SimNet: A new algorithm for measuring brain networks similarity

Ahmad Mheich, Mahmoud Hassan, Fabrice Wendling
Université de Rennes 1, LTSI, F-35000, France INSERM, U1099, Rennes, F-35000, France
firstname.name@univ-rennes1.fr

Ahmad Mheich, Mohammad Khalil
Centre AZM, EDST
Lebanese University, Tripoli, Lebanon
firstname.name@ul.edu.lb

Olivier Dufor, Vincent Gripon, Claude Berrou
Télécom Bretagne (Institut Mines-Télécom)
UMR CNRS Lab-STICC
Brest, France
firstname.name@telecom-bretagne.eu

***Abstract*— Measuring similarity among graphs is recognized as a non-trivial problem. Most of the algorithms proposed so far ignore the spatial location of vertices, which is a crucial factor in the context of brain networks. In this paper, we present a novel algorithm, called "SimNet", for measuring the similarity between two graphs whose vertices represent the position of sources over the cortex. The novelty is to account for differences at the level of spatially-registered vertices and edges. Simulated graphs are used to evaluate the algorithm performance and to compare it with methods reported elsewhere. Results show that SimNet is able to quantify the similarity between two graphs under a spatial constraint based on the 3D location of edges.**

The application of SimNet on real data (dense EEG) reveals the presence of spatially-different brain networks modules activating during cognitive activity.

Keywords— ***graph similarity; vertices physical location; brain networks*;**

I. Introduction

Functional brain networks represent the interactions between distinct and distant (sub)cortical regions [1] during cognitive activity. These networks are usually represented as graphs where brain regions are denoted by vertices and where the functional connectivity is denoted by edges [2]. In this context, graph theory based analysis was shown to be the most efficient approach to reveal the characteristics of these networks [3].

Over the past decades, graph theory has enormously developed and extensive literature exists about the plethora of methods proposed to characterize graph properties. Briefly, these measures fall into two categories. Some are related to vertex properties, such as the degree, the strength and the presence of hubs [4]. The others are related to the global features of the considered graph such as density and modularity [5].

As compared with the large number of methods aiming at characterizing graph properties [4], less attention has been paid to algorithms aimed at comparing graphs. So far, few algorithms have been proposed to measure graph similarity, such as the use of graphs and subgraphs isomorphism [6], graph edit distance [7] and Levenshtein distance [8]. However, in the context of brain networks, the location of nodes is a key factor for the graph characterization as reported very recently [9]. The physical location of vertices in brain networks is a major factor to measure similarity between graphs. Indeed, two networks with identical properties but interconnecting different brain areas should be considered to have low similarity. Conversely, two graphs with dissimilar properties but interconnecting similar brain regions should be considered being closer. Pineda-Pardo et al. [9] showed in a recent study that the 3D coordinates of the brain vertices are important to characterize the brain networks.

In this paper, we propose a novel algorithm for computing the similarity between two graphs. It makes use of the fact that the topological property represented by the physical location of vertices is a crucial parameter. On the one side, this algorithm combines the vertex and edge similarities and on the other side it takes into account the physical locations (3D coordinates) of the vertices (brain sub-regions in our case). The performance of the proposed algorithm is analyzed on synthetic graphs. The algorithm is then evaluated on a real application aimed at detecting modules in brain networks during a picture naming task.

II. Materials And Methods

A. Definitions

Let us consider undirected, simple, weighted graphs. A graph G is a pair of sets $G=(V,E)$, where V is the set of vertices (with known Cartesian coordinates), and $|V|$ is the order of the graph (number of vertices). $E \subseteq V \times V$ defines the edges. The graph is said to be simple if there is no edge linking a vertex with itself. We denote by $W^{G}_{n,m}$ the weight of the edge between vertices n and m in graph G. The graph is said to be undirected if the adjacency matrix is symmetric. The similarity measure between two graphs G_1 and G_2 is denoted by $sim(G_1,G_2)$. A graph G is located into a grid Gr (u) where u is the total number of vertices in Gr. We introduce the matrix S as the Euclidian distance matrix between all the

This work was supported by AZM and SAADE Association (Tripoli, Lebanon) and the Rennes University Hospital (COREC Project named conneXion, 2012-14). The work has also received a French government support granted to the CominLabs excellence laboratory and managed by the National Research Agency in the "Investing for the Future" program under reference ANR-10-LABX-07-01. This work was also supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement n° 290901

vertices of *Gr*. Lines and columns in S represent the vertices. The values of S are the Euclidian distances between vertices coordinates, S is symmetric with zero values in the diagonal.

B. Proposed similarity algorithm "SimNet"

Let us consider two graphs $G_1(V_1,E_1)$ and $G_2(V_2,E_2)$ with possibly distinct numbers of vertices and edges. The objective is to propose a method able to provide a "distance" between G_1 and G_2. To proceed, our method is based on estimating the minimum number of transformations (deletion, insertion, substitution of vertices and edges) that are necessary to transform G_2 into G_1. More precisely, we decompose the problem into two steps:

First step: "vertex distance" between G_1 and G_2, this part can be summarized as following:

- Calculate the Euclidian distances between each pair of vertices in the grid to obtain the matrix of distances S
- Initialize the vertices distance dv between G_1 and G_2
- Compute the intersection of the sets of vertices $C = V_1 \cap V_2$
- Define $V_1' = V_1 - C$ and $V_2' = V_2 - C$ as the sets of vertices that belong to one graph and do not belong to the other graph.
- Define a sphere with radius R
- Route the sphere on each vertex n of V_1' in G_2
- Substitute the vertex m of V_2' with the vertex n of V_1' in G_2, if m located in the sphere, in other words the distance between n and m is less than R, the cost of substitution is equal to the Euclidian distance between n and m.
- Delete the rest of vertices that belongs to G_2 and do not substitute with any other vertex that belongs to G_1.
- Insert vertices in G_2 that belong to G_1 and do not belong to G_2.

The cost of substitution between two vertices is equal to the Euclidian distance between these two vertices. This cost is less than the cost of insertion and the cost of deletion of a vertex which are equal to a constant value. In our study, the cost of insertion or deletion of a vertex is equal to the maximal distance between two vertices, which is the maximal value in the matrix S.

Second step: edge distance between G_1 and G_2:

In this step, the distance between the edges of G_1 and G_2 is computed. It consists in calculating the weight difference between two edges into two different graphs. We use equation (1) to calculate the edge distance, the $diff(W_{n,m}^{G_1}, W_{n,m}^{G_2})$ score $\in [0,1]$ where 1 means that an edge exists just in one graph and does not exist in the other, while 0 means an edge exists for both graphs between the same vertices.

$$diff(W_{n,m}^{G_1}, W_{n,m}^{G_2}) = | W_{n,m}^{G_1} - W_{n,m}^{G_2} | \quad (1)$$

The distance between two graphs is then calculated by combining the vertices distance *(dv)* and the edges distance:

$$D(G_1,G_2) = dv + \sum_{n=1}^{|V|}\sum_{m=2}^{|V|} diff(W_{n,m}^{G_1}, W_{n,m}^{G_2}) \quad (2)$$

We convert the distance $D(G_1,G_2)$ to the similarity measure $sim(G_1,G_2)$ via the formula $sim = (1/(1+D))$. The similarity score $\in [0,1]$ where 0 means that G_1 and G_2 are totally dissimilar while 1 means that G_1 and G_2 are identical.

C. Methods used for measuring graph simialrities

We compare our algorithm, SimNet, with four already existing algorithms in the state of the art of graph similarity domain:

1- **Graph edit distance:[10]** This distance is based on the transformation of one graph to another using elementary operations. If we associate an elementary cost for each operation, the edit distance is defined as the sequence that requires the least cost to transform G_1 into G_2.

$$sim_{GED}(G_1,G_2) = |V_1| + |V_2| - 2|V_1 \cap V_2| + |E_1| + |E_2| - 2|E_1 \cap E_2|$$

2- **DeltaCon method:** Assess the similarity between two graphs on the same number of vertices, details in [11].

3- **Vertex/Edge Overlap method (VEO):[12]** Two graphs are similar if they share many vertices and edges. For this method the similarity between two graphs $G_1(V_1,E_1)$ and $G_2(V_2,E_2)$ is defined as:

$$sim_{VEO}(G_1,G_2) = 2\frac{|V_1 \cap V_2| + |E_1 \cap E_2|}{|V_1| + |V_2| + |E_1| + |E_2|}$$

4- **λ-distance method (Spectral method): [13]**

$d_\lambda(G_1,G_2) = \sqrt{\sum_{l=1}^{L}(\lambda_{1l} - \lambda_{2l})^2}$, where $\{\lambda_{1l}\}^{G_1}$ and $\{\lambda_{2l}\}^{G_2}$ are the set of eigenvalues of the adjacency matrices of G_1 and G_2 where l is the index of the eigenvalue and L the total number.

D. Real Data

We used real data obtained from EEG signals measured on subjects performing a picture naming task. We asked twenty one subjects to name pictures shown on a screen using E-Prime 2.0 software (Psychology Software Tools, Pittsburgh, PA). The 148 pictures were selected from a database of 400 pictures standardized for French [14] and were used during two sessions of 74 stimuli. We recorded the brain activity using hr-EEG system (EGI, Electrical Geodesic Inc.). The collection of EEG signals was made with a 1 kHz sampling frequency and band-pass filtered between 3 and 45Hz [15, 16]. EEGLAB open source toolbox was used for the preprocessing steps. This study was approved by the National Ethics Committee for the Protection of Persons (CPP), (*conneXion* study, agreement number 2012-A01227-36, promoter: Rennes University Hospital). We excluded the electrodes located on the face as well as the few electrodes showing too high impedance.

A recent study realized by Hassan *et al*. [17] showed that the combination of the Phase Locking Value (PLV) as connectivity method and weighted Minimum Norm Estimator (wMNE) as source localization method applied to dense EEG provides the best performance to study functional connectivity at source level. The functional connectivity graphs obtained following this combination of methods are used for our present purpose.

In this paper, the graph *G* is defined as a set of vertices *V* representing the brain regions segmented from the Destrieux Atlas [18] and the edges *E* represent the functional connectivity computed between the EEG reconstructed sources.

III. Results

A. Synthetic test

To study the sensitivity of the five methods based on the variations of vertex locations, we generate a random graph *G* with 20 vertices located into a grid *Gr (20 × 20)*. The similarity between the graphs was then obtained at each noise level. Briefly, the noise level is defined as random substitution between each vertex of the graph and one of its neighbors (noise=1, the neighbors are the first square around the vertex; noise= 2, the neighbor are the two squares around the vertex; ...). We recalculate the similarity between the initial graph and the noised graphs at each noise level.

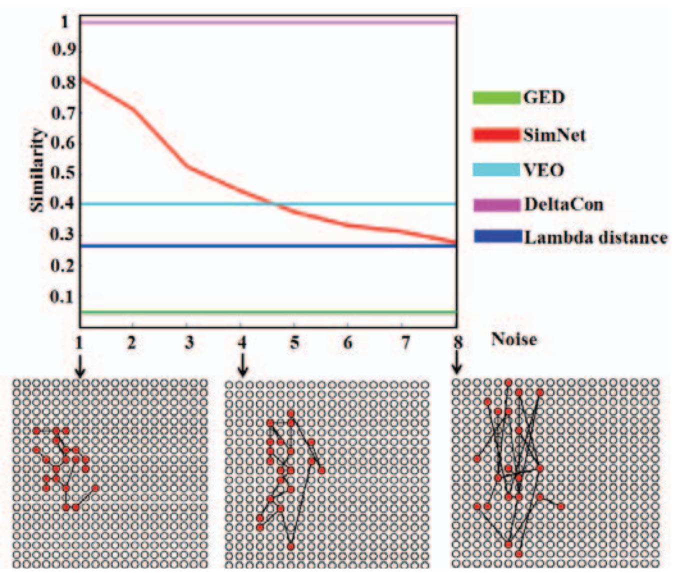

Figure 1: Variation of similarity values computed from the 5 algorithms under evaluation with respect to the level of noise.

Typical examples of the generated graphs are shown in figure 1 at three different noise levels. These steps were applied 1000 times and results were averaged for each similarity method. In figure 1, the dark color represents the averaging result of the similarity measures and the shadow represents their standard deviations. Results show that only SimNet algorithm presents a high ability to detect shifting of the vertices location. The other algorithms were unable to detect any shifting of vertices coordinates and their similarity values did not change during the variation of noise level.

B. Application to brain networks

SimNet was then applied to brain networks computed from the 21 subjects performing a picture naming task, the results being presented in figure 2. Figure 2A represents the similarity matrix that contains the similarity scores computed between all available graphs at each milliseconds scale. To visualize the highest value of similarities, we applied a threshold to obtain the matrix B that retains only the highest top 10% of the similarities score. In figure 2C we detect modules (groups of graphs that having high similarity values between them) in the similarity matrix using Louvain algorithm [19]. Three modules are detected. Module (1) in blue color appears in two intervals of time, module (2) in red color appears between 280 ms and 370 ms and very briefly around 540 ms. Finally module (3) in green color appears between 440 ms and 560 ms. We rearrange the three modules to show their lengths (figure 2D). The representation of the second module, from three different views is also shown in figure 2E. It shows a network with long-range occipital-frontal connections.

IV. Discussion

In this paper, we proposed a new algorithm called SimNet aiming at measuring the similarity between graphs in the context of brain networks when the physical location of vertices is supposed to be crucial.

On synthetic graphs, SimNet algorithm outperforms the other methods to detect shifting of the vertices location. Note that the other algorithms were developed for specific applications where the spatial location of graphs was not the primary intent.

One of the issues faced in our study is to specify the substitution zone between two vertices because increasing the radius (R) may increase the similarity score between two graphs. This parameter must be studied carefully to find the relation with the similarity score value. In this study we take the minimum distance between vertices in *S* as the radius value. The application of SimNet on real brain networks revealed the presence of different modules. These preliminary results seem to provide a new way of segmenting the cognitive process into different modules. These results have to be deeply compared with other algorithms mainly those detecting the so-called functional connectivity states of clusters [16].

In this paper, the distance between vertices was assumed to be Euclidian which seems to be not fully adequate as the brain surface is consisting of sulci and gyri (folded brain surface). Using the geodesic distance instead the Euclidian distance would likely be more appropriate in this case.

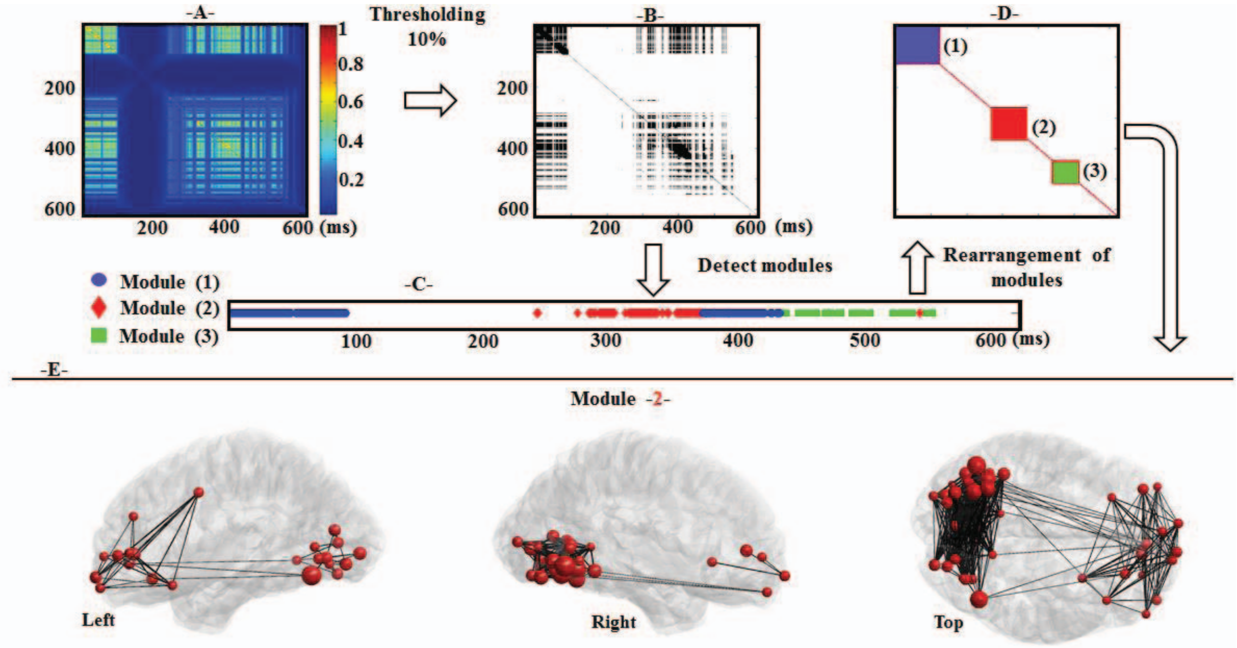

Figure 2: A-similarity score between connectivity graphs at each millisecond scale during picture naming task. -B-The same matrix in -A- after applying a threshold that retains only the top 10% of all similarities values. -C- Identified modules using Louvain method[19].–D- Rearrangement of the three modules. -E- Graphs representation of the second module from three different views (Top, Right, and Left).

V. Conclusion

In this paper, a new algorithm 'SimNet' for detecting similarity between graphs was proposed. The algorithm performance was evaluated using synthetic complex graphs. It shows a high ability to detect shifting of the vertices location and robustness to noise added to the vertex locations. SimNet also shows a good performance for the detection of modules in brain networks during picture naming task.

References

[1] A. R. McIntosh, "Towards a network theory of cognition," *Neural Networks,* vol. 13, pp. 861-870, 2000.
[2] K. J. Friston, "Functional and effective connectivity: a review," *Brain connectivity,* vol. 1, pp. 13-36, 2011.
[3] M. Rubinov and O. Sporns, "Complex network measures of brain connectivity: uses and interpretations," *Neuroimage,* vol. 52, pp. 1059-1069, 2010.
[4] L. d. F. Costa, F. A. Rodrigues, G. Travieso, and P. R. Villas Boas, "Characterization of complex networks: A survey of measurements," *Advances in Physics,* vol. 56, pp. 167-242, 2007.
[5] E. Bullmore and O. Sporns, "Complex brain networks: graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience,* vol. 10, pp. 186-198, 2009.
[6] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento, "A (sub) graph isomorphism algorithm for matching large graphs," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 26, pp. 1367-1372, 2004.
[7] X. Gao, B. Xiao, D. Tao, and X. Li, "A survey of graph edit distance," *Pattern Analysis and applications,* vol. 13, pp. 113-129, 2010.
[8] B. Cao, Y. Li, and J. Yin, "Measuring Similarity between Graphs Based on the Levenshtein Distance," *Appl. Math,* vol. 7, pp. 169-175, 2013.
[9] J. Á. Pineda-Pardo, K. Martínez, A. B. Solana, J. A. Hernández-Tamames, R. Colom, and F. del Pozo, "Disparate Connectivity for Structural and Functional Networks is Revealed When Physical Location of the Connected Nodes is Considered," *Brain topography,* pp. 1-10, 2014.
[10] H. Bunke, P. J. Dickinson, M. Kraetzl, and W. D. Wallis, *A graph-theoretic approach to enterprise network dynamics* vol. 24: Springer Science & Business Media, 2007.
[11] D. Koutra, J. T. Vogelstein, and C. Faloutsos, "DeltaC on: A Principled Massive-Graph Similarity Function," *SDM,* pp. 162-170, 2013.
[12] P. Papadimitriou, A. Dasdan, and H. Garcia-Molina, "Web graph similarity for anomaly detection," *Journal of Internet Services and Applications,* vol. 1, pp. 19-30, 2010.
[13] R. C. Wilson and P. Zhu, "A study of graph spectra for comparing graphs and trees," *Pattern Recognition,* vol. 41, pp. 2833-2841, 2008.
[14] F.-X. Alario and L. Ferrand, "A set of 400 pictures standardized for French: Norms for name agreement, image agreement, familiarity, visual complexity, image variability, and age of acquisition," *Behavior Research Methods, Instruments, & Computers,* vol. 31, pp. 531-552, 1999.
[15] A. Mheich, M. Hassan, O. Dufor, M. Khalil, C. Berrou, and F. Wendling, "Spatiotemporal Analysis of Brain Functional Connectivity," in *6th European Conference of the International Federation for Medical and Biological Engineering*, 2015, pp. 934-937.
[16] A. Mheich, M. Hassan, M. Khalil, C. Berrou, and F. Wendling, "A new algorithm for spatiotemporal analysis of brain functional connectivity," *Journal of neuroscience methods,*vol.242, pp.77-81, 2015.
[17] M. Hassan, O. Dufor, I. Merlet, C. Berrou, and F. Wendling, "EEG source connectivity analysis: from dense array recordings to brain networks," *PloS one,* vol. 9, p. e105041, 2014.
[18] C. Destrieux, E. Halgren, A. Dale, B. Fischl, and M. Sereno, "Variability of the human brain studied on the flattened cortical surface," in *Abstr.-Soc. Neurosci*, 1998, p. 1164.
[19] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment,* vol. 2008, p. P10008, 2008.