze colored networks, allowing the user to bring more information into the network rather than just connectivities. FANMOD detects motifs by using the so-called RAND-ESU algorithm which makes it faster than similar programs based on other algorithms, especially for larger motifs. More details on the RAND-ESU algorithm can be found in [5]. FANMOD features a graphical interface for easy setup of algorithm parameters; the results can be exported to HTML.

Even though this manual has become quite long by covering all the details, the use of FANMOD should be quite intuitive. We have paid attention to giving all parameters sound default values, so if you wish to start detecting motifs right away the short introduction in Section 3 should be all you need. Whenever you want to tune the parameters, find out more about some details of the program, or if anything else remains unclear, please consult the more detailed sections:

- Section 4 deals with the algorithm parameters you can set before running the main algorithm.
- Section 5 details what happens when the algorithm is running and how it can be influenced.
- Section 6 provides details on how to view and process the results that are output by the main algorithm.
- Section 7 provides the necessary details on working with colored networks.

Happy motif detecting!



SETUP - area

Set up the options of the search here.

ALGORITHM -area

See the search progress

RESULTS - area

View and export the results.

Subgraph size: 3

Samples to estimate number of subgraphs: 100000

● Full enumeration

○ Sampling probabilities 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5

Input file: verke\coli1_1Inter_st.txt

☑ Directed

□ Colored vertices

□ Colored edges

No. of bidir. edges: Local const

□ Regard vertex colors

□ Regard edge colors

□ Reestimate subgraph number

Number of networks: 1000

Exchanges per edge: 3

Exchange attempts: 3

Output file: \coli1_1Inter_st.txt.OUT

● ASCII-Text

○ CSV

Total Progress: 0%

Progress original network: 0%

Results summary will be shown here...

Fig. 1. The three areas of the main interface window of FANMOD. They correspond to the three working steps when using the tool—setting up the algorithm, running the algorithm, and processing the results.

2 License

You may use FANMOD for all of your research purposes as long as you do the following:

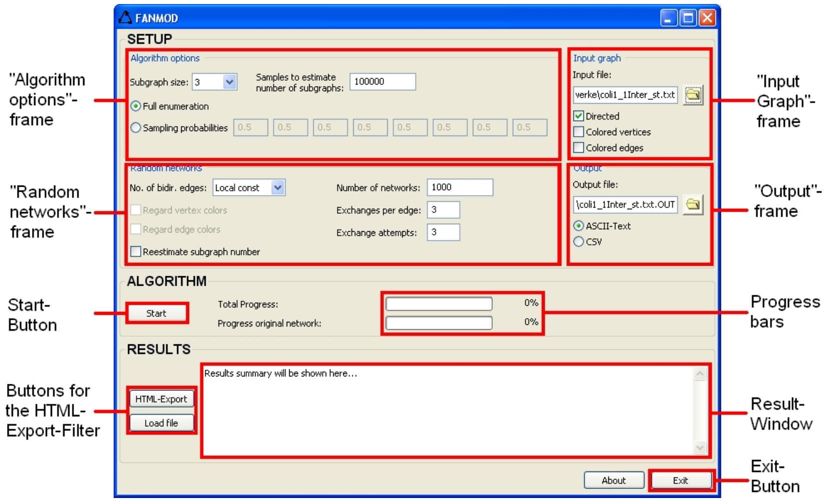
- You acknowledge its use in any published work that fully or in part derives its results from its usage. (If you wish to make a bibliographical reference, please consider citing [5] for that purpose.)
- You do not sell anything that is based on or derived from your use of FANMOD.

Absolutely no guarantees or warranties are made concerning the suitability, correctness, or any other aspect of the distributed files. Any use is at your own risk. Note that FANMOD uses the NAUTY program version 2.2 by Brendan McKay [1]; hence NAUTY's license restrictions also apply to your use of FANMOD.[1]

3 Quick Start

This section describes the basic steps of a motif search without going too much into detail; we have tried to give all options in FANMOD sound default values. Hence, this section is perfect for some first steps with FANMOD—and for those who simply cannot wait to find their first motifs ;-).

[1] More specifically, this means that you may not detect motifs for any application with nontrivial military significance as NAUTY's license forbids that.



FANMOD

SETUP

Algorithm options

Subgraph size: 3

Samples to estimate number of subgraphs: 100000

● Full enumeration

○ Sampling probabilities 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5

Input graph

Input file: verke\coli1_1Inter_st.txt

☑ Directed

□ Colored vertices

□ Colored edges

Random networks

No. of bidir. edges: Local const

□ Regard vertex colors

□ Regard edge colors

□ Reestimate subgraph number

Number of networks: 1000

Exchanges per edge: 3

Exchange attempts: 3

Output

Output file: \coli1_1Inter_st.txt.OUT

● ASCII-Text

○ CSV

ALGORITHM

Total Progress: 0%

Progress original network: 0%

RESULTS

Results summary will be shown here...

Fig. 2. The frames and buttons of the main window, named as they are referenced in the text.

Figure 1 shows the main window that appears when you start FANMOD. The interface follows a top-down logic, i.e., the upper half is for setting the algorithm parameters and the lower half for running the algorithm and viewing/processing its results. A more detailed subdivision of these *areas* into *frames* is shown in Figure 2.

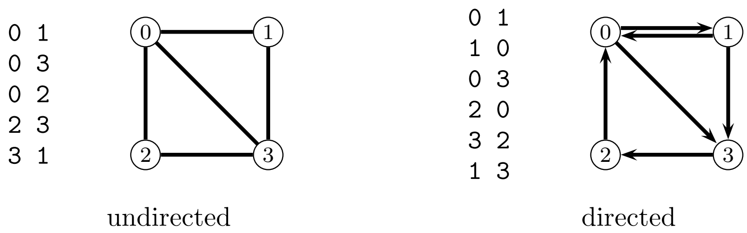
3.1 Selecting an Input Graph

The first step is to choose an input file that describes the network you would like to analyze. You can select this file within the `input graph` frame. Also, you can select whether the network is directed and whether the vertices and/or edges are colored. In this quickstart section, we will only deal with uncolored networks for the sake of simplicity. If you wish to analyze colored networks, please read Section 7.

The input file should contain one line for each edge in your network, each line must contain at least two integers. It may contain up to five integers, but that is only important for colored networks. Hence, a line in the input file looks like this:

```
int1 int2 [int3] [int4] [int5]
```

If the network is *undirected*, the line is interpreted as an edge between the vertices labelled `int1` and `int2`; if the network is *directed*, it is interpreted as a directed edge from vertex `int1` to vertex `int2`. This interpretation is illustrated in Figure 3. Note that it is recommended that the vertices in the network are



```
0 1
0 3
0 2
2 3
3 1
```

undirected

```
0 1
1 0
0 3
2 0
3 2
1 3
```

directed

Fig. 3. Interpretation of the input file for undirected (left) and directed networks (right).

labelled with consecutive integers, the lowest integer being zero (see Section 4.1 for details).

3.2 Setting the Options

The most important options of the motif search are in the `algorithm options` frame. Here you can choose the subgraph size (that is, the size of the motifs to be detected) and whether to perform a full enumeration of subgraphs or just sample some of them in the network. For information on sampling and the appropriate parameters, please refer to Section 4.2.

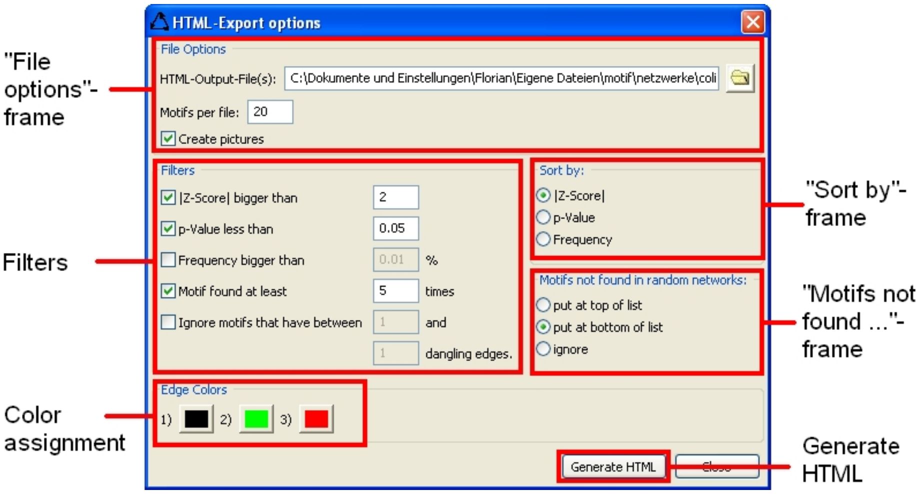
As motifs are detected by comparing their frequency of occurrence in the original network to their frequency of occurrence in a number of similar, yet randomized networks, you have to state how many random networks should be looked at and how they should be created. The number of randomized networks can be set in the `number of networks` box on the right of the `random networks` frame. An explanation of the other available options is deferred to Section 4.3.

In the `output` frame you can set the file where the output is to be written to and select which file format is to be used. As a default, an output file name is derived from the input file.

3.3 Performing the Search

If you have followed the instructions of the previous subsections, you are now ready to start detecting network motifs. Simply click on the `start` button in the `algorithm` area of the program. The progress bars show the overall algorithm progress and the progress per network. An elapsed time shows how long the search has already run. Get yourself a cup of coffee[2] and wait until the program is done. . .

[2] If the progress bars should indicate this and you like coffee, you can get more coffee. The Chair for Theoretical Computer Science I endorses tea.


HTML-Export options

File Options

HTML-Output-File(s): C:\Dokumente und Einstellungen\Florian\Eigene Dateien\motif\netzwerke\coli

Motifs per file: 20

☑ Create pictures

Filters

☑ |Z-Score| bigger than 2

☑ p-Value less than 0.05

□ Frequency bigger than 0.01 %

☑ Motif found at least 5 times

□ Ignore motifs that have between 1 and 1 dangling edges.

Sort by:

● |Z-Score|

○ p-Value

○ Frequency

Motifs not found in random networks:

○ put at top of list

● put at bottom of list

○ ignore

Edge Colors

1) ⬛ 2) 🟩 3) 🟥

Fig. 4. The frames and buttons of the filter window, named as they are referenced in the text.

3.4 Exporting the Results to HTML

Once the algorithm is completed, you can see some brief information about its run in the results box. The best way to view and analyze the found motifs is by using the HTMLexport- and filter tool. Click on `HTML export` in the `results` area to start it.

The layout of the HTML export filter dialogue is shown in Figure 4. You can choose an HTML file name if the default that has been set for you does not seem convenient. Within the `filters` frame of the window, you can switch some filters on and off (and set their threshold values) that determine which found subgraphs will be written to the HTML file. In the `sort by` frame, you can select by which values your results (i.e., the subgraphs that were found) should be sorted. For a more detailed description of the filter tool, please refer to Section 6.3. If you are happy with your filter settings, click the `generate html` button. Your file is then being generated and the pictures of the motifs are drawn. This can take a while, especially if your filters are not very strict and many subgraphs have to be drawn.

If you are a user of Mac OS or Windows, the index file of the generated HTML files is automatically opened in your default browser once they have been completed. As there is no such thing as a default browser under Linux, Linux users will have to open the file themselves. At the bottom of the index file, there are links to the result pages. These contain the motifs found and their statistics.

This concludes our quickstart guide to FANMOD. Now you are welcome to go into all the gory details in the next sections. Especially Section 7, which deals with finding motifs in *colored* networks, might be of interest because the ability to handle colors is unique to FANMOD up to now (compared to other motif detection tools).

4 Algorithm Parameters

This section covers the options which are available before starting a motif search. They can all be found in the `algorithm` area of the main interface.

4.1 Input File Options

The Input File The input file should contain numbers only. The tool assumes that the input file is well-formatted and may not behave gracefully if this isn't so. Each edge in the network is represented as one line in the input file. This line may consist of up to five integers, the first two of which are mandatory. So a line should look like this:

```
int1 int2 [int3] [int4] [int5]
```

If the network is *undirected*, the line is interpreted as an edge *between* the vertices labelled `int1` and `int2`; if the network is *directed*, it is interpreted as a directed edge *from* vertex `int1` *to* vertex `int2`. If you would like to specify a bidirected edge, just add another line for an edge with the opposite direction. The interpretation of line is illustrated in Figure 3.

Note that it is highly recommended that the vertices in the network are labelled with consecutive integers, the lowest integer being zero (otherwise, memory is wasted and the vertex count of the tool is not correct).

Input File Textbox Here you can simply key-in the filename of the network file. It can be absolute or relative to the path containing the executable.

Input File Selection Button When you click this button, a file selection window pops up. Here you can choose the file containing the input network. The look and feel of this window is OS dependent, so you should feel comfortable with it (that is, assuming you feel comfortable with your OS).

Directed Checkbox With this checkbox you can choose whether your network shall be handled as directed, which is the default, or as undirected. When unselecting this check box, all the edges in the network file will be treated as undirected.

When specifying the input network as undirected, there is no further option to specify how to handle the number of bidirectional edges during randomization because directions no longer are relevant.

Colored Vertices Checkbox With this checkbox you can state that the network has colored vertices/nodes (depending which term you prefer to use). If the network file has specified vertex colors, those will now be regarded. If this option is not selected, even an input file containing a vertex-colored graph will be treated as if those colors were not specified. Please refer to Section 7 to learn exactly how to specify colors and how many colors may be used.

Colored Edges Checkbox This acts quite similar to the `Colored Vertices` checkbox. You can toggle the use of edge colors here. Again, please refer to Section 7 to learn some important details about color specification and limits.

4.2 Subgraph Detection Options

Subgraph Size Drop Down Menu The size of the motifs to be searched is selected here. It may range from three up to eight vertices. Be aware that although FANMOD is quite fast, detecting large motifs may take very, very long. Also, you should have as much memory as possible available on your machine.

If you choose to detect motifs in colored networks, note that the number of colors you may use is limited by the motif size (see Table 1 in Section 7).

Samples Textbox Here you can select how many samples shall be taken in order to estimate the number of subgraphs (motif candidates) that will be found during the search (see [5] to find out how this works). On the one hand, as this estimation is used for the progress bars, these become more accurate with a higher number of samples. On the other hand, this option does not influence the final result. Therefore, it is recommended to simply keep the default which has proven itself to be sufficient for most practical cases.

Full Enumeration Radiobutton Choose this if you want *all* subgraphs (i.e., motif candidates) of the selected size to be counted/enumerated in the input network. This will ensure the most precise results but is of course slower than taking only a sample of subgraphs into account.

Sampling Probabilities Button and Text Fields If you choose this option, the algorithm will only sample a subset of all subgraphs in the network. In the fields, you can type a set of probabilites; we strongly recommend that you read [5] for a detailed and exact explanations of these parameters. However, if you do not wish to read [5], then the following sums up the meaning of the parameters:

If you multiply the probabilities, you get the probability for one single subgraph in the network to be sampled. So the default of 0.5 is not to be kept for good results. The probabilities are used in the algorithm as follows: First a seed/starting vertex for an subgraph is selected. Each vertex becomes a seed with the first probability. Then a list of suitable neighbours for this vertex is

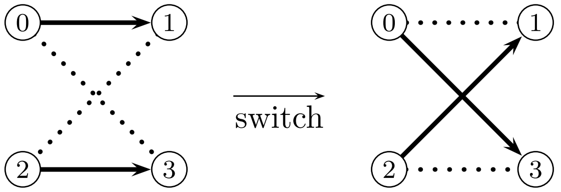


Fig. 5. Example for an edge-switch operation in order to randomize the network.

generated. Here, each neighbour is further considered with the second probability. Then a list of suitable neighbours of these vertices is generated and the third probability applied. And so on...

To spread the samples evenly in your network, you should choose high probabilities in the left fields and lower probabilities in the right fields.

4.3 Random Network Options

Number of Bidirectional Edges Dropdown Menu Here you can choose which model is to be used for random network generation. In all cases, the random network is generated from the original network by a series of edge switching operations (see Figure 5 for an example). Also in any case, the degree sequence of the network is preserved.

With directed networks, the following randomization models are available:

- `Local const.` In this model, unidirectional edges are only exchanged with unidirectional ones. The same applies for bidirectional edges. Therefore, the number of incident bidirectional edges remains locally constant, that is, it constant for each vertex. This is the default randomization model and the only one which can be applied to undirected networks.
- `Global const.` This means that the number of bidirectional edges is kept constant in the overall network. However, a specific vertex may loose or gain incident bidirectional edges.
- `No regard.` This states that bidirectional edges can be created and destroyed during randomization. This option usually increases the number of bidirectional edges compared to the original network, which is often unwanted because it makes unidirectional edges falsely appear significant.

Figure 6 further illustrates these different randomization modes.

Regard Vertex Colors Checkbox This option is only available if the input network has vertex colors, which can be specified by the input options. When activated, edges are only exchanged if their endpoints have the same color. E.g., an edge from a red to a blue vertex is only exchanged with another edge from a red to a blue vertex. See Figure 7 for an illustration of this. Having this option activated is the default when working with vertex colors.

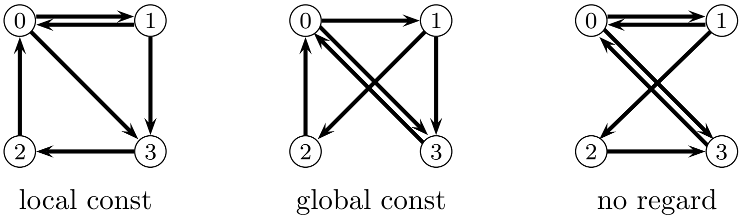


Fig. 6. Illustration for the different randomization modes when the right network from Figure 3 is randomized.

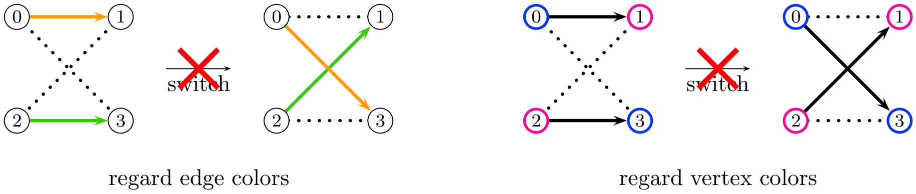


Fig. 7. When colors are regarded, the edge switches are restricted to preserving the color of edges and the colors of endpoints.

Regard Edge Colors Checkbox This option is only available if the input network has edge colors, which can be specified by the input options. When activated, edges are only exchanged with edges of the same color, thus keeping the color degree of a vertex constant (i.e., if vertex has, say, two outgoing red edges, this property will not change during randomization). See Figure 7 for an illustration of this. Per default, this option is activated when using edge colors.

Number of Networks Here you set the number of random networks that are to be generated and analyzed in order to obtain a comparison to the original input network. The higher the number, the more accurate the results. But to search a random network takes approximately as long as searching the original input, so a high number requires more calculation time. As a general rule, consider increasing this number if there are a lot of motifs with undefined Z-Score among the results.

Exchanges per edge When randomizing the network, the edges are exchanged one after the other. This number states how often the program walks over all the edges. Usually the default of three fits here, but if the results from the random networks are too similar to those of the original one, then you should increase this number.

Exchange attempts When it is an edge's turn to be exchanged, an exchange partner edge is randomly selected from those edges that fulfil the desired properties (biderected/unidirected, correct vertex/edge colors). For some reasons, this partner may not be suitable for the exchange (e.g., if both edges start at the same vertex, then an exchange would not change the network at all). In this case, another partner is selected. This is repeated until the exchange succeeds or the number of exchange attempts—which can be specified in this box—is exceeded. So, if too few exchanges succeed (the number of successful exchanges can be found in the results summary), this number should be increased.

But keep in mind that a lot of unsuccessful exchanges may also result from the input network structure.

4.4 Output Options

Output File This is quite similar to the input file settings. Either you key-in your desired output file name directly or you use the selection button. The dialogue that pops up works exactly like a `save as` dialogue on your platform. When you specify the input file, a default is given here, appending either "`.OUT`" for textfiles or "`.csv`" for CSV files to the input file name.

ASCII-Text Radiobutton If this button is activated, the output will be a human readable plain text table—useful if you want to look at your results in a text editor. The plaintext can be exported into HTML as described in Section 6.3.

CSV Radiobutton If this button is activated, the output will be a comma-separated-values (CSV) table which can easily be read by most spreadsheet applications—but is "a bit" difficult to read for humans. HTML export is possible from this format just as for ASCII.

5 During the Search

5.1 The Status Bars

In the `algorithm` area, there are two status bars which show quite accurately how far the search has progressed. The upper one gives you the total progress for all networks, the bottom one shows the progress of the network that is currently being processed. Note that the accuracy of the progress bars depends on the number of samples you have entered into the `number of samples to estimate subgraphs` box.

If the networks are being searched really fast, the second progress bar might seem disabled because it takes longer to update it than to search a single network.

Next to the progress bars, there is a clock showing the elapsed time of the search. It is not accurate to the second because it keeps getting confused by the search running in the background. (We use a more accurate clock for the time measurement reported in the output file.)

5.2 The Result Window

During the search, intermediate results—such as the number of vertices and edges in the input network, the estimated subgraph number, and the number of subgraphs that have been found in the original network—will appear in the `results` area. Once the search has been finished, a result overview is shown in this window. The detected subgraphs and statistics, though, can only be found in the output file.

5.3 Pausing or Aborting the Search

During the search, there are only two active buttons: The former `start` button can now be used to `pause` the search (if you need your precious calculation time for something else). The second option is more final: `abort`ing the search by clicking onto the former `exit` button ends the search—all the intermediate results are then lost.

6 Viewing the results

FANMOD offers two ways to view results: Either you read the ASCII or CSV file directly, or you use the filter and HTML generator.

6.1 The Values of the Result Table

The result table has the following columns:

- `Motif ID.` The Motif ID is characteristic integer for the motif and independent of its colors (i.e., motifs with the same ID are always isomorphic if color is ignored); it is obtained by interpreting the corresponding 0-or-1-adjacency matrix as a binary integer.
- `Adj(acency Matrix).` The adjacency matrix of the subgraph. An edge is directed from row to column. Zero denotes that there is no edge, any other number codes the edge color. The diagonal of this matrix codes the vertex colors, using the hexcode-digits A to F if more than ten vertex colors are used. In the HTML output, images are drawn to replace the adjacency matrices if the `pictures` option was checked.
- `Frequency [Original].` The third column of the result table denotes the frequency with which a motif occurred in the original network.
- `Mean-Frequency [Random].` The mean frequency with which the motif occurred in random networks.
- `Standard-Deviation [Random].` The standard deviation from the mean frequency is noted here.
- `Z-Score.` The Z-Score is the original frequency minus the random frequency divided by the standard deviation. If you find an undefined value here, the standard deviation was zero. If there are a lot of undefined Z-Scores, consider generating more random networks or increasing the random parameters like `exchanges per edge`.

- `p-Value`. The p-Value of a motif is the number of random networks in which it occurred more often than in the original network, divided by the total number of random networks. Therefore, p-Values range from 0 to 1; the smaller the p-Value, the more significant is the motif.

6.2 In the ASCII or CSV File

The ASCII or CSV file contains the output as produced by the core algorithm. At the beginning of the file, the options of the search and properties of the input network are given, as well as the total number of subgraphs found and how long the search took. Then follows the result table which includes all the subgraphs found (no matter how rare or insignificant they were). The table is either human-readable or in CSV. The CSV format can easily be read by a spreadsheet application where you can sort, filter, or scan it for certain values.

6.3 Using the HTML-Export-Filter

Compared to the pure text outputs, the more comfortable way to view the results is to use the filter and HTML export tool of FANMOD. The HTML generator also draws the motifs for you, so you do not have to worry about interpreting adjacency matrices. A screenshot of the filter is shown in Figure 4.

Loading an output file into the filter If you have just finished a motif search and now want to export its results to HTML, the proper file is automatically loaded in when you click `HTML-export` in the `results` area. If you want to analyze the results of a previous search, you have to load the respective ASCII or or CSV file first. For this, first click on `load file` in the `results` area and then `HTML-export` to open the export window.

General HTML-Output Options In the `file options` frame of the export window, the file name of the resulting HTML file can be entered. The default is just to use the ending "`.html`" with the existing ASCII file name. You can also browse through your directories by clicking the folder icon. In the frame, you can also select the number of motifs which shall be shown per HTML file. FANMOD generates an index file and as many result files as are necessary to show all motifs matching your criteria. This avoids files becoming too large.

The last option in this area to be discussed is whether pictures shall be drawn or not. If you expect an excessive amount of output motifs, better switch pictures off as drawing takes some time and the pictures need some storage space.

Filters The filters can be toggled on and off by the corresponding checkbox. You can filter according to the following criteria:

- `Z-Score` The higher the Z-Score, the more significant is a motif. Here you can select to show only motifs that have a Z-score higher than a specified value. As the absence of a motif in the original network can also be significant, the absolute value of the Z-score is used here, please use the p-Value filter if you are not interested in negative Z-scores (subgraphs with negative Z-scores have high p-Values).
- `p-Value` The lower the p-Value, the higher the significance. Therefore, all motifs above a certain p-Value can be filtered out here.
- `Frequency` Show only motifs that occur at a certain frequency or higher in the original network.
- `Absolute occurrence` Here you can specify that any motif shown has to occur at least a certain number of times in the original network in order to be written to the HTML output.
- `Dangling edges` A dangling edge in a motif is one incident to a degree-one vertex. These motifs are often considered as uninteresting, as you end up with a smaller motif when removing this dangling edge. A motif with two dangling edges (two degree-one vertices) might be a path and therefore again be interesting. That is why you can enter upper and lower bounds in this filter.

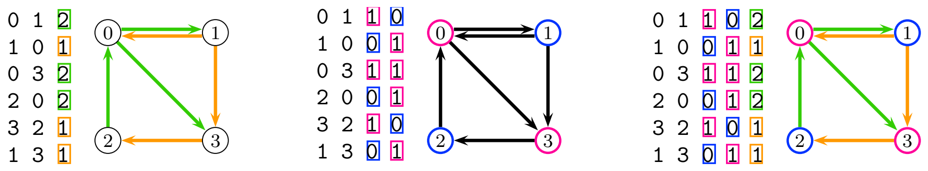
Sorting Options An additional option is which value is used to sort the results. In the `motif not found...` frame you can choose how motifs that were not found at all in the random networks shall be handled. For these motifs, the Z-Score is $+\infty$ (undefined), so they would appear on top of the list if sorted by Z-Score, which can be quite annoying as it is hard to further evaluate their usefulness due to lack of data. If you find one of these motifs interesting, you should consider selecting different random options to create proper statistics for it.

Choosing the Colors If the original network had edge and/or vertex colors, you can select which number in your network file corresponds to which RGB color. Some default colors are set. By clicking on a colored rectangle, you can choose a different one. These colors are used to draw the pictures. A color legend is provided in the index file, even if no images are drawn.

Generating the HTML-Output The HTML output is generated by clicking the appropriate button. The pictures are also drawn if this was specified. If your operating system knows a default browser, the created index file is opened. Otherwise, you have to open it by hand. In the index file, the general information about the search is given as well as links to all the result pages. In all result pages you can jump to the previous or next page, as well as back to the index page.

6.4 Organisation of the Generated Files

When an HTML output is generated, the following files are created:



```
0 1 2
1 0 1
0 3 2
2 0 2
3 2 1
1 3 1
```

```
0 1 1 0
1 0 0 1
0 3 1 1
2 0 0 1
3 2 1 0
1 3 0 1
```

```
0 1 1 0 2
1 0 0 1 1
0 3 1 1 2
2 0 0 1 2
3 2 1 0 1
1 3 0 1 1
```

Fig. 8. Examples for specifying edge and vertex colors in the input file.

1. An index file which has the file name you originally specified.
2. Result files, for which a number is appended to your specified filename. These numbers start at zero.
3. A folder/directory containing the motif images if you asked for them to be drawn. For the directory name, "-images" is appended to the index file name. Inside this folder, the pictures are named according to the adjacency matrix of the network they depict.

If you want to copy or move HTML exported results, be sure to move all the files mentioned above.

7 Colored Networks

As often mentioned before, FANMOD is able to search colored networks for colored motifs. Both vertices and edges may be colored, so a great variety of information can be included into the network. E.g., you might have a biological network where the vertices correspond to genes (one vertex color) or proteins (another vertex color), and between those exist inhibiting or activating interactions (two edge colors).

7.1 Specification of Colors in the Input File

Colors are specified by using more than two integers per line in the input file. The first two integers code the begin and end vertex of an edge as already discussed. The other integers have different meanings depending on how many you put in a line, the specifications are also illustrated in Figure 8:

- **Five integers.** The third number gives the color of the first vertex in this line. The fourth number is the color of the second vertex and the fifth one denotes the edge color.
- **Four integers.** As above, the third an fourth number denote the vertex colors. The edge will have the default edge color 1.
- **Three integers.** The third number specifies the color of the edge, the vertices will keep their colors specified in another line or retain the default vertex color 0 if none was specified before.

Subgraph Size	number of different colors (if only vertices colored)	number of different colors (vertices / edges)
3	16	16 / 7
4	16	16 / 7
5	16	8 / 3
6	16	1 / 3
7	4	1 / 1
8	1	1 / 1

Table 1. Limits on the number of different colors that a network may have. Note that this number not only depends on the subgraph size, but also on whether *only* vertices are colored (middle column) or edges and vertices are *both* colored (right column).

– **Two integers.** The edges will have the default edge color 1 and the vertices will keep their colors specified in another line or have the default vertex color 0 if none was specified before.

Two important notes:

1. The smallest and default vertex color is 0, the smallest and default edge color is 1. (The reason for this is that there has to be a possibility to code “no edge”, which is done by the edge color 0.)
2. For directed networks, you may give bidirectional edges different colors for each direction.

7.2 Color Limits

Due to limitations of FANMOD’s data structure, only a certain number of colors can be used when searching for motifs of a certain size. These limits can be found in Table 1.

7.3 Special Options When Using Colors

In the Input Options You have to activate the `colored vertices` and/or `colored edges checkboxes` if you want to search for colored motifs. It is not sufficient to just specify the colors in the network file.

Having to activate colors explicitly is useful if you have a colored network in a file but want to interpret it as being uncolored.

For the Random Model You can activate `regard vertex colors` and/or `regard edge colors` when vertex/edge colors are used. Those restrict the random model in that only edges leading from and to vertices of the same color and edges of the same color may be exchanged. For details on these options, read Sections 4.3 and 4.3 and refer to Figure 7.

In the Export Filter RGB colors may be assigned to the the numbers used in the input file. This is done by clicking on the colored rectangles in the HTML export window. More information on this can be found in Section 6.3.

Acknowledgements

Florian Rasche's position as a student research assistant at the Chair for Theoretical Computer Science I of the Universität Jena is funded by the Deutsche Forschungsgemeinschaft project PEAL (Parameterized Complexity and Exact Algorithms), NI 369/1. Sebastian Wernicke's PhD research is funded by a scholarship from the Deutsche Telekom Stiftung.

We used Brendan McKay's NAUTY algorithm for network isomorphy [1]. The graphical user interface was built with the WXWIDGETS framework [4] written by Julian Smart and many others—many thanks for the great programming work which made FANMOD excellently portable without a glitch!

Rolf Niedermeier gave us every bit of support needed to develop FANMOD. Michael Dom, Jiong Guo, and Falk Hüffner of the Chair for Theoretical Computer Science I in Jena gave us lots of helping comments and input. We also thank Erik Braun in Jena for various installations of the WXWIDGETS framework and the Apple Beratungszentrum of the Technische Universität München for providing us with a Mac account to build and test the Mac OS version of FANMOD.

References

1. B. D. McKay. Practical graph isomorphism. *Congressus Numerantium*, 30:45–87, 1981.
2. R. Milo, S. Itzkovitz, N. Kashtan, et al. Superfamilies of designed and evolved networks. *Science*, 303(5663):1538–1542, 2004.
3. R. Milo, S. S. Shen-Orr, S. Itzkovitz, et al. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002.
4. J. Smart, K. Hock, and S. Csomor. *Cross-Platform GUI Programming with wxWidgets*. Prentice Hall PTR, 2005.
5. S. Wernicke. A faster algorithm for detecting network motifs. In R. Cassadio and G. Myers, editors, *Proceedings of WABI '05*, number 3692 in LNBI, pages 165–177. Springer-Verlag, 2005.
6. S. Wernicke and F. Rasche. FANMOD: a tool for fast network motif detection. *Bioinformatics*, 22(9):1152–1153, 2006.