Info Vis

Article

Information Visualization
2020, Vol. 19(2) 96–113

DOI: 10.1177/1473871619882019
journals.sagepub.com/home/ivi
SAGE

State-based network similarity visualization

Sugeerth Murugesan[1,2], Kristofer Bouchard[2], Jesse Brown[3], Mariam Kiran[2], Dan Lurie[4], Bernd Hamann[1] and Gunther H Weber[1,2]

Abstract
We introduce an approach for the interactive visual analysis of weighted, dynamic networks. These networks arise in areas such as computational neuroscience, sociology, and biology. Network analysis remains challenging due to complex time-varying network behavior. For example, edges disappear/reappear, communities grow/vanish, or overall network topology changes. Our technique, TimeSum, detects the important topological changes in graph data to abstract the dynamic network and visualize one summary representation for each temporal phase, a state. We define a network state as a graph with similar topology over a specific time interval. To enable a holistic comparison of networks, we use a difference network to depict edge and community changes. We present case studies to demonstrate that our methods are effective and useful for extracting and exploring complex dynamic behavior of networks.

Keywords
Dynamic networks, visual analytics, cluster analysis, data visualization, time-series analysis

Introduction

Representations and analysis of networks are important to investigate the interactions between entities in a variety of applications. For example, to investigate teacher–student and student–student interaction patterns in a classroom,[1] social scientists use networks to model behavioral patterns. In neuroscience, understanding network properties, with nodes representing brain regions and edges representing functional dependencies, is important to gain insight into the cognitive functions of the brain. Visualization of connectivity in such networks is crucial to extract insights concerning global topology, local neighborhood patterns, and overall community structure.

Networks are often large and time dependent, evolving over several hundreds of timesteps. Network changes are reflected through the addition or deletion of edges, the birth or death of communities, or fluctuations in modularity. In most cases, such dynamic changes do not affect all the structures in a network, and many core structures remain stable over certain periods of time. We call such phases with similar topological structure as states of a dynamic network. If we can detect and understand these states and core structures that remain stable over states, we will then be able to answer questions related to their role in the context of an entire system's evolution. For example, neuroscientists are keenly interested in such states to comprehend interactions that relate cognitive functions to the functional states; social scientists, studying classroom interactions, are concerned about the

[1]Department of Computer Science, UC Davis, Davis, CA, USA
[2]Lawrence Berkeley National Laboratory, Berkeley, CA, USA
[3]University of California—San Francisco, San Francisco, CA, USA
[4]University of California—Berkeley, Berkeley, CA, USA

Corresponding author:
Sugeerth Murugesan, Department of Computer Science, UC Davis, One Shields Avenue, Davis, CA 95616, USA.
Email: smuru@ucdavis.edu

correlation of communication patterns with their educational outcomes.

The complex dynamic behavior (within and between states) of a network affects its overall topology. Identifying the dependencies between these properties over time can provide us with high-level information about the causes and effects of evolution behavior. In other words, a minor change, such as the addition or deletion of an edge, might explain an entire global state change of a dynamic network. For example, in a classroom social network, the removal of a communication channel between two popular students (who are linked to many other students) could explain a major shift in social dynamics of the classroom.

Analyzing and depicting such data in a comprehensible manner for large, dense networks remains challenging, as purely visual approaches become inadequate and require algorithmic techniques to meaningfully present and abstract the data without actual information loss. Many existing approaches that visualize dynamic networks and their community structures depict an entire network and all its changes for every timestep. Such techniques can lead to duplication of visual elements that represent nearly the same information (due to temporal correlation). Furthermore, they require large screen space for visualization. For example, an electrocorticography (ECoG) data array capturing neuronal activation patterns of brain regions for just 1 h may need several thousand copies of the original network, resulting in a representation with several million nodes and edges. Moreover, most current methods do not effectively show changes in topology between timesteps or their effects on the overall system. Analyzing these changes are essential for understanding the time-varying phenomena in dynamic social networks or neurodegenerative diseases, for example, schizophrenia.

These challenges motivated us to devise a new approach for detecting, visually representing, and exploring summaries and differences concerning topological changes, including the use of depictions of summary structures, that is, states, see Figure 1(c). We summarize and represent underlying dynamic behavior for each detected state by computing a representative summary graph using unique glyph designs, see Figure 1(b). To represent and explore the topological changes between networks over states, we have developed a network difference technique, see Figure 1(d). Our main contributions are as follows

- A new approach for the visualization of dynamic networks through detecting temporal states.
- Effective visual graph designs emphasizing similarity and differences in topology over time.

Related work

We review related work in three areas: dynamic network visualization, dynamic network simplification, and the difference-graph framework.

Dynamic network visualization

Static network visualization techniques[2] often use two major views, that is, matrices or graphs (when a depiction via a graph is possible), to understand the global topology of the system. In dynamic graph representations, the major distinguishing feature is the temporal aspect. Key challenges in visualizing such graphs concern visual scalability of graphical primitives and computational complexity of graph processing. Various visualization techniques have been proposed to communicate changes effectively. The survey by Beck and colleagues[3,4] categorizes most existing work in dynamic network visualization into two main methodologies, that is, animations and small-multiples.

Through animation, major evolving patterns, such as community changes, are shown by interpolating smooth transitions between the underlying layout or partitioning the rendering space into hierarchically arranged blocks.[5] Techniques proposed by Ghani et al.[6] explore various metrics for enhancing user perception of the animation. The space to time mapping approach[7] draws a sequence of graphs along a timeline. Unlike animations, the space to time mapping technique enables easy comparison between objects at discontinuous timesteps. However, for a large number of timesteps, such views can be cognitively overwhelming.

Other techniques include parallel edge splatting,[8] where all the changes in edges between graphs across timesteps are visualized to identify general trends in the data set. Furthermore, alluvial diagrams model the links between clusters in successive timesteps as split-merge ribbons[7,9,10] to enhance visual traceability of important cluster evolution patterns. Techniques like EgoNetCloud[11] use an egocentric summarization approach to analyze major events in a data set, while GraphDiaries[12] use animated transitions between timesteps in a network to highlight changes. Further work by van den Elzen et al.[13] involves reducing the time-varying network into interpretable timepoints, giving the user an intuition to detect states within the data set. Another work by Dal Col et al.[14] further detect connectivity patterns on large time networks using wavelet transform methodology.

In contrast, the approach introduced in our work, uses a simplification algorithm to deal with networks that evolve over a large number of timesteps and utilizes summary and difference representations to reveal

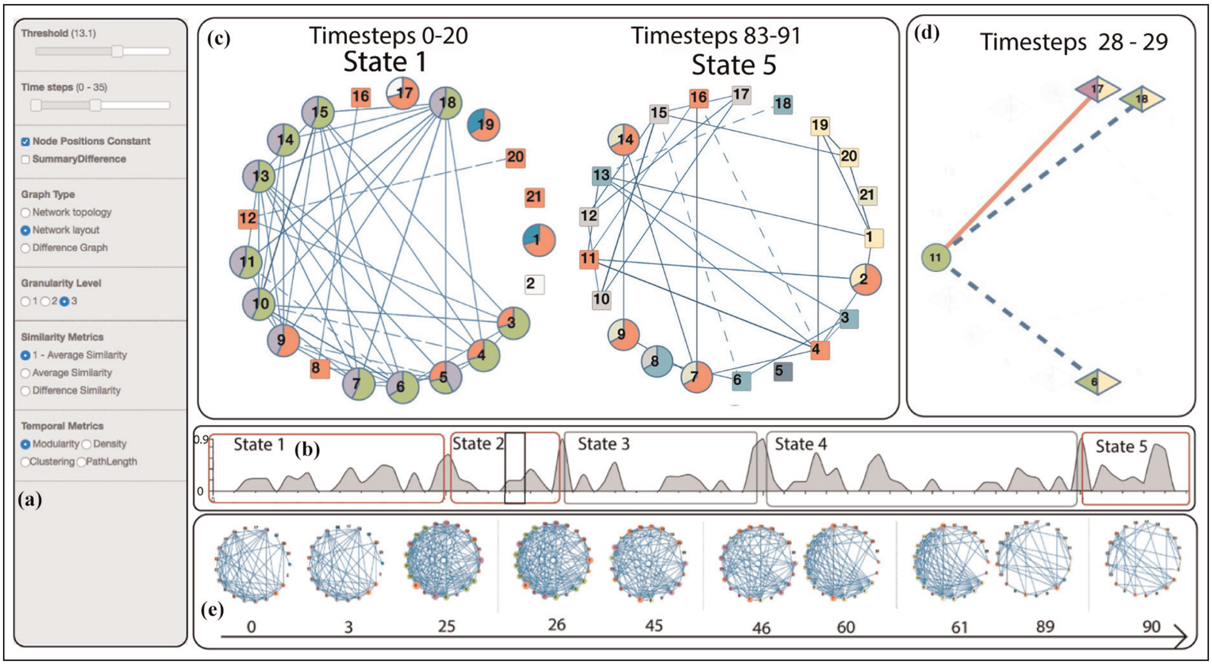

Figure 1. Our approach enables a holistic understanding of time-varying networks through two complementary techniques, Summary (c) and Difference Graphs (d). (a) Control menu for selecting plot styles and visualization parameter values. (b) Similarity plot showing magnitude of topological change between two graphs at adjacent timepoints. Each of the five states is defined by a time interval corresponding to a stable topological structure. The brain network possesses major topological changes through timesteps, 22, 32, 53, and 82. (c) Visualization of topology in the summary form. States 1 and 2 are visualized for further analysis. Pie Glyphs represent nodes that change community membership for each state, while squares represent stable, unchanging communities. (d) Difference Graph of network states—Diamonds represent nodes that change community membership across time. Disappearance of an edge (from brain region 11 to 18 and to 6) caused the community to change, indicated by the change from green to yellow from nodes 18 and 6. (e) Visualization of original network with a traditional small-multiples method.

variabilities and complex dynamics underlying dynamic networks.

Dynamic network simplification

One of the important problems in visualizing large-scale dynamic networks is depicting the connectivity information without substantial information loss.[15] Several techniques have been published that deal with the visual complexity problem. These approaches rely on manual filtering of edges and nodes or deriving a minimum spanning tree, when possible, preserving connectivity. The path-oriented simplification[16] method removes edges that do not affect the quality of best paths between any pair of nodes. Approaches like Structural Equivalence Grouping (SEG),[17] Apostolico,[18] and EgoNetCloud[11] condense large network information without sacrificing connectivity information. Other approaches compress weighted graphs[19] or use motif-based methods for static graphs.[20] Methods presented by Sun et al.[21] and Eagle and Pentland[22] use minimum description length and Fourier transform analysis, respectively, to compress complex networks respectively.

In several applications, it is important to detect states in dynamic networks, see for example, Rashid et al.[23] and Mutlu et al.[24] In contrast, the abstraction procedure in our method happens at the level of the network, preserving the rich topological structure of the individual state. Furthermore, we compute a summary and difference representation to explore the complex dynamics underlying each state. Such methods are essential for an in-depth comprehension of the overall role of states in the context of system dynamics.

Difference-graph framework

A difference graph describes the changes between the graphs of two timesteps. Given two graphs, only the changes concerning edges and nodes are visualized.[25] To handle large-scale changes in difference graphs, Archambault[26] used hierarchies to show where large areas of a graph change. Bourqui and Jourdan[27] described a method that visualizes edges having similar pathways, focusing on structural similarity. Rufiange and McGuffin[28] used a hybrid method involving small-multiples and animations to determine local topological changes between graphs.

Our method focuses on changes in edges (constant nodes) and community membership. We characterize changes by defining the importance of each change in a graph and visualizing the effect of the change on the topology using a community-based difference graph.

Design goals

Based on existing literature regarding dynamic network visualization,[29] we identified primary design goals to be met by our approach. We base our approach on the following goals.

Simplification of temporal features

Topological patterns in dynamic networks often reoccur, for example, brain network patterns. Methods are needed that can aggregate data by considering and taking advantage of similarity between graphs.

Support of effective visualization of topological shifts

Visualizing series of large, dense graphs causes visual information overload hindering data interpretation. Due to the limited cognitive processing ability of humans, techniques should detect and convey changes in a visually comprehensible manner.

Interactive exploration capability for temporal data

Networks function at multiple scales. To efficiently derive insights at different scales, the methods must allow a user to explore summary and detail information on demand.

Motivated by major network analysis tasks covered by Alper et al.,[30] for example, we identified the following important capabilities that our method should address:

- Identify and characterize patterns defining temporal states (T1).
- Analyze summary topologies that represent temporal states (T2).
- Analyze topological variations within states (T3).
- Analyze local dynamics governing global community and state changes (T4).

Method

Our computational framework incorporates algorithmic analysis with interactive visualization to extract and summarize recurring patterns in graphs. As shown in Figure 1, in the first stage, we use a similarity metric to identify time intervals within the dynamic network that possess similar network topology (Figure 1(b)); in the second stage, using the detected interval points, we run our summary graph representation algorithm to compute the most common topology representing the detected state, see Figure 1(c); in the third stage, to allow a user to explore topological changes across graphs, we compute the importance of each edge change, to construct and visualize a difference graph using novel visual designs, see Figure 1(d). Throughout all the stages, we enable interactive filtering, aggregation, and selection allowing users to interactively explore major recurring patterns within the graph.

To reliably generate summary graphs, our technique requires an understanding of temporal change within the graph. To numerically quantify such a change, we define a similarity measure. This measure allows us to detect states (T1) having similar graph-level properties. A well-constructed similarity metric allows us to detect states (T1) having similar graph-level properties.

Notation and definitions

Mathematically, we model a dynamic graph G_s as a sequence of static graphs, denoted as $G_s = \{G_0, G_1, G_2, \ldots\}$. We detect communities for each timestep, where a community consists of a subset of nodes within a particular timestep. All communities between subsequent timesteps are matched using the maximum overlap algorithm.[31] We do not employ temporal smoothing for communities as we assume that the underlying connectivity is temporally correlated. Most naturally occurring time-series show significant auto-correlation.

We assume that the number of nodes is the same for every timestep and that edges are undirected and weighted. Three major quantities for a data set include, that is, time, number of nodes, and number of edges. We denote the temporal state set as $S_t = \{S_1, S_2, S_3, \ldots\}$, where each state S_i contains a range of continuous timepoints within the timepoints in the data set.

State-based similarity measure

In order to detect topological change between graphs and group them into states, we establish a means for identifying similarities and changes within graphs. While general graph properties, such as degree distribution and community membership, can be used to characterize change in networks, they are too generic to extract complex time-varying behavior in the graph. We have adopted the similarity metric described by Koutra et al.[32] for similarity of network topology. It is an accepted similarity measure for quantifying changes in graphs, and, through discussions with domain experts, we determined that this method is applicable to our use cases. Based on the demands of a specific application, we can include different measures in our visual analysis system to reliably extract temporal states that possess the most meaning.

We compute a similarity value $Sim_i(G_k, G_{k+1}) \in [0, 1]$, where a value of 1 implies that two graphs G_k, G_{k+1} are similar, and a value of 0 indicates that two graphs are maximally dissimilar.[32] To compute such a value, we need to define the influence that any node i has on all other nodes j, for all nodes within the graph. To perform such an operation, we define N (N being the number of nodes in a graph) column vectors $\vec{s}_i$ for every node i and arrange them in a matrix S, with $\vec{s}_i$ being a column in S. Intuitively, for Koutra et al.[32] measure, influence scores, $s_{i,j}$, between nodes i and j are higher when sum paths of the edge weights are larger and are at most one hop away. The vector matrix S, encompassing such score is defined as

$$S = [s_{ij}] = [I + \varepsilon^2 D - \varepsilon A]^{-1} \tag{1}$$

Here, $\varepsilon = 1/(1 + max(d_{ii}))$ is a value used to capture the influence between neighboring nodes, and D is an $n \times n$ diagonal matrix, where d_{ii} is the degree of node i. A is the adjacency matrix of the raw graph data for each timestep, and I is the identity matrix. In order to compute a distance between graph vectors, we use the root-mean-square (RMS) measure, allowing us to detect changes in graphs. Formally

$$d = RootED(S_1, S_2) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\sqrt{S_{1,ij}} - \sqrt{S_{2,ij}}\right)^2} \tag{2}$$

and the eventual distance value of two graphs is defined as

$$sim(G_1, G_2) = \frac{1}{1 + d} \tag{3}$$

States defined by similarity matrix

To extract temporal states from the metric, a method is needed to compute the time intervals belonging to each state. Such a method must take into account the similarity information for all pairs of timesteps, taking into account reoccurring patterns across discontinuous timesteps.

We define a distance matrix of size $N \times N$, where N is the number of timesteps, and matrix entry $c_{i,j}$ is the similarity (equation (3)) of the graph structures at timesteps i and j. The higher the value, the more similar the graphs i and j. As we are mainly interested in the most common topological properties associated with the state, we use a threshold that is large enough not to "swamp" the matrix with moderately similar entries—but not too large in order to capture significant entries. We employ a modified change detection mechanism proposed by Mutlu et al.[24]

$$I(G_i, G_j) = \begin{cases} Sim_{i,j}, \text{if } (Sim_{i,j} - (u_G + \delta_1 * \sigma_G) \geqslant 0) \\ 0, \text{ otherwise} \end{cases} \tag{4}$$

where u_G and σ_G are the mean and standard deviation of the values in the distance matrix and δ_1 is the threshold coefficient for σ_G. We assume that equation (4) relies on a normal distribution of the similarity values without a large number of outliers. Based on the thresholded matrix, we use a connectivity-based clustering algorithm[33] to identify clusters with similar graph structure to define states. An advantage of this approach is the fact that the states obtained are based on values of the entire time data set, rather than the values based on a current timestep t and previous timestep $t + 1$.

Algorithm 1: Summary graph representation

Data: Sequence of similar graphs G_1,G_2,G_3,...
Result: Representative summary graph G_{Av}

1. Compute adjacency matrix of pairwise distance values for time-steps t_k to t_{k+w}, using Eq. 3, where w is the time-interval;
2. Perform standard change detection procedure, using Eq. 4, to threshold the matrix;
3. Rank all graphs G_{t_i} based on number and average of non-zero values in column i, $avgSim_i$;
4. Filter top k candidates for comparison based on a threshold δ_2 for filtering the similarity matrix;
5. Pick most frequent community membership for each node in G_{Av} from k graphs—in case of a tie, use membership from the graph with highest $avgSim_i$ value;
6. Intersect edge lists of the k graphs to obtain edge list of G_{Av};
7. Set weights of edge lists of G_{Av}, averaging over all k graphs—ignore non-existing edges;

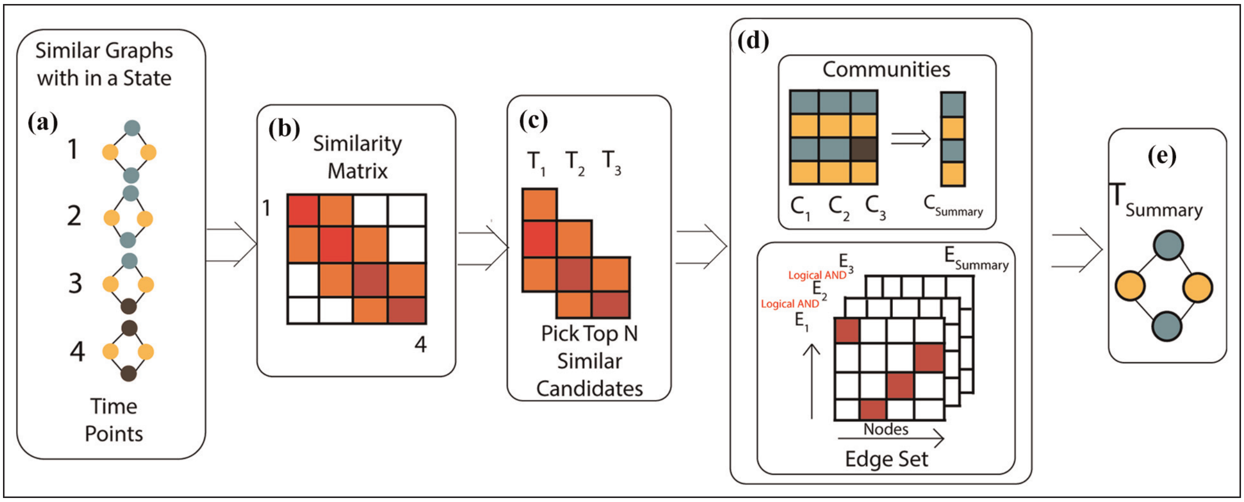

Figure 2. The algorithm used to construct the summary graph identifies the most common local and global topological attributes from a set of similar graphs.
(a) Networks for a given time interval define the input. (b) Similarities between networks are computed which define the entries of a similarity matrix. (c) Using a threshold δ_2, we pick the top k candidates for intersection of edge lists and computation of (d) summary community memberships, characterizing the (e) summary graph.

Consensus-based state summary graphs

To understand dynamic network topology reflecting state changes and reduce the amount of information used for visual inspection, we derive a single representation for each state. The reduced graph shows overall stability (community changes and edge changes) of the network and variability and mean of important changes, for the time interval of a detected state.

Our temporally reduced summary graph representation should satisfy the following design objectives:

- The graph representation must capture the most common topological properties of the system.
- The graph representation must be indicative of the overall community structure for the interval.
- The representation must highlight nodes that often change communities.

We consider different approaches to generate a temporally reduced summary graph in order to satisfy the design criteria. One can accumulate the nodes over time to form edges. However, one of the assumptions is that the meaning of edges is ignored. The representation of such graphs depends on the lengths of the time intervals considered. The larger the time-windows, the higher the “density” of the graph.

An alternative approach identifies changes from a pool of similar graphs and averages the weights over the detected state-intervals. This approach treats all edges equally and produces a dense network representation, which may not be representative of the most common topology.

Our approach

Our approach for defining a summary graph, see Figure 2, produces informative summaries based on salient patterns. We construct one representative visualization of the state. Our problem definition can be stated a follows: given N similar graph structures, how should one depict the most representative graph? We consider these main aspects for the representative graph:

- Identification of the community membership of a node in the graph.
- Determination of the presence of edges between all pairs of nodes.
- Determination of the weight of an edge.

Algorithm 1 and Figure 2 describe the procedure that constructs a summary graph from a set of similar networks for a given temporal state. To determine the quality and uncertainties associated with a graph generated by the summary algorithm, we use a metric, Average Number of Edges Pruned per Timestep (AEP), defined as total number of edges pruned by algorithm per state between same nodes/time interval of the state. This estimates the average amount of dynamic edges (edges that frequently get added or deleted within the state interval pruned by the algorithm). This value provides us with insight into the overall variability of the topologies within the state. For example, states with edges exhibiting significant variability (addition or deletion) have relatively high AEP values. Such a metric allows us to determine whether the sparseness of the graph is representative of the underlying data or is the result of the dynamics occurring within the state.

Visual representation of summary graph

The reduced graphs are a simple representation of a complex dynamic phenomena occurring within the represented time interval.

To facilitate understanding such data, we need visual representations that not only depict the summarized topological information but also depict variances in the properties where there might be one. Such depiction of dynamics is crucial to understanding the cause and effect of the formation of the temporal states. For example, in neuroscience, transient nodes (brain regions) with more flexibility (frequent community memberships changes) may be involved in performing a wide range of cognitive functions and may be particularly involved for changing global brain states.[34]

To facilitate such a detailed exploration process in an intuitive manner, we want to represent the following quantities: (1) nodes with frequent community changes, (2) stability of communities, (3) variability in edge weights, and (4) mean value of edge weights.

We considered three visual designs for this purpose, all following the data aggregation principles discussed by Elmqvist and Fekete[35] (Figure 4). In the first design, to enable comparison between edges in the same graph, we encode variance in edge weights using color, and line thickness represents mean edge weight. Outlier edges with high variances are shown as dashed lines. Transient nodes that change their community membership over time are represented as vertically stacked bars with an enclosed circle. The length of a bar represents longevity of a community in that node for a given time interval. The second and third designs use the same scheme for the edges; however, each node is represented as a pie chart, where each slice represents the longevity of a community in a particular node. In the third design, based on the principle of selective visual attention as described by Lavie and Cox[36] and to visually distinguish dynamic behavior of transient nodes, we represent other stable nodes as static rounded rectangles, with color representing community membership.

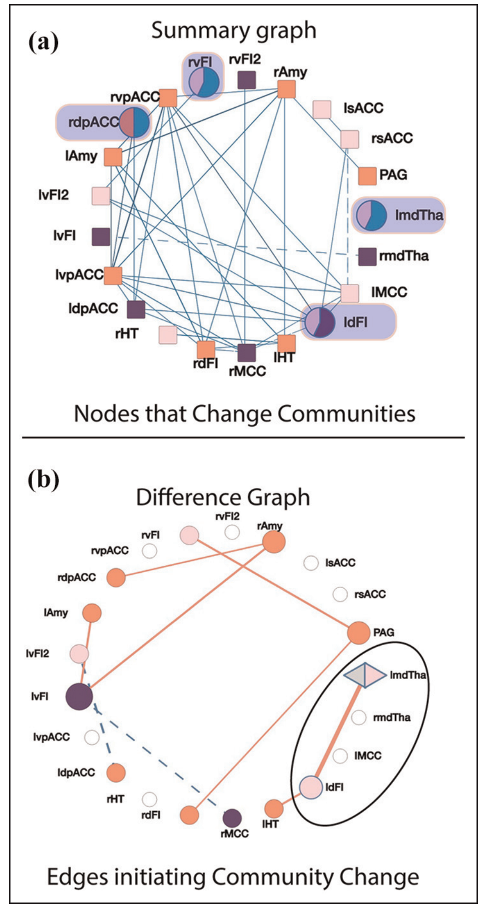

Figure 3. Our methods applied to a real brain network data set. The summary graph in conjunction with the difference graph depict similarities and differences in topologies, respectively. (a) Summary graph involving four nodes (glyphs) depict the frequent change in communities during a particular time interval. (b) A difference graph explaining the community change of ldFI, left dorsal frontoinsula and lmdTha, mediodorsal thalamus, from gray to pink.

Overall, based on different trial runs, we found the third design to express the changes in a visually clear and concise manner. Particularly, the unique visual outlines convey the changes in transient nodes more effectively. We use this as our design for the summary graph (Figure 3).

Difference graph framework

Analyzing every graph at each timestep and identifying change is cognitively overwhelming. The problem becomes more pronounced for large graphs with minuscule change occurring between timesteps. The analyst has to manually correlate changes from many views, increasing context-switching costs associated with every view.

Given two graphs, to enable assessment of the functional difference between them, it is crucial that we not only show the change but also show the importance that change has on its overall topology. For example, the disappearance of an edge causing a community change is far more important than the disappearance of an edge within a community and causing no change in the global structure.

To address this issue, we propose a community-based difference-graph approach to depict the dynamics occurring over a set of similar graphs for a given state.

Community-based difference graph visual design

Since we do not deal with addition or deletion of nodes, we define the nodes of the difference graph to have the same meaning as the static graph. For edges, we define three major types (in the order of decreasing importance) (Figure 4(c)).

1. Modular edge change—defined as an event involving addition or deletion of an edge when comparing adjacent timepoints A and B. Such type of a change always results in change of communities, for example, at timepoint B. Figure 4(c-1).
2. Inter-modular edge change—defined as addition or deletion of edges between graphs across two timepoints, unlike modular edge changes, such edges connect between different communities and transfer information over multiple communities, Figure 4(c-2).
3. Intra-modular edge change—such change events occur across two timepoints, causing no overall changes in community, however decreasing/increasing the overall connectivity within or across a community, Figure 4(c-3).

We use thickness of the edge line to convey the importance of these edges. The thicker the edge, the greater is its importance. Based on previous studies in a difference graphs,[12,37,38] we use the colors red and blue to indicate the addition and the deletion of edges respectively. There are cases when a node in a graph is becoming transient, that is, changing the community membership. Such events are the pivotal timepoints of nodes and are of utmost importance in its evolution.

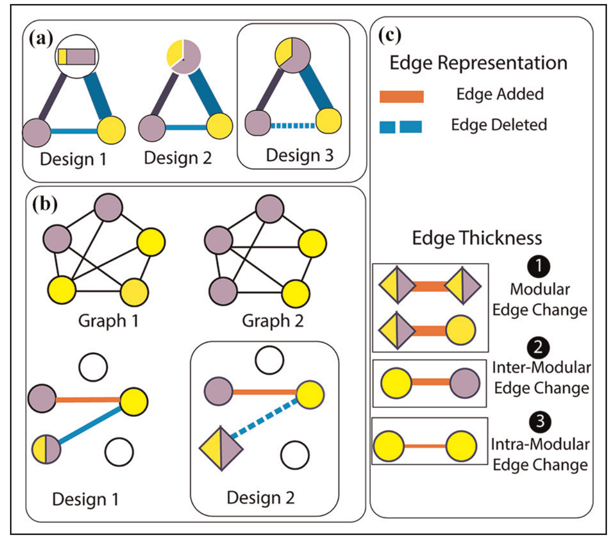

Figure 4. We investigate different visual designs to convey uncertainty and variability in the dynamic network data. (a) Summary graph representing stable nodes as rectangles and nodes that change their communities as glyphs. (b) Difference graphs representing deletion and addition of edges as dashed blue and solid red lines, respectively. (c) Visual encodings that depict the importance of each edge change in the difference graph.

We explored two designs to convey such dynamics occurring in difference graph.

In the first design (Figure 4(b)), to depict a community change at a given node, we subdivide the circle into two equal parts, the left semicircle denoting the community membership of the previous timestep and the right one denoting the membership at the current timestep. In our second design to further visually distinguish the most important change, we encode dashed lines to depict deletion of edges and encode a diamond with two sides to depict the change in community membership for a given node. To enhance visual clarity, we only show the nodes that are involved in change. The size of the nodes in our graph represents the amount of local change).

Different trials of various visual designs based on a real brain network data set made it clear that the second design showed variations in data more convincingly. However, in cases where there is a lot of change occurring, the user would have to analyze the original raw graph to assess the change in the difference graph. There is a switching cost involved in such operations. Nonetheless, we found that this technique is better suited for our use-case. Furthermore, we utilize various interaction techniques such as egocentric navigation

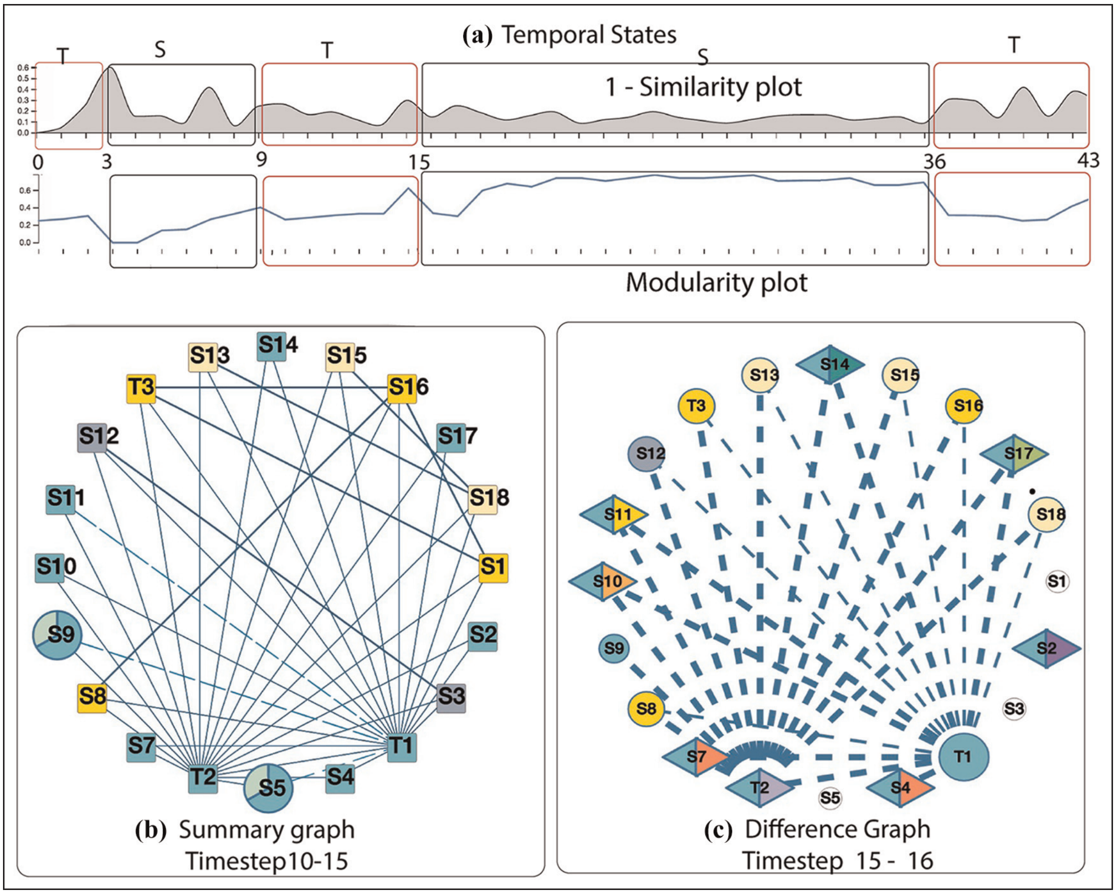

Figure 5. The two dominant states in the McFarland data set. (a) Red rectangles represent lecture sessions by the teachers, and black rectangles represent states when the teachers are not lecturing. (b) Broadcasting edges from teachers T1 and T2 depict lecturing student groups. (c) State change as a consequence of students forming groups and teachers not lecturing.

(Figure 8), filtering (Figure 7), and focus + context exploration (Figure 5) to steer through the complexity.

Case studies

We now apply our methods to generic domains to test the generalizability of our methods. We note that domain-specific studies require similarity measures specific for the study at hand. For our case, as the similarity has been proven generalizable for diverse data sets, we apply our methods to a network flow and social science data sets.

McFarland's classroom data set

We have applied our visual analysis technique to the publicly available "McFarland's classroom" data set.[1,39] This data set contains information concerning conversations between teachers and students in a high-school economics class (11th and 12th grades). We used the McFarland data set as a use-case for our system to analyze how students and teachers interacted in this class. We pre-processed the given data to create a dynamic network data set, appropriate as input to our system, consisting of 20 nodes (17 students and 3 teachers) spanning 49 min (converted to 82 timesteps). We were able to identify evolving communities, states, and a summary topology for each state.

The state-detection algorithm found two dominant conversation states in the data set, that is, a first state where teachers did broadcast information to students and a second "sociable state" where students interacted with each other, forming multiple communities (modular) (T1 and T2). The detected intervals in the states have the following intervals, state one has associated intervals (1-2, 10-15, 37-42, and 49), and state two has associated intervals (3-9, 16-36, 42-48, and

50-82), see Figure 5(a). The states identified by our system are in agreement with those reported in a study concerning this data set.[1]

Specifically, the class started with the teacher lecturing (state one, timesteps 1–2), followed by students performing a group task (state two, timesteps 3–9). The students in this state formed closely knit communities talking with each other over a few known social groups. This state's structure is dispersed as a consequence of teacher intervention, defining the structure for timesteps 10–15. The fluctuating state patterns with consistent sparse and dense core connectivity are captured by the summary graph, see Figure 5(b).

To understand the dynamics of state one, we selected a sub-interval (10–15, within state one) and visualized its corresponding summary and difference graphs, see Figure 5(a) and (b). The broadcasting edges emanating from teachers T1 and T2 represent communication to all students over the entire interval. During this interval, a community (light yellow) involving student groups (S13, S15, and S18) can be detected, possibly reflecting conversations among these students during lecture (T3). The circles depicting students S5 and S9 indicate that they potentially have formed their own communities before listening to the lectures of teachers T1 and T2. The summary graph visualizes the consistent dense temporal behavior in this interval.

The difference graph depicting topological change (15–16), see Figure 5(c), contains many dashed lines, that is, edges disappear, indicating that teachers have stopped lecturing and assigned group tasks to students.[1] The relatively large number of rotated diamonds (seven) in the graph could be indicative of student groups (S4 and S7) and (S8, S11, T3, and S16) switching communities, which could have caused a system state change (T3 and T4).

In conclusion, the summary graphs effectively visualize consistent similar network structure over time, where either teachers teach or students perform group tasks, see Figure 5(b). The difference graphs reflect the topological effects teachers' pausing their lectures, when students form their own communities.

Primary school data set

We also tested our method by applying it to an academic collaboration network, the Primary School data set.[40] The goal was to extract contact patterns of students in a primary school. The data represent interactions between 232 school children and 10 teachers for five classes. The data were collected for period from 10:00 am to 12:00 am, Thursday, 1 October 2009. Radio frequency identification (RFID) readers were placed on the contacts to record conversations had in the cafeteria, on stairways, on playgrounds or in classrooms. It is poorly understood how children interact with each other during different times of the day in a school. A better understanding of children's interactions is desirable from a pedagogical viewpoint. We have used our system to better understand this school scenario.

To handle the size of this data set, we used force-directed edge bundling, see Holten and Van Wijk,[41] and placed nodes onto circles for each class. We have explored the following issues:

- How do contact patterns evolve over time (within the timeframe recorded)? Are they tightly knit or modular? (T1)
- What are the communication patterns of students during certain temporal states, for example, during a break? (T2)
- How different are the communication patterns for classroom time versus break time? (T3)
- What student communication patterns cause a state change? (T4)

Figure 6(a) shows the different states detected by our system. The summary and difference networks were used to explore connectivity patterns underlying the data set for the period from 10:00 am to 12:00 pm. Our tool helped with the identification of four states with different connectivity structure.

Based on the temporal states, we explored and identified major shifts in student contact patterns. For example, considering the (1-Similarity) plots shown in Figure 6(a), one can identify timepoints indicative of students moving to different rooms, and explore how that movement affected communication. Timepoints 10:18 am and 10:50 am reflect major shifts in the topology of the graph. Furthermore, based on the connectivity data, the second state corresponds to student interactions on the playground. The third state corresponds to students going back to the classroom.

By examining topology when students were on the playground, from 10:18 am to 10:50 am, we can see that students started forming communities with their grade peers, particularly third graders. This configuration reflects how students interacted during breaks. As annotated in the graph of third graders, some students conversed with specific communities within first graders, light-green and light-red communities. We also see that fifth graders talked less with students in other grades. Second graders talked intensely, compared to others, and formed diverse and rich communities among themselves, see the large number of colored

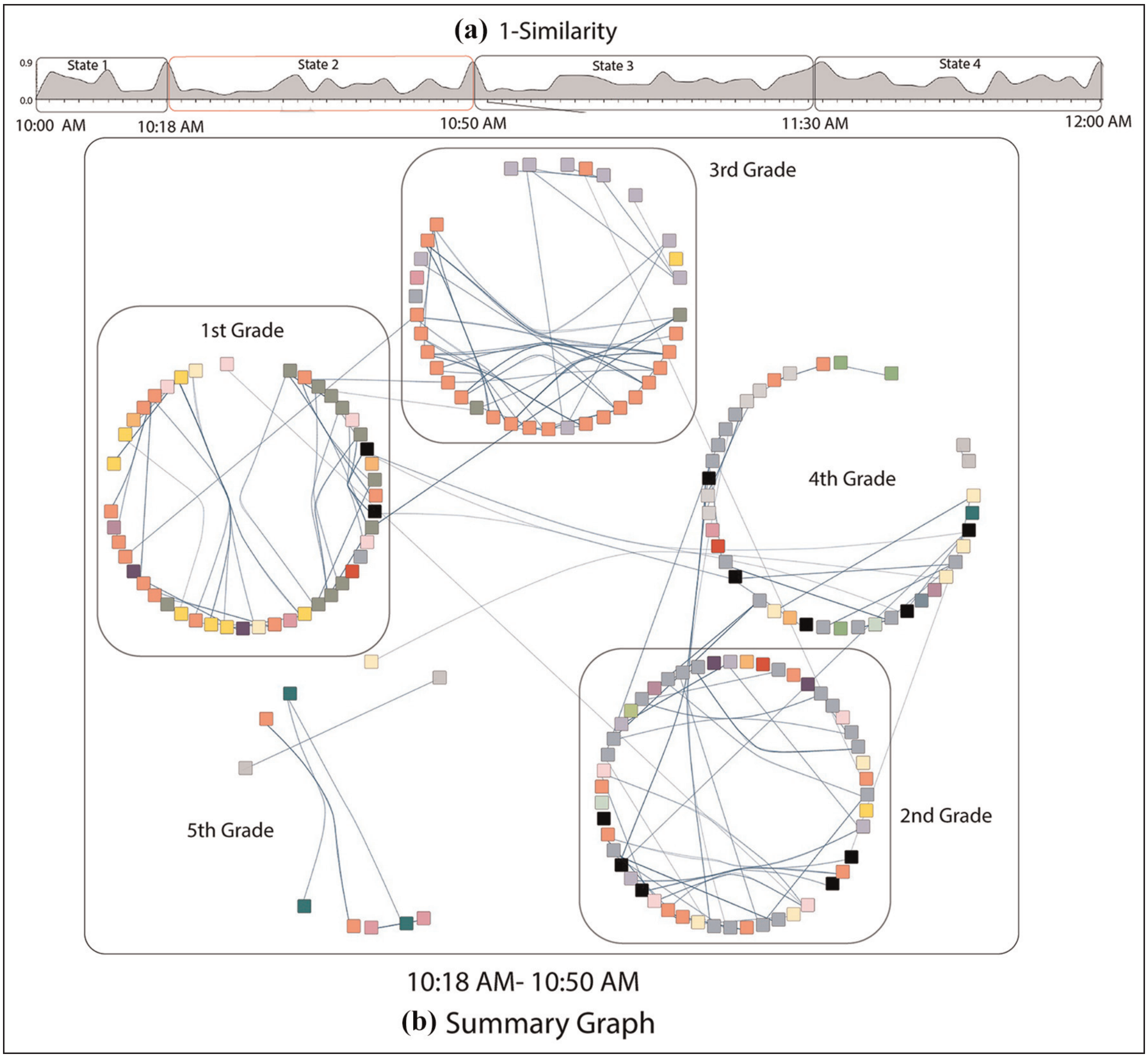

Figure 6. Major states of primary school data set. The five circular node layouts represent students from first grade to fifth grade. (a) Topological changes between adjacent timesteps. The state-detection method uses all pair similarity values between nodes to identify highly similar states. (b) Summary graph depicting major communication patterns of students on the playground.

communities. Fourth graders kept a smaller group of students working with them, compared to students in other grades.

Our system allows one to observe consistent network structure over time, for example, where students performed group tasks within grades in our scenario, see Figure 6(b). The summary graphs effectively capture emerging communication between third graders and fifth graders. Approaches like small-multiples or animations rely on the user to identify states or phases in a data set, while our approach, together with a meaningful distance metric, can identify and visualize states automatically. Furthermore, inter- and intra-community patterns, for example, are apparent in graph-based analysis and visualization.

TimeSum used in neuroscience

To demonstrate the value of our approach, we applied it to two brain network data sets. We performed the presented case studies in collaboration with neuroscientists, co-authors of this article, who are experts in brain network neuroscience. The studies concern exploratory data analysis, where the goal was to visualize global network dynamics and their relationships to brain regions, with the purpose of generating scientific hypotheses that would later be studied with rigorous statistical methods. Neuroscientists use functional magnetic resonance imaging (fMRI) to measure whole-brain activity, which can be modeled as a set of brain regions (nodes) connected by edges reflecting

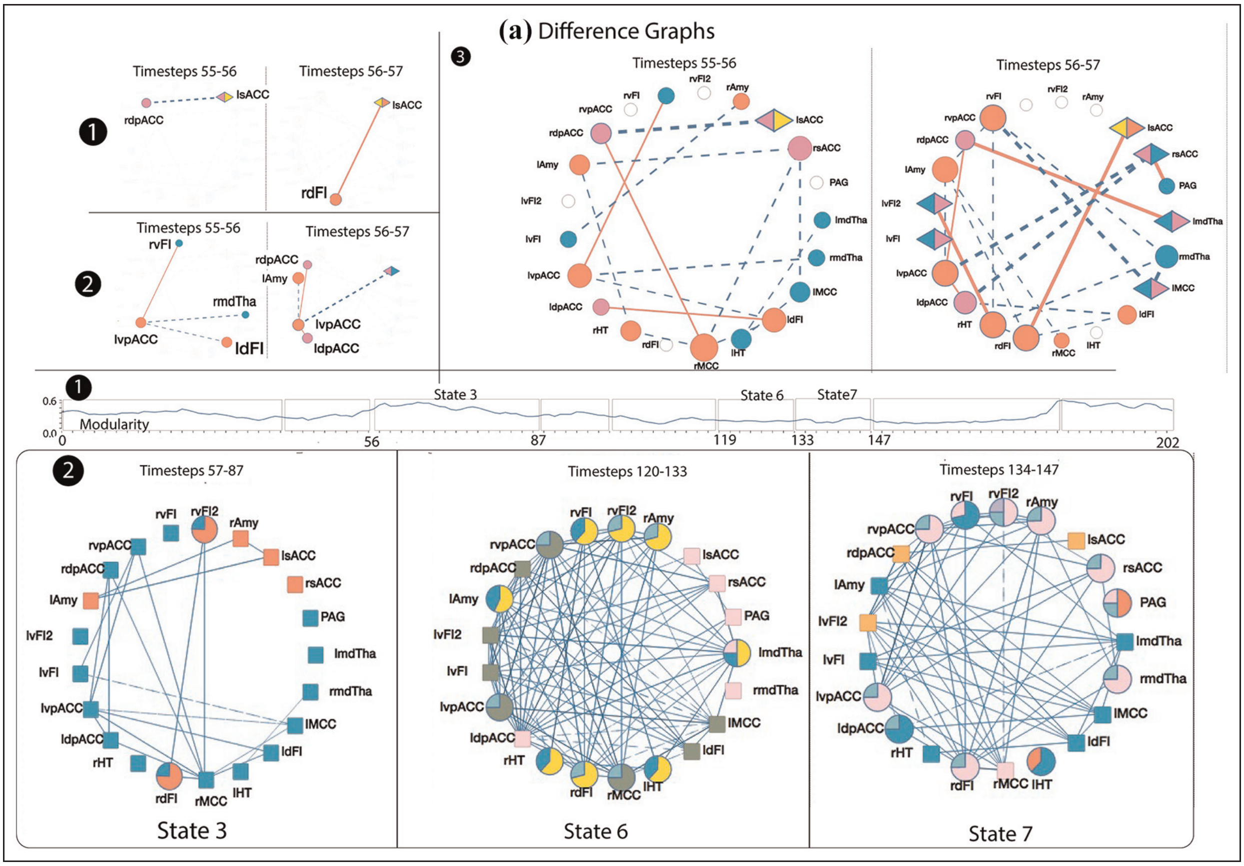

Figure 7. (a) Local dynamics of nodes lsACC and lvpACC (1, 2) for timesteps 55–58. During timestep 55–56 lvpACC changes from receive to transmit mode, see section "Case study 1." (b) Global topological dynamic behavior during timesteps 55–58, defining an important change of brain state from integrated to modular. (c) Summary graphs depicting evolution of average topology, that is, from sparse to dense to sparse. Integrated states (1,2,3,4,5,6,7, and 8) are energetically demanding, requiring increased blood flow,[45] while modular states, for example, states 2 and 9, have shorter edges, indicative of maintaining a stable structure (many squares).

the correlations between their activities. Changes in topology over time can be conceptualized as switches between multiple quasi-stable functional states.[42] Identifying the temporal intervals with similar topological structure is important to understand the interaction of behavioral systems when performing a task or their impairment due to a disorder.[43]

Case study 1

The objective of this case study was to explore and understand the dynamics concerning the salience network[44] of a healthy older adult. This network represents the correlations between regions in the brain responsible for the monitoring of sensory, visceral, and the reward or threat system. It consists of input nodes in anterior insula (e.g. dFI and vFI) that gather sensory information and output nodes in anterior cingulate (e.g. dpACC and vpACC) that initiate behavioral actions. A major challenge in analysis is understanding the dynamic fluctuations of connectivity profiles between these two groups or nodes over time.

The processed time-varying data (pairwise time-series correlations between 21 regions, 202 timesteps) was used as input to our system to identify dynamic communities, state-intervals, and abstract graph topologies. The system detected nine states with similar graph-level properties. The modularity and (one-similarity) plots, see Figure 7(b), convey the temporal structure and the extent of the individual states in the data set (T1).

We can clearly see consistent patterns for each state. Two major structures are revealed, a sparse (highly modular) structure (states 3 and 9) and a densely integrated structure (states 1, 2, 4, 5, 6, 7, and 8) (T2), which may correspond to existing studies in neuroscience.[42] The state-detection method was able to abstract variabilities within those two major categories,

Table 1. The table statistically compares the properties of the detected states.

States	Time	*C*	*Q*	ED	AEP
State 1	0–40	0.79	0.32	0.30	2.56
State 2	41–56	0.77	0.19	0.30	6.4
State 3	57–87	0.83	0.38	0.28	3.2
State 5	100–119	0.78	0.14	0.40	8.3
State 6	120–133	0.74	0.16	0.53	5.21
State 7	134–146	0.72	0.12	0.44	6.94
State 8	147–182	0.84	0.16	0.31	3.14
State 9	183–202	0.74	0.56	0.19	7.03
Data set	0–202	0.79	0.29	0.22	NA

We generally find two major types of states, with low modularity and high density or with high modularity and low density. We further find that the difference in modularity is drastic from state 2 to state 3. *C*: average temporal correlations (average of equation (4)); *Q*: modularity; ED: edge density; AEP: average number of edges pruned.

see Table 1. One also observes this phenomenon as a large number of squares (stable communities) in the summary graph for time 57–87, see Figure 7(b-3), state 3. Most of the other summary graphs have glyphs, representing regions that dynamically change communities.

The more globally integrated and less modular states had earlier been suggested to be more energetically demanding, requiring increased cerebral blood flow, explaining the dense structure.[45] States having high modularity and high stability, like states 3 and 9, require less overall "cost" (fewer/shorter edges) than more globally integrated states. This could explain its maintenance as a stable structure. When analyzing the dynamics of state 3, see Figure 7(b-3), state 3, one can quickly identify two brain regions right ventral frontoinsula, that is, rvFI2 and right dorsal frontoinsula, that is, rdFI with flexible community membership. This could suggest that these regions may have switched their roles (T3).

To explore intrinsic structural change of the network between states two and three, we drill down and visualize the difference graphs from times 55 to 58, see Figure 7(a-3). A drastic topological change is visible. The large number of diamonds at time 56–57 indicates the extent of community changes taking place. In general, many regions lose their correlations (dashed lines), which could explain the network's shift toward a higher degree of modularity (high *Q* in state 3) (T3).

To understand integral node dynamics of brain regions during state transitions, we hover over and interact with the brain region ventral pregenual anterior cingulate cortex, that is, IvpACC, see Figure 7(a-2). The change of local neighborhood of the node is clearly visible, that is, changes in the lvpACC's connectivity from the vFI and dFI, its main input nodes, to the dpACC, one of its main output nodes. The node lvpACC eventually changes its membership (T4). One explanation of this highly flexible node configuration is that the region oscillates between receive (input) and transmit (output) modes. A detailed fMRI analysis would be necessary to test this hypothesis.

Case study 2

Exploratory visual analysis in dynamic networks is key to observe general trends, explore temporal variability, and understand higher-level organization of the network. This case study involved a domain expert who was able to construct and articulate hypotheses on-the-fly, during interactive data analysis process. The goal was to understand the dynamics of the resting state brain network connectivity and its impact on individual nodes and their connections. Specific questions were as follows: What are the major topological changes (spontaneous global dynamics) (T1) in the data set? How do they relate to local changes in the network (T4)?

We used a whole-brain, high-temporal resolution (TR = 720 ms) resting state fMRI data set[46] and extracted time-series from a 36-node functional parcellation. Using multiplication of temporal derivatives,[42] we were able to calculate 1200 time-varying network matrices. To remove noise, we smoothed the data set with a 14 times (volume) sliding window.

Based on initial plots (1-similarity and modularity) of the full 1200 volume time-series data set, we filtered a 140 times (400–560) window containing consistent similarity patterns, visible as peaks in the similarity plot. Using this data set as input, TimeSum identified time-varying communities, state-intervals, and summarized topology. Two major classes of network states were identified: one with relatively high density and low modularity (times 42–54, 96–120); one with low density and high modularity (times 0–40, 60–96, and

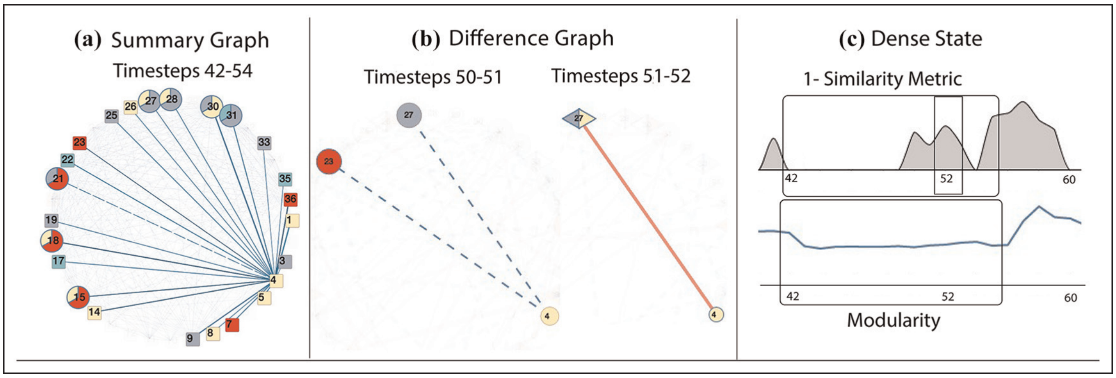

Figure 8. (a) Exploration of posteromedial cortex, node 4, in an integrated state from timesteps 42–54 through the summary graph. (b) Detailed exploration of changes in local topology of node 4 through difference graph for timesteps 50–52. (c) Plots for 1-similarity and modularity of the identified state. Filtering within-state variability for timesteps 50–52.

121–140), see Figure 8(b) (T1). This behavior may correspond to the results from a previously performed statistical analysis of the data, which had identified tight, integrated and modular, segregated states.[42]

To better understand network changes driving dynamic behavior, we drilled down into a time sub-interval (42–54), investigating a state transition, see Figure 8(b). Our domain expert focused on visualizing the dynamics of node 4, encompassing brain region posteromedial cortex, one of the most connected areas of the human brain. As shown in the summary graph in Figure 8(a), posteromedial cortex exhibited diverse connectivity with multiple flexible regions (frequent community changes) within the state, (i.e. nodes 27, 28, 30, and 31) (T2 and T4). Interestingly, node 4 (posteromedial cortex) itself does not change its community during this time, and this relative stability matches previous research done regarding the dynamics of this brain region.[42]

Within the selected sub-interval, the similarity plot suggested large changes of network structure during time 50–52. Visualization of these changes using a difference graph, see Figure 8(b),(T3 and T4), only for node 4, may provide an explaination of the relatively large shift in global network structure (visible in the similarity plot), driven by changes in few edges and shifts in community structure. Additional research and rigorous statistical analysis is needed to validate the network dynamics inferred by TimeSum.

To summarize, our domain experts could easily identify major topological shifts in the data sets used for the presented case studies. The case studies demonstrate that our approach has great potential for rapidly formulating initial hypotheses for different types of applications requiring complex network analysis. Traditional approaches used to visualize such data use small-multiples or animations. Such methods do not effectively depict major topological patterns, like birth or death of communities or transient nodes, see Figure 8(b), node 27.

Discussion

Traditionally, analysts use their expert knowledge, experience, and perceptual skill to identify and glean core topological structure of the network and its dynamic behavior. TimeSum, based on its algorithmic and visual methods, is able to characterize important changes happening in a network and compute the core network structure characterizing a state.

Domain expert feedback

Our domain expert (neuroscientist) has used our tool and is convinced of its value. He was quickly able to identify higher-level topological network changes and identify trend behavior in the connectivity profiles of the nodes. He stated,

> The interactive visual techniques introduced, enabled an intuitive understanding of the systems we study. For example, the summary graphs provided high-level overviews of network structure and how it changes. The difference graph allowed me to get a sense of variability within each state and comprehend the dependence between dynamics across timepoints. It would be great to add more layout techniques to understand modular structure better.” He added that “there haven’t really been any good ways to visualize time-varying network data in a way that facilitates detailed and intuitive understanding. TimeSum

allows me to perform the kind of deep data exploration that is likely to help substantially for gaining new insights.

Our method was designed primarily to address the temporal scalability problem arising in time-varying network data analysis. Through our experiments, we determined that our system can be used as an interactive, near-real-time system for data sets consisting of 240+ nodes, 4,608,400 edges, and 200+ timesteps. The processing time required for interactive filtering and switching views is negligible. As all abstraction algorithms are executed online, the system requires approximately 5–6 s to start up.

Scalability

For larger data sets, our technique demands more screen space and computational power. Nevertheless, our method produces intuitively understandable results when graphs exhibit temporal correlation and inherent modular structure. Many naturally occurring networks have such properties. While scalability of computational aspects of our approach is important, we believe that "perceptual scalability" is perhaps even more important, and perceptual scalability was the main focus of our research. Our approach provides an effective means for visualizing similarities and differences between graphs, while avoiding visual clutter and greatly reducing edge crossings. Reduction of complexity can be achieved by sampling using centrality, for example, aggregating nodes of lower importance. Furthermore, methods like semantic zooming, providing additional details on demand further deal with visualizing large number of nodes. However, our method focuses on the complexity of temporal behavior. Other methods, such as edge-bundling or meta-node reduction, could be used in conjunction to deal with perceptual scalability.

Tuning parameter values

The threshold used for the computation of summary graphs affects the computation of topology. A resulting graph will be sparse if the algorithm finds similar topological behavior between dissimilar graphs in a state. For example, AEP, see section "Consensus-based state summary graphs," is an indicator one can use to determine the reliability of the summarization algorithm. Our technique requires a user to specify the values of several parameters, for example, δ values used as weights to define similarities, threshold values for summary graph generation and state detection, and so on. Proper choices of values depends on the particular data set to be analyzed and the questions to be answered.

Comparison with other techniques

Domain experts commonly use small-multiples or animations to understand the dynamics and evolution of graphs over time, see Figure 9. We compare our approach to common methods and explain the conceptual advances our technique offers. Figure 9(a) shows two graphs (101 and 102) in the same state (timestep interval 100–105), with nodes being colored by the dynamically changing community membership. Figure 9(b) and (c) shows the difference and summary network computed by our technique.

Considering small-multiples, to identify (T1) states and analyze summary topologies (T2), the method relies on a user's visual perceptual skills to identify commonalities in the behavior of similar graphs. Generally, a user must manually inspect a large number of views. To understand such variations in topology (T3) and comprehend complex local dynamics (T4) in dense graphs, finding differences is a cognitively overwhelming and extremely time-consuming task. Users often fail to notice major differences between two adjacent graphs since recognizing changes can be perceptually challenging.

Animations

The use of visual animations, which is limited to a user's short-term memory, is often ineffective for the analysis of complex time-varying networks. A user must remember changes in community membership, deal with unstable layouts necessary to classify states (T1), and mentally determine commonalities (T2). Furthermore, identifying small-scale changes in the topology of dense graphs is a complex and cognitively demanding task (T3 and T4). Tasks like identifying major events like growth or death of communities is challenging through animations due to its constantly fluctuating and dynamic feature of the visualization.

To support efficient state analysis of complex time-varying networks, our method automatically computes an unchanging network topology and its corresponding time interval, see Figure 9(c). This approach reduces redundancy in visual representations used to depict the same information. It also increases visual comprehension of common connectivity patterns (T1 and T2) through abstraction and reduction of visual clutter. With our approach, small-scale changes in topology-causing community changes can be identified quickly via community-based difference graphs, see Figure 9(b), depicting a community change of node 15 (T3 and T4). The combination of these views supports a user substantially to understand similarities and differences in toplogy and communities. Without our system, such a detailed analysis of a time-varying network

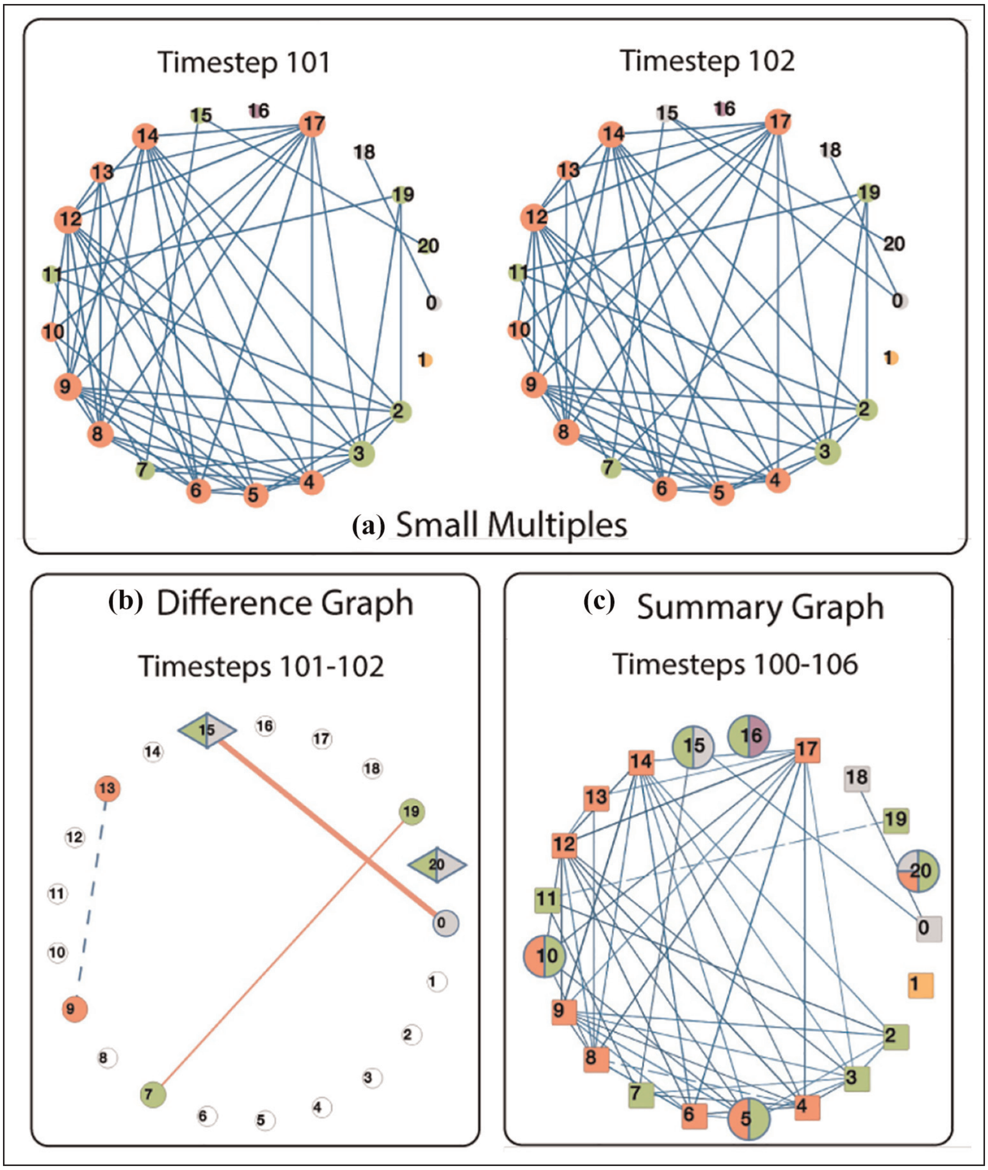

Figure 9. Our technique reduces the time-varying data set into summary graphs. Difference graphs convey the causes and effects of topological changes. Here, the small multiples technique (a) is compared with the summary graph (c) and difference graph (b). Considering the summary graph, we can identify the nodes that often change community membership. The addition of the edge from node 0 to node 15 causes nodes 15 and 20 to change their community (gray).

would require a user to spend a significant amount of time for manual inspection of a much larger number of views and develop an instinct to find the common topological structure.

Conclusion and future work

We have presented TimeSum, an approach for exploring dynamic networks and presenting their complex topologies via effective abstractions. We have introduced a unique set of algorithms to identify time intervals related to similar topological network properties, allowing one to comprehend global trends in a network data set. TimeSum is a powerful tool for dealing with the temporal visual scalability problem, greatly reducing the need for time-consuming manual network analysis steps. With the help of our tool, experts could comprehend rapidly the causes and effects of topological changes occurring in networks, which, using commonly available methods, would require cumbersome off-line data processing.

We plan to focus on the crucially important aspect of scalability for effective network data processing and visualization. As data sets become larger, in terms of nodes and timesteps, methods are needed that can display large data meaningfully. We plan to consider combining temporal and spatial network information by leveraging dynamic techniques, such as orchestrating changes in node appearance (e.g. semantic zooming). Furthermore, it is possible to improve our exploration approach by enabling difference graph computation with not only adjacent timepoints but also time-discontinuous timepoints.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

ORCID iD

Sugeerth Murugesan https://orcid.org/0000-0002-9889-3905

References

1. McFarland DA. Student resistance: how the formal and informal organization of classrooms facilitate everyday forms of student defiance 1. *Am J Sociol* 2001; 107(3): 612–678.
2. Becker RA, Eick SG and Wilks AR. Visualizing network data. *IEEE T Vis Comput Gr* 1995; 1(1): 16–28.
3. Beck F, Burch M and Diehl S. Towards an aesthetic dimensions framework for dynamic graph visualizations. In: *Proceedings of the 13th international conference information visualisation*, Barcelona, 15–17 July 2009, pp. 592–597. New York: IEEE.
4. Beck F, Burch M, Diehl S, et al. The state of the art in visualizing dynamic graphs. In: *Proceedings of the Eurographics conference on visualization*, 2014, https://pdfs.semanticscholar.org/a4db/7fdb7224456e98bd41b6803b5fb29296ec01.pdf
5. Hu Y, Kobourov SG and Veeramoni S. Embedding, clustering and coloring for dynamic maps. In: *Proceedings of the Pacific visualization symposium (PacificVis)*, Songdo, South Korea, 28 February–2 March 2012, pp. 33–40. New York: IEEE.
6. Ghani S, Elmqvist N and Yi JS. Perception of animated node-link diagrams for dynamic graphs. *Comput Graph Forum* 2012; 31: 1205–1214.
7. Rosvall M and Bergstrom CT. Mapping change in large networks. *PLoS ONE* 2010; 5(1): e8694.
8. Burch M, Vehlow C, Beck F, et al. Parallel edge splatting for scalable dynamic graph visualization. *IEEE T Vis Comput Gr* 2011; 17(12): 2344–2353.
9. Murugesan S, Bouchard K, Chang E, et al. Multi-scale visual analysis of time-varying electrocorticography data via clustering of brain regions. *BMC Bioinformatics* 2017; 18(6): 236.
10. Vehlow C, Beck F, Auwärter P, et al. Visualizing the evolution of communities in dynamic graphs. *Comput Graph Forum* 2015; 34: 277–288.
11. Liu Q, Hu Y, Shi L, et al. EgoNetCloud: event-based egocentric dynamic network visualization. In: *Proceedings of the visual analytics science and technology (VAST)*, Chicago, IL, 25–30 October 2015, pp. 65–72. New York: IEEE.
12. Bach B, Pietriga E and Fekete JD. GraphDiaries: animated transitions and temporal navigation for dynamic networks. *IEEE T Vis Comput Gr* 2014; 20(5): 740–754.
13. Van den Elzen S, Holten D, Blaas J, et al. Reducing snapshots to points: a visual analytics approach to dynamic network exploration. *IEEE T Vis Comput Gr* 2016; 22(1): 1–10.
14. Dal Col A, Valdivia P, Petronetto F, et al. Wavelet-based visual analysis of dynamic networks. *IEEE T Vis Comput Gr* 2018; 24: 2456–2469.
15. Margulies DS, Böttger J, Watanabe A, et al. Visualizing the human connectome. *Neuroimage* 2013; 80: 445–461.
16. Toivonen H, Mahler S and Zhou F. A framework for path-oriented network simplification. In: *Proceedings of the international symposium on intelligent data analysis*, 2010, pp. 220–231. Berlin: Springer, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.478.725&rep=rep1&type=pdf
17. Shi L, Liao Q, Sun X, et al. Scalable network traffic visualization using compressed graphs. In: *Proceedings of the international conference on big data*, Silicon Valley, CA, 6–9 October 2013, pp. 606–612. New York: IEEE.
18. Boldi P, Rosa M, Santini M, et al. Layered label propagation: a multiresolution coordinate-free ordering for compressing social networks. In: *Proceedings of the 20th international conference on World Wide Web*, Hyderabad, India, 28 March–1 April 2011, pp. 587–596. New York: ACM.
19. Toivonen H, Zhou F, Hartikainen A, et al. Compression of weighted graphs. In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, San Diego, CA, 21–24 August 2011, pp. 965–973. New York: ACM.
20. Dunne C and Shneiderman B. Motif simplification: improving network visualization readability with fan, connector, and clique glyphs. In: *Proceedings of the SIGCHI conference on human factors in computing systems*, Paris, 27 April–2 May 2013, pp. 3247–3256. New York: ACM.
21. Sun J, Faloutsos C, Papadimitriou S, et al. GraphScope: parameter-free mining of large time-evolving graphs. In: *Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining*, San Jose, CA, 12–15 August 2007, pp. 687–696. New York: ACM.
22. Eagle N and Pentland AS. Reality mining: sensing complex social systems. *Pers Ubiquit Comput* 2006; 10(4): 255–268.
23. Rashid B, Damaraju E, Pearlson GD, et al. Dynamic connectivity states estimated from resting fMRI identify

differences among Schizophrenia, bipolar disorder, and healthy control subjects. *Front Hum Neurosci* 2014; 8: 897.
24. Mutlu AY, Bernat E and Aviyente S. A signal-processing-based approach to time-varying graph analysis for dynamic brain network identification. *Comput Math Method M* 2012; 2012: 451516.
25. Archambault D, Purchase HC and Pinaud B. Difference map readability for dynamic graphs. In: *Proceedings of the international symposium on graph drawing*, Konstanz, 21–24 September 2010, pp. 50–61. Berlin: Springer.
26. Archambault D. Structural differences between two graphs through hierarchies. In: *Proceedings of graphics interface 2009*, Kelowna, BC, Canada, 25–27 May 2009, pp. 87–94. Toronto, ON, Canada: Canadian Information Processing Society.
27. Bourqui R and Jourdan F. Revealing subnetwork roles using contextual visualization: comparison of metabolic networks. In: *Proceedings of the 12th international conference on information visualisation*, London, 9–11 July 2008, pp. 638–643. New York: IEEE.
28. Rufiange S and McGuffin MJ. DiffAni: visualizing dynamic graphs with a hybrid of difference maps and animation. *IEEE T Vis Comput Gr* 2013; 19(12): 2556–2565.
29. Beck F, Burch M, Diehl S, et al. A taxonomy and survey of dynamic graph visualization. *Comput Graph Forum* 2017; 36: 133–159.
30. Alper B, Bach B, Henry Riche N, et al. Weighted graph comparison techniques for brain connectivity analysis. In: *Proceedings of the SIGCHI conference on human factors in computing systems*, Paris, 27 April–2 May 2013, pp. 483–492. New York: ACM.
31. Greene D, Doyle D and and Cunningham P. Tracking the evolution of communities in dynamic social networks. In: *Proceedings of the advances in social networks analysis and mining (ASONAM)*, Odense, 9–11 August 2010, pp. 176–183. New York: IEEE.
32. Koutra D, Vogelstein JT and Faloutsos C. DeltaCon: a principled massive-graph similarity function. In: *Proceedings of the SIAM international conference on data mining*, 2013, pp. 162–170, https://pdfs.semanticscholar.org/d267/8fbdff395ff8b0312b0f15336c7f66702242.pdf
33. Blondel VD, Guillaume JL, Lambiotte R, et al. Fast unfolding of communities in large networks. *J Stat Mech-Theory E* 2008; 2008; P10008.
34. Hutchison RM, Womelsdorf T, Allen EA, et al. Dynamic functional connectivity: promise, issues, and interpretations. *Neuroimage* 2013; 80: 360–378.
35. Elmqvist N and Fekete JD. Hierarchical aggregation for information visualization: overview, techniques, and design guidelines. *IEEE T Vis Comput Gr* 2010; 16(3): 439–454.
36. Lavie N and Cox S. On the efficiency of visual selective attention: efficient visual search leads to inefficient distractor rejection. *Psychol Sci* 1997; 8(5): 395–396.
37. Purchase HC, Hoggan E and Görg C. How important is the “mental map”? An empirical investigation of a dynamic graph layout algorithm. In: *Proceedings of the 14th international conference on graph drawing*, Karlsruhe, 18–20 September 2006, pp. 184–195. Berlin: Springer.
38. Zaman L, Kalra A and Stuerzlinger W. The effect of animation, dual view, difference layers, and relative re-layout in hierarchical diagram differencing. In: *Proceedings of graphics interface*, St. John’s, NL, Canada, 25–27 May 2011, pp. 183–190. Waterloo, ON, Canada: Canadian Human-Computer Communications Society, University of Waterloo.
39. McFarland D. McFarland classroom dataset, 2001, https://rdrr.io/cran/networkDynamic/man/classrooms.html
40. Stehlé J, Voirin N, Barrat A, et al. High-resolution measurements of face-to-face contact patterns in a primary school. *PLoS ONE* 2011; 6(8): e23176.
41. Holten D and Van Wijk JJ. Force-directed edge bundling for graph visualization. *Comput Graph Forum* 2009; 28: 983–990.
42. Shine JM, Koyejo O and Poldrack RA. Temporal metastates are associated with differential patterns of time-resolved connectivity, network topology, and attention. *Proc Natl Acad Sci U S A* 2016; 113: 9888–9891.
43. Braun U, Schäfer A, Walter H, et al. Dynamic reconfiguration of frontal brain networks during executive cognition in humans. *Proc Natl Acad Sci U S A* 2015; 112(37): 11678–11683.
44. Seeley WW, Menon V, Schatzberg AF, et al. Dissociable intrinsic connectivity networks for salience processing and executive control. *J Neurosci* 2007; 27(9): 2349–2356.
45. Zalesky A, Fornito A, Cocchi L, et al. Time-resolved resting-state brain networks. *Proc Natl Acad Sci U S A* 2014; 111(28): 10341–10346.
46. Van Essen DC, Smith SM, Barch DM, et al. The Wu-Minn Human Connectome Project: an overview. *Neuroimage* 2013; 80: 62–79.