A System for Interactive Visual Analysis of Large Graphs Using Motifs in Graph Editing and Aggregation

Tatiana von Landesberger[1,2], Melanie Görner[2], Robert Rehner[2], Tobias Schreck[2]

[1]Fraunhofer-Institut für Graphische Datenverarbeitung
[2]Technische Universität Darmstadt
Email: `tatiana.von.landesberger@igd.fraunhofer.de`,
`{melanie.goerner,robert.rehner}@stud.tu-darmstadt.de`,
`tobias.schreck@gris.tu-darmstadt.de`

Abstract

Network analysis is an important task in a wide variety of application domains including analysis of social, financial, or transportation networks, to name a few. The appropriate visualization of graphs may reveal useful insight into relationships between network entities and subnetworks. However, often further algorithmic analysis of network structures is needed. In this paper, we propose a system for effective visual analysis of graphs which supports multiple analytic tasks. Our system enhances any graph layout algorithm by an analysis stage which detects predefined or arbitrarily specified subgraph structures (motifs). These motifs in turn are used to filter or aggregate the given network, which is particularly useful for search and analysis of interesting structures in large graphs. Our approach is fully interactive and can be iteratively refined, supporting analysis of graph structures at multiple levels of abstraction. Furthermore, our system supports the analysis of data- or user-driven graph dynamics by showing the implications of graph changes on the identified subgraph structures. The interactive facilities may be flexibly combined for gaining deep insight into the network structures for a wide range of analysis tasks. While we focus on directed, weighted graphs, the proposed tools can be easily extended to undirected and unweighted graphs. The usefulness of our approach is demonstrated by application on a phone call data set [18].

1 Introduction

The analysis of large graphs plays an important role in many important research and application areas, e.g., in finance, biology, sociology, transportation, etc. The proper understanding of global and local network structures is an essential aspect in many applications. This in turn may be supported by effective network visualization techniques and by the analysis of a given graph for topologic properties or the occurrence of certain substructures. *Network motifs* are predefined graph patterns, which based on their definition and domain semantics can capture important functional information of interest in an analysis application. For example, in phone call network analysis, the so called feedback motif identifies circular calling relationships among three people. Note that the choice of interesting motifs is application, data and task dependent and may even change throughout an analytic process. A set of selected graph motifs is presented in Figure 1.

In this paper, we present a motif-based graph visualization system focused on three main network analysis scenarios:

1. Analysis of networks for occurrence of user-defined motifs in static graphs;
2. Analysis of network changes in dynamic graphs and their impact on identified motifs;
3. Analysis of network structures on multiple levels of detail based on motif aggregation.

In the first case, it may be interesting to inves-

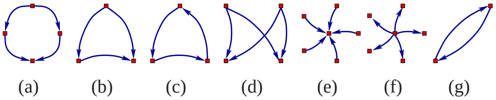

Figure 1: Selected graph motifs. a) Caro, b) Feedforward, c) Feedback, d) Double cross, e) In-Star, f) Out-star and g) Reciprocity.

tigate which structures, how often and where are present in the network. For example, we can identify whether in the network and how often mutual exchange of information takes place, or identify persons who communicate with many people.

In the second case (data driven or user defined graph changes), the analysis focuses on determination of significant changes in relations. In particular, the analysis of implications of data changes on graph structures is relevant. For example, what happens when an important person leaves a network or a new connection between two persons is created? Do new communication structures (e.g. cycles) appear or do old ones disappear?

In the third case, in certain scenarios, analysis of structures created between local relationships may be of interest. For instance, to investigate whether and what types of connections exist between cliques of persons in a social network. Such type of analysis can be interesting at multiple abstraction levels.

In related work, there exist several methods for analysis and visualization of graph motifs and general graph visualization techniques. Graph drawing approaches often disregard graph substructures and do not offer analysis possibilities. Motif-based approaches are in general restricted to predefined motifs, static graphs and do not scale up for large graphs. Moreover, the use of motifs for graph aggregation is not broadly explored yet.

Our goal is to keep the analysis of motifs flexible and interactive, so that it can be easily adapted to varying analytic tasks across application areas. We combine effective state of the art graph visualization with interaction and graph analysis techniques. Our approach is based on the search and the interactive visualization of graph motifs. As various application areas or even analytical tasks require searching for a specific, often a-priori unknown motif pattern, in addition to predefined motifs, interactively user-defined motifs can be considered as well. For the support of dynamic graphs, in particular interactive “what-if-analysis”, the implications of graph changes on local structures are analyzed and visualized. For the analysis of higher level graph structures (and graph simplification), a hierarchical motif-based graph aggregation is introduced. These features can be combined on demand to address various tasks as they emerge in the analysis process.

The remainder of this paper is structured as follows: Section 2 introduces related work in the area of graph analysis and interactive graph visualization with focus on motifs. Section 3 defines graph motifs used in our work. Section 4 describes visual analysis techniques of static graphs using motifs, and details on the use of graph motifs for “what-if-analysis” and graph aggregation. Section 5 shows applications of our techniques on a phone call network from VAST 2008 challenge [18]. Section 6 concludes and outlines future research.

2 Related Work

The *analysis of graph for motifs* is applied mainly in biology, as motifs often play an important role in biologic reactions [17]. Several tools for the analysis of graphs using motifs exist in this area, some also offering simple motif visualization. MAVisto [16] offers a motif search function displaying motif frequencies also in comparison to randomized networks. The distribution of a particular motif is shown using a specified layout. FANMOD [19] allows fast detection of networks motifs and display of found motif with their frequencies. Its functionality is similar to Mfinder with mDraw, and SNAVI [14, 13]. These approaches, however, are computationally intensive, searching for all possible motifs, and do not offer graph visualization, or restrict the drawing to small graphs ($<$100 nodes) only.

Considering the *use of graph motifs for graph visualization*, only few approaches exist, as most graph drawing works focus on effective graph layouts without specific focus on substructures. Holleis et al. [8] introduce a graph layout based on subgraphs. They construct the subgraph layout first and then introduce connections between subgraphs. Also, a motif-preserving layout based on force-direction was presented by Klukas [10]. They show matches of a selected motif from all found motifs in the main network view. Huang et al. [9] present visualizations that highlight found motifs. They concentrate on non-motif parts of the graph by firstly simplifying a selected pre-defined graph motif and then showing motifs on one layer, and the non-motif edges and vertices on a second layer. This creates a 3D view on the graph. The simplification is specific to a selected small set of motifs. Alternative views provide motif disconnection using node duplication and motifs inside transparent spheres.

The visualization of large graphs often uses graph reduction techniques, e.g., graph filtering [5, 12] or graph aggregation [2, 4, 11]. Aggregation approaches usually use a predefined graph hierarchy, not motifs. For laying out large undirected graphs many approaches have been presented e.g. [6, 1]. TopoLayout [3], being mostly related to our work, uses the structural (topologic) information of the graph in the layout algorithm.

In summary, several methods consider analysis and visualization of graphs based on motifs. Motif-based visual approaches are restricted to predefined motifs, static graphs and do not scale up for large graphs. Leveraging free-defined motif analysis for graph aggregation is to the best of our knowledge unexplored yet.

3 Graph Motifs

The analysis of network structures often employs examination of topologic properties of the graph and search in the graph for certain substructures. Motifs are interesting graph patterns as they often carry functional information for different graph classes. In general, motifs occurring with higher frequency than it would be expected for random graphs are of interest. The set of relevant substructures is dependent on the task and application. Nevertheless, there are several structures that are of common interest to multiple scenarios.

We first identified a set of baseline motifs possibly relevant in a large number of applications (see Figure 1). These motifs frequently occur in biological, social or financial networks and are supposed to reveal important information on relationships in the network. For example, reciprocity shows persons calling each other. The presented motifs may not cover all analysis tasks and therefore, in our work these motifs can be interactively extended by user defined structures on demand.

Motifs can be parametrized by posing constraints on the values of edge or node attributes (e.g., edge weight or node label). These parameters are used for filtering of interesting motifs according to their properties. For example, in a phone call network, motifs in which persons call each other longer that a certain threshold can be identified.

The algorithmic determination of motif frequencies for all possible motifs of a certain size is a NP-hard problem [7]. In general exact search is preferred [16, 19]. However in many approaches heuristics are used in order to accelerate the analysis [17, 20]. These heuristics however do not guarantee finding of all motifs and usually are developed for finding all motifs of a certain size. In our case, however finding of all motifs is relevant and concentration on one selected (pre- or user-defined) motif is sufficient. Therefore, we generally follow approach from [7] for finding motifs which uses exact search.

4 Visual Graph Analysis Using Motifs

This section details on our system for interactive visual analysis of graphs focused on motifs. We introduce three scenarios: 1) analysis of occurrence and location of user-specified graph motifs, 2) analysis of implications of graph changes (e.g., for "what-if-analysis" scenarios) and 3) analysis of network structures on various abstraction levels by sequential motif-based graph aggregation.

4.1 Motif Visualization

The visual analysis of graph motifs usually employs motif search and subsequent visualization of found motifs. In our approach, we search for all occurrences of a specified motif either in the whole graph, or in a local area around a focused node or edge. The first case gives a general overview of motif frequencies and distributions in the network (being computationally more intensive) and the second one allows for the analysis of structures in which a specific node or edge is involved (the latter being more focused and less computationally intensive). The motif type can be selected from a set of pre-defined motifs or interactively specified using a visual graph editor (see Figure 2 and 3). Furthermore, the set of motifs that have been found can be filtered in order to focus on structures obeying certain constraints (see Figure 4).

Graph visualization uses node-link diagrams and user-selected graph layout (from JUNG library [15]). The visualization of the detected graph structures employs highlighting of the located motifs in a network. Our approach is layout independent and therefore flexible for application on various use cases. It shows the found motifs in the context of the whole graph or focuses on the found motifs only. In general, simple highlighting of these graph motifs may lead to overloaded views owing

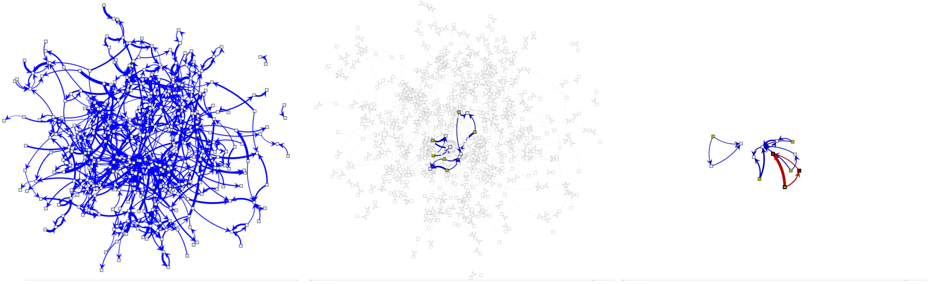

Figure 2: Search and visualization of various pre-defined motifs in a network. Left: Original graph. Center: Caro motif highlighting im context. Right: Feed-forward motif without context with highlighting of a motif included in a node.

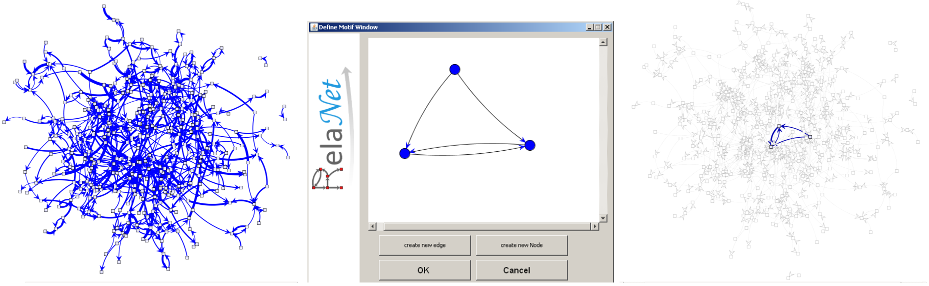

Figure 3: Interactive motif definition (center) and visualization of found user-defined motif with labeled names of persons (right). Original graph is shown on the left side.

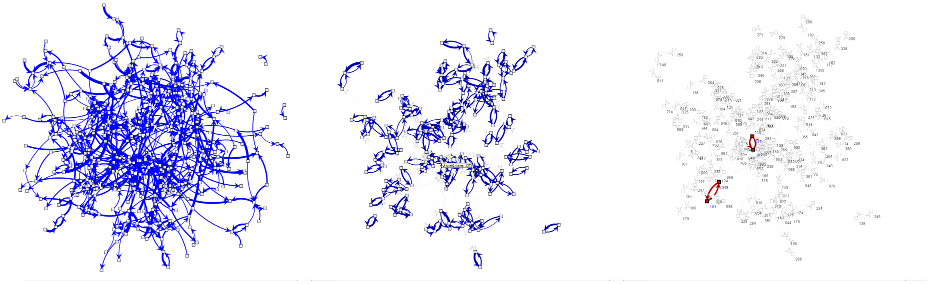

Figure 4: Motif filtering example. Left: Original graph. Center: all occurrences of reciprocity motif. Right: Occurrences of reciprocity motif for edge weights larger than a given threshold.

to high motif occurrence or motif overlap. This can be addressed either by highlighting of a motif including a selected graph entity or motif filtering. In the first case, a selection of a node or edge shows all motifs that include the selected element. It can be used for detailed analysis of particular entities in the graph. In the second case, filtering of motifs using attribute constraints can be employed. For example, motifs which include only edges with weight larger than a certain threshold can be filtered (see Figure 4).

4.2 Visual Analysis of Graph Changes

We also support motif-based visual analysis of graph changes. Usually, when showing graph dynamics, changes in the graph compared to the previous graph are highlighted (e.g., new/changed edges or nodes). This might be insufficient as for the analyst also the impact of those changes on graph structures can be of interest. For example, by adding an edge, a new connection between two persons is created, which may lead to closing a calling chain (see Figure 5). In these cases, motif analysis and visualization helps to discover indirect change effects. After performing a graph change, the modified entities (e.g. new edges) can be locally searched for occurrence of motifs for identification of structural changes in the network.

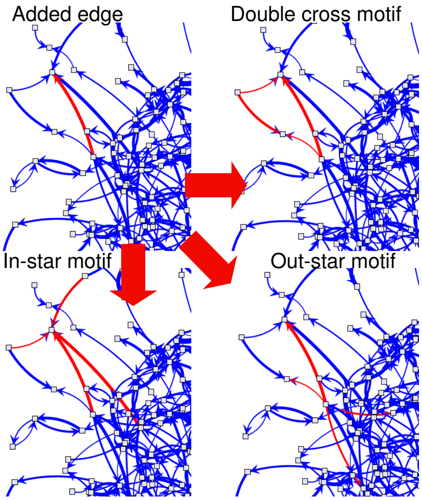

Figure 5: Visual analysis of changes in local graph structures after adding an edge.

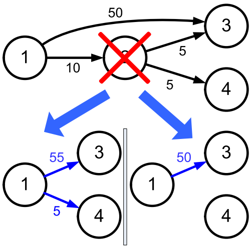

Figure 6: Two examples node deletion implications.

The graph changes can be data or user driven. In the data driven case, the graph changes are loaded from a data source, in the other case, the user performs manual changes to the graph using a graphic interface. In the graph, vertices and edges may be deleted, added or modified (attribute values are changed). In the following, we focus on user defined graph changes (used in “what-if-analysis”) and the visual analysis of its impacts.

When manually applying graph changes, a change in any particular node and edge deletion may imply further needs for modifications to the network (see Figure 6). In case of deleting a vertex, for example, its adjacent edges need to be deleted or redirected, and, possibly, attribute values of the new/other edges may need to be altered. These implications may not be trivial to solve automatically without loosing semantic information. In our work, we use a semi-automatic change management. The computer identifies the need for indirect changes and shows a proposal for the edge redirection, which can be accepted or altered by the user (see Figure 7).

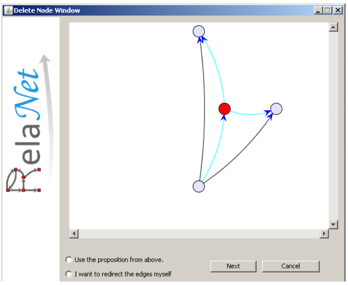

Figure 7: Proposal for edge redirection after deleting a node.

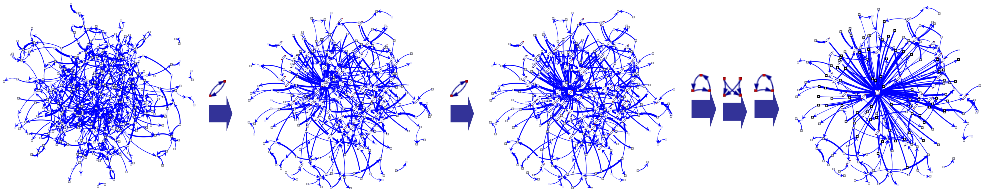

Figure 8: Example of hierarchic motif-based graph aggregation.

4.3 Graph Aggregation Using Motifs

We present a new approach for the visual analysis of large graphs based on graph aggregation. The main advantage of this method is the merging of local (functional) graph substructures. The aggregated graph then shows connections between local structures of the original graph. The aggregation may be done in a hierarchic way, revealing *structures of structures*. Choosing a sequence of graph motifs, graph reductions may be created on demand and on arbitrary level of abstraction. In the following, we introduce motif aggregation and then its visualization.

For motif-based graph aggregation, the graph is first searched for occurrence of a selected motif and then all nodes included in that motif are merged. Each aggregated node thereby includes a particular graph structure (see Figure 9).

In a simple case each found motif is aggregated into a single merged node. However motifs may overlap, i.e., one node or one edge is included in several substructures. For aggregating overlapping motifs, we can either duplicate the nodes in multiple motifs or aggregate multiple connected motifs into one node. In this paper we employ the latter technique.

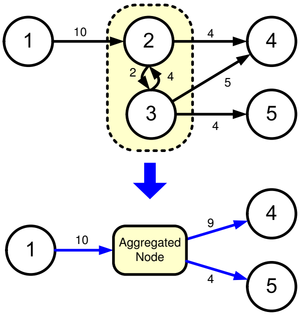

Figure 9: Reciprocity motif aggregation example.

The aggregation process can be successively repeated on the already aggregated graph. A new merged node thereby includes several simple and possibly also already merged nodes. In this way, a “node hierarchy” is created. The type of motif used for aggregation can stay constant or can be changed in each step on demand. In each iteration a more simplified graph (a new aggregation layer) is created. This is especially useful for very large graphs, as the multiple successive aggregation may reveal “higher level“ structures in the original graph (see Figure 8). It needs to be noted, that the order of motif types used for aggregation plays a significant role in the structure of the output graph (see Figure 10). Different orders may lead to different results.

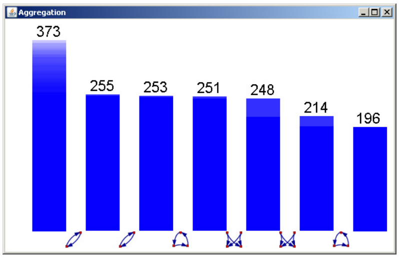

Figure 10: Tracking of aggregation process showing graph size and type of aggregation.

In our system, the aggregated nodes/edges visually completely replace the original nodes and edges. The advantage is that the aggregated graph is simpler and therefore better readable. However, it is not possible to see the original graph structure in the aggregated one directly. For this purpose, we offer the option to show icons in the node center to indicate the motif type used for the aggregation (see Figure 11) and a detailed view on the aggregation hierarchy for a selected node (see Figure 12).

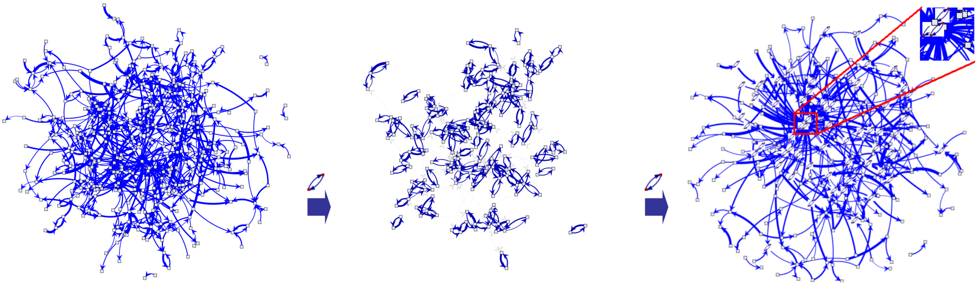

Figure 11: Example of motif-based graph aggregation. Left: Original graph. Center: Reciprocity motifs selected for aggregation. Right: Aggregated graph with node icons indicating type of aggregation.

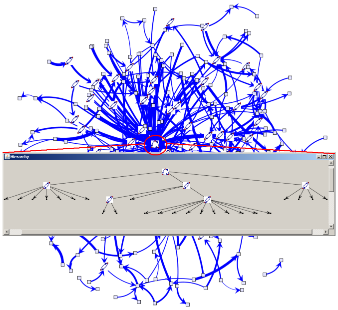

Figure 12: Aggregation hierarchy for a node.

5 Application

We demonstrate our approach using the data set from one of the VAST 2008 contest mini-challenges [18]. The data set contains data on a phone call network (who calls whom, time and call duration). Each node of the network represents a person. An edge between two persons indicates who calls whom and for how long during the day. The data was provided for 10 days in June 2006. In this paper, we look at the communication between people during the second day in the period. The visualization of this network is shown in Figure 2 (left).

In general, the participants of the challenge examined the time-varying network in an undirected case. They concentrated on finding differences in the network over time and identification of important and similar persons in the network. In contrast to these approaches, we visually analyze the weighted directed network with respect to the three tasks: a) occurrence and location of network motifs, b) two what-if-scenarios: analysis of implications of emergence of a new connection between two persons and c) identification of network structures on multiple abstraction levels.

(a.) *Motif analysis:* In phone call networks, appropriately defined motifs show typical ways of flow of information in the network. For example, reciprocity motif shows mutual exchange of information, feedback motif shows circular flow among three people, out-stars identify persons who communicate to many people and in-stars those who are informed by many people. Interestingly, in accordance with the results provided in the challenge, we also could in this way identify the important person number 1 being center of an in-star motif (called by 23 persons). Similarly, nodes 0, 13 and 200 are centers of out-stars, each sending information to at least 5 other persons. The importance of nodes 0 and 1 was confirmed also by their membership the user defined motif (see Figure 3). Moreover, filtering of motifs can help identifying people calling each other for a long time or many times. Figure 4 shows identification of people calling each other and those calling each other longer than 30 minutes during the day. We see that only two couples are called back on the same day and spend long time calling each other (nodes 281 with 137 and 163 with 348). This characteristic is interesting and was not identified by the challenge participants.

(b.) *What-if-scenario:* The what-if-scenarios allow to analyze impacts of changes in the network on its functions. It may be interesting what would happen if a person leaves a network (a node is deleted) or a new connection between two people is created (an edge is added to the network). For example a new connection can create a caro motif, which strengthens flow of information as it can be lead from one person via two people to another person. This case is presented in Figure 5.

(c.) *Aggregation:* Motif-based aggregation reveals higher-level structures in the network. In this example, we merge those people that communicate to each other (reciprocity motif). After merging them we can see how the information flows between such couples (see Figure 11). In this case, a new reciprocity motif emerges. It shows that there is an exchange of information between two of the original couples. Interestingly, other couples do not exchange information directly. Successively, we can aggregate such nodes in order to see the main paths of information in the network on a higher abstraction level. Figure 8 shows results of successive aggregation creating one important merged node in the middle. The out- and inflow of information spreads almost through the whole network (see Figure 13). This means that people inside this structure are a median of information flow from top to bottom and at the same time communicate to each other on multiple levels.

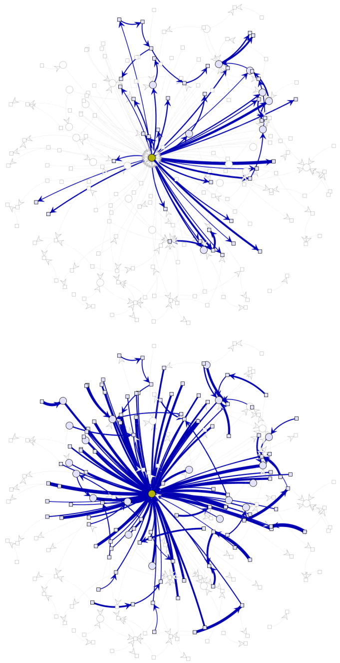

Figure 13: Outgoing and incoming connections of the central aggregated node

6 Conclusion and Future Work

In this paper, we explored how the use of graph motifs can contribute to the visual analysis of large graphs. A novel system for the visual analysis of graphs based on search and interactive visualization of graph substructures was proposed. It also supports the analysis of graph changes (in particular useful for “what-if-analysis”) by showing the implications of data-driven or user-defined graph changes on local structures. Moreover, it offers a new way of graph aggregation based on motifs, which allows the analysis of graph structures and their relations on multiple abstraction levels. In this system, the predefined graph motifs can be interactively extended by user proposed substructures, thereby offering the possibility to discover and search for new patterns. Based on interactive combinations of these functions, the visual analytics system supports a wide range of analytic tasks. We have shown efficiency of the motif-based approaches for directed graphs, however, these techniques can be easily adopted on other graph types.

The system can be used in various applications including social networks, references in document collections, transportation networks. While this system makes a first step toward interactive motif- and change-based visual graph analysis, interesting problems to be solved exist on the side of motif-based graph analysis, aggregation, and visualization. In particular we will address the computational complexity of identification of general motifs in graphs of arbitrary size, visualization and aggregation of overlapping motifs and visualization of multi-level aggregated graph.

Acknowledgements

The authors thank Jerome Kirchhoff, Ralf Guttbel and Jordan Kavaldjiev for their support in system development. This work was partially supported by the German Federal Ministry of Economics and Technology within the THESEUS project (`http://www.theseus-programm.de/`). It was also partially supported by the German Research Foundation (DFG) within the project Visual Feature Space Analysis as part of the Priority Program Scalable Visual Analytics (SPP 1335).

References

[1] A. T. Adai, S. V. Date, S. Wieland, and E. M. Marcotte. Lgl: creating a map of protein function with an algorithm for visualizing very large biological networks. *Journal of Molecular Biology*, 340(1):179–190, June 2004.

[2] D. Archambault, T. Munzner, and D. Auber. Grouseflocks: Steerable exploration of graph hierarchy space. *IEEE Transactions on Visualization and Computer Graphics*, 14(4):900–913, July 2008.

[3] Daniel Archambault, David Auber, and Tamara Munzner. Topolayout: Multilevel graph layout by topological features. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):305–317, 2007.

[4] Daniel Archambault, Tamara Munzner, and David Auber. Tuggraph: Path-preserving hierarchies for browsing proximity and paths in graphs. In *Proceedings IEEE VGTC Pacific Visualization Symposium*, pages 113–120, April 2009.

[5] D. Auber, Y. Chiricota, F. Jourdan, and G. Melancon. Multiscale visualization of small world networks. *IEEE Symposium on Information Visualization*, pages 75–81, 2003.

[6] Pawel Gajer and Stephen G. Kobourov. Grip: Graph drawing with intelligent placement. In *GD '00: Proceedings of the 8th International Symposium on Graph Drawing*, pages 222–228, London, UK, 2001. Springer-Verlag.

[7] Joshua Grochow and Manolis Kellis. Network motif discovery using subgraph enumeration and symmetry-breaking. *Research in Computational Molecular Biology*, pages 92–106, 2007.

[8] Paul Holleis, Thomas Zimmermann, and Daniel Gmach. Drawing graphs within graphs. *Journal of Graph Algorithms and Applications*, 9(1):7–18, Oct. 2005.

[9] Weidong Huang, C. Xiaobin Shen Murray, Le Song, Ying Xin Wu, and Lanbo Zheng. Visualisation and analysis of network motifs. In *Proceedings. Ninth International Conference on Information Visualisation, 2005.*, pages 697– 702, July 2005.

[10] Christian Klukas, Falk Schreiber, and Henning Schwöbbermeyer. Coordinated perspectives and enhanced force-directed layout for the analysis of network motifs. In *Proceedings of Asia-Pacific Symposium on Information Visualisation (APVis 2006)*, pages 39–48, 2006.

[11] G. Kumar and M. Garland. Visual exploration of complex time-varying graphs. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):805–812, Sept.-Oct. 2006.

[12] Jure Leskovec and Christos Faloutsos. Sampling from large graphs. In *Proc. of the 12th ACM SIGKDD Int. Conference on Knowledge Discovery and data mining*, pages 631–636, 2006.

[13] Avi Ma'ayan, Sherry Jenkins, Ryan Webb, Seth Berger, Sudarshan Purushothaman, Noura A. Husn, Jeremy Posner, Tony Flores, and Ravi Iyengar. Snavi: Desktop application for analysis and visualization of large-scale signaling networks. *BMC Systems Biology*, 3(1), 2009.

[14] MFinder and MDraw. `http://www.weizmann.ac.il/mcb/UriAlon/groupNetworkMotifSW.html` (accessed 5.6.2009).

[15] Joshua O'Madadhain, Danyel Fisher, and Scott White. Java universal graph framework (JUNG) `http://jung.sourceforge.net/`.

[16] Falk Schreiber and Henning Schwöbbermeyer. Mavisto: a tool for the exploration of network motifs. *Bioinformatics*, 21(17):3572–3574, July 2005.

[17] Henning Schwbbermeyer. *Analysis of Biological Networks*, chapter 5, pages 85 – 112. Wiley Series on Bioinformatics, Computational Techniques and Engineering. Wiley, 2008.

[18] VAST Contest. `http://www.cs.umd.edu/hcil/VASTchallenge08/` (accessed 19.6.2009).

[19] S. Wernicke and F. Rasche. Fanmod: a tool for fast network motif detection. *Bioinformatics*, 22(9):11521153, 2006.

[20] Sebastian Wernicke. Efficient detection of network motifs. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 3(4):347–359, 2006.