Eurographics Workshop on Visual Computing for Biology and Medicine (2010)
D. Bartz, C. P. Botha, J. Hornegger, R. Machiraju, A. Wiebel, and B. Preim (Editors)

Graph averaging as a means to compare multichannel EEG coherence networks

Alessandro Crippa[1,2], Natasha M. Maurits[2,3] and Jos B. T. M. Roerdink[1,2]

[1] Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen, The Netherlands
[2] BCN Neuroimaging Center, University of Groningen, The Netherlands
[3] Department of Neurology, University Medical Center Groningen, University of Groningen, The Netherlands

Abstract

A method is proposed for quantifying differences between multichannel EEG coherence networks represented by functional unit (FU) maps. The approach is based on inexact graph matching for attributed relational graphs and graph averaging, adapted to FU maps. The mean of a set of input FU maps is defined in such a way that it not only represents the mean group coherence during a certain task or condition but also to some extent displays individual variations in brain activity. The definition of a mean FU map relies on a graph dissimilarity measure which takes into account both node positions and node or edge attributes. A visualization of the mean FU map is used with a visual representation of the frequency of occurrence of nodes and edges in the input FUs. This makes it possible to investigate which brain regions are more commonly involved in a certain task, by analysing the occurrence of an FU of the mean graph in the input FUs. Furthermore, our method gives the possibility to quantitatively compare individual FU maps by computing their distance to the mean FU map.

Categories and Subject Descriptors (according to ACM CCS): Data [E.1]: Graphs and networks—; Life and Medical Sciences [J.3]: Health—

1. Introduction

Nowadays, many neuroimaging methods are available to assess the functioning brain, such as functional Magnetic Resonance Imaging (fMRI), Positron Emission Tomography (PET), Electroencephalography (EEG) and Magneto-Encephalography (MEG). A recording with one of these imaging modalities provides a measurement of brain activity as a function of time and position. A more recent innovation is connectivity analysis, in which the anatomical or functional relation between different (underlying) brain areas is calculated [Fri94].

A promising approach is to study the resulting networks of interrelated brain regions. Of particular interest is the *comparison* of functional brain networks under different experimental conditions, or comparison of such networks between groups of subjects. Complex brain networks can be represented by graphs, in which nodes and links represent anatomical or functional units and their interdependencies, respectively. In the last decade a multitude of topological network measures has been developed [RS09, SR07, MKR*05] in an attempt to characterize and compare brain networks. However, such topological measures are calculated by thresholding, binarizing and symmetrizing the connectivity matrix of the weighted and directed brain network. Thus, spatial information is lost and only global network information is retained. Other network measures, in which weights and directionality information is retained, do exist, but still result in single numbers for the entire network.

The above approaches do not always yield the information that is necessary, and calls to go beyond network topology have recently appeared [FCP*09, SLL*09]. There is a clear need for methods to quantify differences in networks beyond global network properties. For interpretation and diagnosis it is essential that local differences can be visualized in the original network representation. This asks for the development of mathematical methods, algorithms and visualization tools for the *local comparison* of complex networks – not necessarily of the same size – obtained under different conditions (time, frequency, scale) or pertaining to different (groups of) subjects.

In this paper we propose a basis for such a local network comparison method for the case of EEG coherence networks. EEG is the oldest noninvasive functional neuroimaging technique. Electrodes, positioned on the scalp, record electrical activity of the brain. Synchronous electrical activity recorded in different brain regions is assumed to imply functional relations between those regions. A measure for this synchrony is EEG coherence, which is computed between pairs of electrode signals as a function of frequency [HRA*95, MSvdHdJ06]. Gaining insight in EEG data is getting more and more difficult as the improvements in EEG acquisition and computer simulation produce increasing amounts of numerical data. Visualization aids the interpretation of the results by transforming large quantities of data into visual representations.

A typical visualization of an EEG coherence dataset is a two dimensional graph layout (the EEG graph) where vertices represent electrodes and edges represent significant coherences between electrode signals. Vertices are usually represented as dots and edges as lines. For multichannel EEG (at least 64 electrodes) [KBS97, SRSP99] this layout suffers from a large number of overlapping edges and results in a cluttered layout. To solve the problem of cluttered visualizations, several solutions were proposed earlier. For instance, reorganizing the edges or varying the attributes of the edges without reducing their number can lead to less cluttered visualizations [WCG03, HMM00]. The positions of the vertices in the layout can also be reorganized [FR91], but in the case of EEG this is not appropriate, because the electrodes have meaningful positions as they relate to brain activity in specific areas.

Another approach to simplify the EEG graph is based on the selection of a small number of electrodes as markers. Each marker is supposed to be representative for all other electrodes in a certain region of interest (ROI), which are assumed to record similar signals because of volume conduction effects [KBS97, SPR*98, GLdJ06]. Several researchers have employed a hypothesis-driven selection of markers. This method neglects individual variations and does not make optimal use of the available information. An alternative is a data-driven approach where electrodes are grouped into functional units (FUs), which are defined as spatially connected cliques in the EEG graph, i.e., sets of electrodes that are spatially close and record pairwise significantly coherent signals [tCMR08]. A representation of the FUs in an EEG recording is called an FU map. This is a derived graph, in which the nodes represent FUs and are located at the barycenter of the electrodes in the FU, while edges connect FUs if the corresponding inter-FU coherence exceeds a threshold. To determine spatial relationships between electrodes, a Voronoi diagram is employed with one electrode in each Voronoi cell. FU maps can be used as a preprocessing step for conventional analysis.

In EEG research, several datasets are usually compared in a group analysis, for which several methods exist. Obviously, multiple FU maps can be compared visually when displayed next to each other, but this method is limited as humans are notoriously weak in spotting visual differences in images. Hence a more analytical method is required, which allows a quantitative comparison of different FU maps. This is a much more difficult problem than comparing "raw" EEG graphs, since in different FU maps the number of nodes and their positions are generally different, and there is no *a priori* correspondence between nodes in different FU maps.

The goal of this paper is to find a quantitative method for comparing several FU maps. Our method is based on inexact graph matching for attributed relational graphs [BA83] and graph averaging [BK00]. In our work we introduce a modification of the algorithm proposed in [BK00] to be able to compare multichannel EEG coherence data and to obtain a mean FU map, given a set of FU maps corresponding to different subjects or different experimental conditions. The mean FU map is defined in such a way that it not only represents the mean group response to a certain stimulus but also to some extent displays individual variations of brain activity. This makes it possible to investigate which brain regions are more commonly involved in a certain task, by analysing the occurrence of an FU of the mean graph in the input FUs. Furthermore, our method gives the possibility to quantitatively compare individual FU maps by computing their distance to the mean FU map.

Although our method was specifically designed for EEG coherence network comparison, we believe it to be of sufficient generality to be extended to other types of networks as well.

The main contributions of this paper can be summarized as follows:

- The definition of a graph dissimilarity measure for EEG functional unit maps, which takes into account both node positions and node or edge attributes;
- A definition of the mean of two attributed graphs representing FUs, following [BK00], and its extension to an arbitrary number of such graphs;
- An algorithm for computing the mean of a set of FU maps, with a quantitative measure of dissimilarity between this mean FU map and each of the input FU maps;
- Visualization of the mean FU map employing a visual representation of the frequency of occurrence of nodes and the average coherence between nodes in the input FUs.

2. Related Work

The principal concept in our approach is that of graph matching, that is, the problem to find a one-to-one mapping among the vertices of two graphs (graph isomorphism). This is a very challenging problem and several solutions are available in the literature. The standard algorithm for graph matching is by Ullman [Ull76]. Other algorithms

for the detection of maximum common subgraphs were proposed in [McG82, Lev72]. Graph matching is an NP-complete problem and thus exponential time is required to find an optimal solution. Approximate methods, with polynomial time requirements, are often used to find suboptimal solutions. Probabilistic approaches [CKP95, WH88], neural networks [FLD94, XO90], genetic algorithms [CWH96, WFH19], maximum flow methods [WZC94] and linear programming [Alm93] have been proposed for this purpose.

In many cases, exact graph matching is not possible, and one has to resort to inexact graph matching. Bunke and Allerman [BA83] proposed such a method for structural pattern recognition, where one has to find which of a set of prototype graphs most closely resembles an input graph. This requires some notion of graph similarity. They considered attributed relational graphs [TF79], where nodes and edges carry labels of the form (s,x) where s is the syntactic component and $x = (x_1,\ldots,x_n)$ is a semantic vector consisting of attribute values associated with s. Their similarity notion was defined in terms of graph edit operations (deletion, insertion, and substitution of nodes and edges) by which one graph can be (approximately) transformed to another one. The costs apply both to the syntactic and semantic part. The optimal inexact match was then defined as the inexact match with minimal graph edit distance. These notions were used by Bunke and Kandel [BK00] to define the *weighted mean* of a pair of graphs G, G' as a graph G'' such that $d(G,G'') = (1-\gamma)d(G,G')$ and $d(G'',G') = \gamma d(G,G')$, where $d(\cdot,\cdot)$ is the graph edit distance and $0 \leq \gamma \leq 1$. It was shown how to compute the weighted mean graph based on the algorithms for graph edit distance computation.

Another area in which graph comparison plays a role is that of graph animation. For example, Diehl et al. [DG02] consider drawing of dynamic graphs where nodes can be added or removed in the course of time. They proposed a foresighted layout method which considers all graphs to be drawn simultaneously and does not require a complete redraw of the graph after each update, thus preserving the mental map. This problem is simpler than ours since in graph animation a significant fraction of nodes and edges in different time frames do not change and can be identified *a priori*. So the graph matching problem does not arise here.

A different approach for comparing multiple FU maps for EEG coherence was proposed in [tCMR08]. First a mean EEG coherence graph was computed, i.e., the graph containing the mean coherence for every electrode pair computed across a group. This is possible in EEG analysis as the number and the positions of electrodes in different multichannel recordings are identical. Then an FU map was created for this mean EEG coherence graph just as for a single EEG graph. (Note that this mean-coherence FU map is not obtained by averaging FU maps, but by averaging coherences in the original EEG graph.) Such a mean-coherence FU map is meant to preserve dominant features from a collection of individual EEG graphs. Nevertheless, this approach has some drawbacks. Most importantly, individual variations are lost in such a map. Hence one still would have to visually compare individual FU maps to the mean-coherence FU map, and so the need for a quantitative method for comparing FU maps remains.

3. Methods

3.1. Matching of two attributed graphs

Given an EEG coherence graph, a functional unit (FU) represents a spatially connected set of electrodes recording pairwise significantly coherent signals (for the definition of significance, see [HRA*95]). The *intra-node coherence* of an FU is defined as the average of the coherences between the electrodes in the FU. Given two FUs, the *inter-node coherence* is the average of the coherences between all electrodes of the first FU and all electrodes of the second FU. FUs are displayed in a so-called *FU map* which preserves electrode locations. An example is given in Figure 2, where two FUs are connected by a link if the average coherence between them exceeds a threshold, which was set to 0.22, corresponding to a confidence level of 0.99 [tCMR08].

An FU map A can be represented as an attributed graph G_A, that is, a graph where nodes and edges are equipped with attributes. The nodes in this graph G_A correspond to FUs of A, and two nodes of G_A are connected by a link if the average coherence between the corresponding FUs exceeds the significance threshold. Each node m of G_A is equipped with the following information: (i) the set of electrodes of the FU corresponding to m; (ii) the position of the barycentre of these electrodes; (iii) the intra-node coherence of the FU corresponding to m. The weights of the edges between two nodes m and n of G_A represent the inter-node coherence between the two FUs of A corresponding to m and n.

The problem of comparison among FU maps is thus reduced to the comparison of attributed graphs. From now on, we will tacitly identify FUs of an FU map A and nodes of the attributed graph G_A representing these FUs (for example, when m is a node of G_A, instead of "electrodes of the FU corresponding to m" we will simply say "electrodes of m"). Also, by "graph" we will always mean "attributed graph". When m is a node in the graph G_A, the FU corresponding to m is denoted by $FU_{m,A}$, and an electrode i in this FU is referred to as $FU_{m,A}(i)$. Also, by the "position" of a node m we mean the position of the barycentre of the electrodes in $FU_{m,A}$.

Let A and B be two FU maps we intend to match. In general, the number of FUs in A will be different from that in B and also their positions could differ. Furthermore, the number of edges in A and in B, and their weights, are generally expected to be different. To be able to quantify the difference between A and B, our first goal is to find the best possible match between the nodes of A and those of B, i.e., to

determine which nodes of A correspond to which nodes of B. Secondly, given this match we quantify the difference between the two graphs by a dissimilarity measure, which is based on the matching of the two attributed graphs.

Definition 1 (**Matching of two graphs**). *Given a graph A with M nodes and a graph B with N nodes, where $M \leq N$, we call $\tilde{A}$ the* extension *of A obtained by adding $N-M$ nodes to A. A* matching *between A and B is a bijective function* match $: V_{\tilde{A}} \rightarrow V_B$ *which assigns any node of $\tilde{A}$ to a node of B and vice-versa.*

With a finite sequence of addition and shifting of nodes we can transform any attributed graph A to any other graph B via its extension $\tilde{A}$. Assigning a cost to each of these operations allows us to quantify the total cost of the transition from A to B. The assignment of such costs is obviously application dependent. Intuitively, in the case of an FU map comparison both the spatial position of nodes and the number of common electrodes between nodes in two different FU maps determine the costs. Therefore we use the following criteria for assigning costs.

Given a node m in graph A and a node n in graph B, we define their *spatial distance* $D(m,n)$ as the 2D Euclidean distance between their positions. Next, this distance is normalized to the interval $[0,1]$ by scaling it to the maximum possible distance in a FU map. Note that the position of the electrodes in an EEG is fixed between successive recordings, so measuring Euclidean distances of two points in two different FU maps is justified. We also define an *overlapping distance*, the Jaccard distance [Jac02], that describes dissimilarity of two FUs m and n according to the number of common electrodes. We recall here that for any two sets, their Jaccard distance is defined as one minus the cardinality of their intersection over the cardinality of their union. So,

$$J(m,n) = 1 - \frac{|FU_{m,A} \bigcap FU_{n,B}|}{|FU_{m,A} \bigcup FU_{n,B}|}$$

Note that $J(m,n) \in [0,1]$. Now we can define several costs related to node operations.

Definition 2 (**Cost of node operations.**) *The **cost of shifting a node** m in A to match a node n in B is defined as the weighted mean between their spatial distance $D(m,n)$ and their Jaccard distance $J(m,n)$.*

$$C^S_{m,n} = \lambda J(m,n) + (1-\lambda) D(m,n)$$

*The **cost of adding a node** $\tilde{m}$ to A is set to the maximum cost of 1.*
*The **total cost** of the matching of A to B is defined as the sum of the costs of the single operations applied to A.*

In our experiments, λ was set to 0.5.

It is easy to see that there is more than one sequence of operations that maps A to B. Since the solution is not unique, we define the *optimal matching* between A and B as the cheapest matching (lowest total cost) from the nodes of A to the nodes of B. If there exists more than one optimal matching one of the cheapest solutions is chosen arbitrarily. We give the pseudo-code for computing the optimal matching between two graphs A and B in Algorithm 1.

Definition 3 (**Dissimilarity measure between two graphs.**) *Given two graphs A and B, let A be the graph with the smallest number of nodes. The dissimilarity $\delta(A,B)$ between A and B is defined as the total cost of their optimal matching divided by the number of nodes of B.*

Algorithm 1 OPTIMAL MATCHING OF TWO GRAPHS

INPUT: graphs A, B with M and N nodes, $M \leq N$.
OUTPUT: optimal matching $\mathscr{M}^*$ and its cost $cost^*$.

```
extend A to Ã
for all m ∈ Ã do
  for all n ∈ B do
    c(m,n) = cost of the matching of nodes m and n
cost* ← +∞
for each possible matching M between Ã and B do
  cost ← cost of the matching M
  if cost ≤ cost* then
    cost* = cost
    M* = M
return M*, cost*
```

We have described how to match two graphs A and B and how to compute their dissimilarity. Given an optimal matching between A and B we can define their mean graph C.

3.2. Mean of two attributed graphs

We start from two FU maps represented by attributed graphs A and B with M and N nodes respectively, where we assume without loss of generality that $M \leq N$, and an optimal matching between the two. To make the definition general we allow that either A or B is already the result of an earlier graph averaging operation (we need this in section 3.3 below). Each electrode e in a graph A has an attribute *multiplicity*, denoted by $mult_A(e)$, which indicates how often the electrode occurs in the graph A. If A represents a single FU map then $mult_A(e) = 1$. If $mult_A(e) > 1$ this means that the same electrode e occurs in more than one of the graphs of which A is the average. Similarly, an additional node attribute *occurrence* is introduced, indicating how many times a node m occurs in a (possibly averaged) graph A; we write $occ_A(m)$ for this occurrence. If m is a node in a graph A corresponding to an individual FU map, we set $occ_A(m) = 1$.

Now we define the mean graph C, denoted by $C = [A,B]$, as follows.

1. If a node m in A matches a node n in B, the occurrence of the corresponding node k in C is computed by $occ_C(k) = occ_A(m) + occ_B(n)$, and the position of k is the average of the positions of m and n.

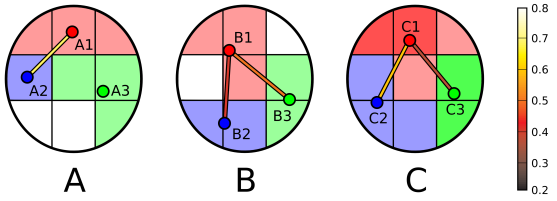

Figure 1: *Synthetic FU maps A and B are used to compute the average synthetic FU map C. Colours indicate different FUs. Each cell represents an electrode.*

2. If a node $\tilde{m}$ was added to A to match a node n in B, we set $occ_A(\tilde{m}) = 0$, so that the occurrence of the corresponding node k in C equals $occ_B(n)$, and we let the position of k be the position of n.
3. The intra-node coherence of a node k in C, corresponding to a node m in A matched to a node n in B, is defined as the average coherence between the electrodes in m and the electrodes in n (excluding electrodes which are common to m and n, i.e., self-coherences are not taken into account).
4. A node k in the graph C, corresponding to a node m in A matched to a node n in B, has as attribute the electrodes of m and the electrodes of n. The multiplicity of an electrode e is the sum of the multiplicities of e in A and in B: $mult_C(e) = mult_A(e) + mult_B(e)$. However, if an electrode e of m or n was already assigned to another node h of C in a previous step of the algorithm, then this conflict is resolved by (re)assigning electrode e to the node with the highest intra-node coherence (i.e., k or h).
5. The weight of an edge between nodes k and h of C is the average of the coherence between the electrodes of k and h which correspond to A, and the coherence between the electrodes of k and h which correspond to B.

The pseudo-code for the creation of the mean graph C is given in in Algorithm 2. Note that the graph average is a commutative operation, i.e., $[B,A] = [A,B]$.

The graph C is visualized in the same way as for the input FU maps A and B. That is, the nodes and edges are superimposed on the Voronoi diagram associated to electrode positions (which are common to A and B). Electrodes which do not belong to one of the input graphs A and B will be drawn as empty Voronoi cells. The result, when drawn in the plane in this way, will be referred to as the "mean FU map".

To illustrate how the average of two FU maps is computed, we show two synthetic FU maps A and B and their average C in Figure 1. In this example each synthetic FU map contains only 9 electrodes (note that the cells in which the electrodes are located are only drawn schematically, i.e., they are not real Voronoi cells). Only three FUs are present in each FU map: $A1$, $A2$ and $A3$ in A, and $B1$, $B2$ and $B3$ in B. Each FU has a different colour. Its barycenter is represented by a coloured circle, and its cells are coloured with a less saturated version of the same colour.

In C, we assume that the optimal matching matched $A1$ with $B1$, $A2$ with $B2$, and $A3$ with $B3$. We also see that because $A1$ and $B1$ have two electrodes in common, those are coloured with a more saturated red. The same holds for $A3$ and $B3$. The central electrode, belonging to $A3$ and to $B1$, was eventually assigned to $C1$ instead of to $C3$ because the intra-node coherence of $C1$ was higher than the intra-node coherence of $C3$.

Algorithm 2 MEAN OF TWO ATTRIBUTED GRAPHS

INPUT: graph A with M nodes and extension $\tilde{A}$, graph B with N nodes, $M \leq N$, and the optimal matching $\mathscr{M}^*$.
OUTPUT: mean FU map C

initialize an empty graph C
for all $n \in B$ **do**
  create a node k in C at the position of n
  $occ_C(k) \leftarrow occ_B(n)$
  $m \leftarrow \text{match}^{-1}(n)$ {m is the node matching to n}
  if $m \in A$ **then**
    $occ_C(k) \leftarrow occ_C(k) + occ_A(m)$
    move the position of k halfway between the position of m and n
    $intra\_coh_k \leftarrow$ average coherence between the electrodes in m and the electrodes in n
    for all electrodes e of m **do**
      for all electrodes e' of n **do**
        $mult_C(e) \leftarrow mult_C(e) + mult_A(e)$
        $mult_C(e') \leftarrow mult_C(e') + mult_B(e')$
        if e is is already assigned to a node $h \neq k$ of C **and** $intra\_coh_k > intra\_coh_h$ **then**
          reassign e to node k
        else
          assign e to node k
        if e' is is already assigned to a node $h \neq k$ of C **and** $intra\_coh_k > intra\_coh_h$ **then**
          reassign e' to node k
        else
          assign e' to node k
for each pair of nodes k, h in C, $k \neq h$ **do**
  weight of edge $(k,h) \leftarrow \frac{1}{2}$ (coherence between the electrodes of k and h which correspond to A + coherence between the electrodes of k and h which correspond to B)
return C

3.3. Generalized mean graph

When more than two subjects are involved in an EEG experiment the need of defining an average among several FU maps arises. Such an average can be defined as a direct extension of the average of two graphs previously defined.

First we extend the definition of the average of two attributed graphs A and B by including a weighting factor μ; we write $C = [A,B]_\mu$ for the weighted average graph. Item 1 and 5 in section 3.2 are adapted as follows. The position of a node k in C, resulting from the matching of a node m in A with a node n in B, is obtained by weighting the position of m by $1-\mu$ and the position of n by μ (line 10 of Algorithm 2). Accordingly, when computing the edge weights in line 25 of Algorithm 2, the FUs in A are weighted by $1-\mu$ and the FUs in B by μ.

Definition 4 (**Average of multiple attributed graphs.**) *Let $A_1, A_2, \ldots, A_n$ be n attributed graphs. The average $\hat{A}_n$ of these n graphs is recursively defined by:*

$$\begin{aligned} \hat{A}_2 &= [A_1, A_2]_{\frac{1}{2}} \\ &\vdots \\ \hat{A}_n &= [\hat{A}_{n-1}, A_n]_{\frac{1}{n}} \end{aligned} \tag{1}$$

This definition entails that for two graphs the weighting factor is $\frac{1}{2}$, i.e., equal weighting. But when the average graph is computed between $\hat{A}_{n-1}$, which itself is an average of $n-1$ graphs, and the last graph A_n, the former is weighted by $1-1/n$ and the latter by $1/n$.

Defining $\hat{c}_1, \ldots, \hat{c}_n$ as the costs of the matching corresponding to the computations of $\hat{A}_1, \ldots, \hat{A}_n$, the dissimilarity $\delta(A_1, A_2, \ldots, A_n)$ among the n graphs is defined as the mean of the costs $\hat{c}_i$.

Note that the result of the graph averaging operation defined in equation (1) depends on the order of the input graphs, i.e., it is not associative. This is due to the following. When the FUs corresponding to two nodes in different FU maps overlap, their common electrodes are assigned to the node with the highest intra-node coherence. Thus, when computing the graph average, nodes with low intra-node coherence could be reduced in size, or even disappear, depending on the order of processing.

Therefore, we consider all possible permutations of the n input graphs. Actually, we need only to consider half of all $n!$ permutations, since averaging two graphs is a commutative operation. The permutation P for which the dissimilarity $\delta(A_{P(1)}, A_{P(2)}, \ldots, A_{P(n)})$ is minimal is the *optimal* permutation and is used to compute the average graph.

4. Results

Five EEG data sets, recorded using 128 electrodes, were selected from a P300 experiment in which the participants had to count target tones of 2000 Hz, that were alternated with tones of 1000 Hz. The alpha frequency band (8-12 Hz) was considered for the computation of the FU maps (please refer to [tCMR08] for details).

Figure 2 shows the FU maps of two subjects A and B (out of the five), their mean FU map C, and the dissimilarities between A and C and between B and C. Figure 3 shows the FU maps of all five subjects. FU maps A and B of Figure 3 are the same as in Figure 2. Figure 4 shows the average of the FU maps shown in Figure 3, and Table 1 shows the dissimilarities between the FU maps in Figure 3 and their mean FU map. We see that FU map B, having only five FUs, has the highest dissimilarity to the mean FU map shown in Figure 4.

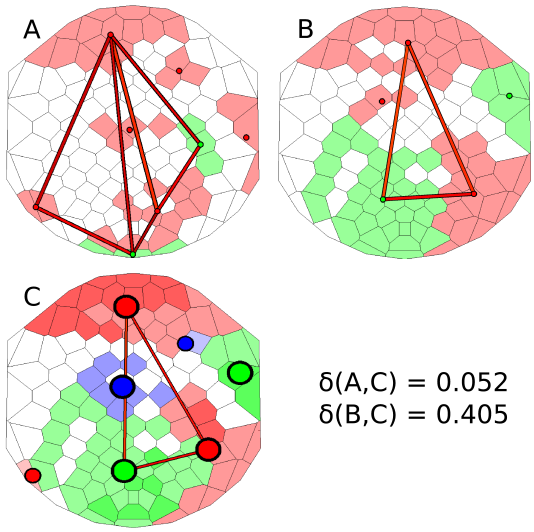

Figure 2: *Two FU maps, A and B, and their average FU map C. Voronoi cells represent electrodes. Spatial clusters of coloured cells correspond to FUs, white cells do not belong to any FU. Circles represent the barycentres of the FUs and are connected by edges whose colour indicates their inter-node coherence. In C, colour saturation is proportional to the multiplicity of a cell (electrode) in a graph node, and the size of the nodes reflects their occurrence in the input graphs. Only statistically significant edges are included. Dissimilarities between A/B and C are shown.*

Table 1: *Dissimilarities between the graphs shown in Figure 3 and their mean graph, shown in Figure 4.*

graph	*A*	*B*	*C*	*D*	*E*
δ	0.347	0.532	0.353	0.325	0.356

The visualization of the average graphs contains two types of information: the graph nodes and edges, and the Voronoi cells corresponding to the electrodes. Nodes are represented as circles and edges as line segments. The colours of the circles are based on a four-colouration of the graph. Cells are drawn in the same colour as the node they belong to, but in a less saturated version. The saturation is proportional to the multiplicity of a cell. White cells do not belong to any node. The size of a circle is proportional to the occurrence

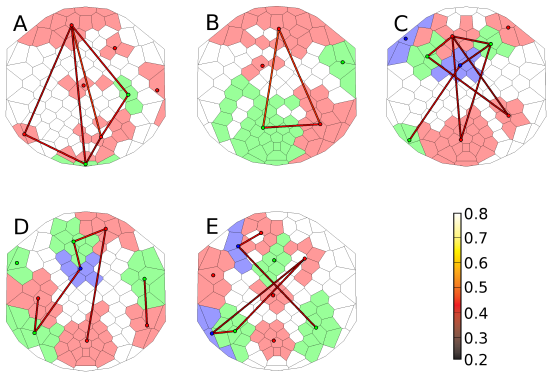

Figure 3: *FU maps of 5 subjects for the α frequency band. Voronoi cells represent electrodes. Spatial clusters of coloured cells correspond to FUs, white cells do not belong to any FU. Circles represent the barycentres of the FUs and are connected by edges whose colour indicates their inter-node coherence, according to the shown colour map. Only statistically significant edges are shown.*

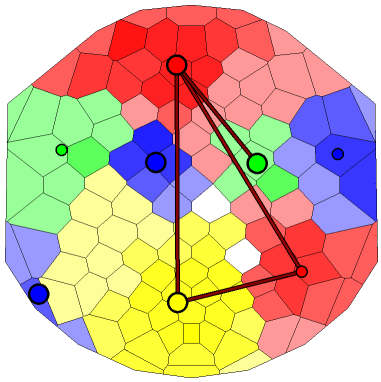

Figure 4: *Average graph of the FU maps shown in Figure 3. For explanation see the caption of Figure 2.*

of that node in the input graphs. That is, when computing the mean among several graphs this size will indicate how many of the input graphs the node belongs to. The edges of the graph represent the statistically significant [HRA*95] coherences between pairs of nodes; the coherence value is mapped to the colour of the edges. Note that the mean FU map differs from an ordinary FU map by the visual enrichments related to node occurrence and cell multiplicity, which represent variations of the input FUs.

Given the usually small number of nodes in the input graphs, computing the optimal matching can be achieved using brute force. The computational time requirements of the exploration of all the possible matchings are $\mathcal{O}(N!)$ with N the maximum number of nodes in A and B, and for $N = 10$ it can be performed in roughly 10 s on a modern PC. The determination of the generalized average graph is achieved by evaluating all possible permutations of the graphs. The total time complexity is thus $\mathcal{O}(n!N!)$ with n the number of graphs. Computing the average of the 5 graphs in Figure 3 took roughly 3 min.

5. Conclusions

In this paper, we proposed a method based on inexact graph matching for quantifying differences between multichannel EEG coherence networks represented by functional unit maps. We defined a class of cost functions to compute the mean of two attributed graphs representing FU maps of two subjects and extended the notion of mean graph to the case with multiple subjects. A visualization of the mean FU map was used with a visual representation of the frequency of occurrence of nodes and edges in the input FUs. A feature of our method is the possibility to localize those FUs which are common among all subjects. This possibly reflects which brain areas are mostly involved in certain tasks.

Our focus in this paper has been on demonstrating a proof of concept, not on obtaining a fast implementation. Some of the algorithms in our method perform exhaustive search and have time requirements which are exponential in the number of FUs in the input graphs. For the limited size of the input data we used here the computation of mean FU maps can still be achieved in a reasonable amount of time, i.e., in the order of minutes. Nevertheless our method would be slow when a much larger number of subjects is included in the study. In such a scenario, a heuristic search approach with polynomial time requirements (cf. section 2) should be considered instead.

Future work will involve a detailed quantitative comparison of EEG coherence networks based on the similarities and dissimilarities between individual FU maps and the mean FU map. Studies with focused experiments will be necessary for the validation of the new method.

References

[Alm93] ALMOHAMED H.: A linear programming approach for the weighted graph matching problem. *IEEE Trans. Pattern Anal. Machine Intell. 15* (1993), 522–525.

[BA83] BUNKE H., ALLERMANN G.: Inexact graph matching for structural pattern recognition. *Pattern Recogn Lett 1* (1983), 245–253.

[BK00] BUNKE H., KANDEL A.: Mean and maximum common subgraph of two graphs. *Pattern Recogn Lett 21*, 2 (February 2000), 163–168.

[CKP95] CHRISTMAS W. J., KITTLER J., PETROU M.: Structural matching in computer vision using probabilistic relaxation. *IEEE Trans. Pattern Anal. Machine Intell. 8* (1995), 749–764.

[CWH96] CROSS A., WILSON R., HANCOCK E.: *Computer Vision - ECCV LNCS*, vol. 1064. Springer Verlag, 1996, ch. Genetic search for structural matching, pp. 514–525.

[DG02] DIEHL S., GÖRG C.: Graphs, they are changing. In *Graph Drawing* (2002), pp. 23–30.

[FCP*09] FAIR D. A., COHEN A. L., POWER J. D., DOSENBACH N. U. F., CHURCH J. A., MIEZIN F. M., SCHLAGGAR B. L., PETERSEN S. E.: Functional brain networks develop from a "local to distributed" organization. *PLoS Comput Biol 5*, 5 (05 2009), e1000381.

[FLD94] FENG J., LAUMY M., DHOME M.: *Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems*. North Holland, 1994, ch. Inexact matching using neural networks, pp. 177–184.

[FR91] FRUCHTERMAN T. M. J., REINGOLD E. M.: Graph drawing by force-directed placement. *Software - Practice and Experience 21*, 11 (1991), 1129–1164.

[Fri94] FRISTON K. J.: Functional and effective connectivity in neuroimaging: A synthesis. *Human Brain Mapping 2* (1994), 56–78.

[GLdJ06] GLADWIN T. E., LINDSEN J. P., DE JONG R.: Pre-stimulus EEG effects related to response speed, task switching and upcoming response hand. *Biol. Psychol. 72*, 1 (2006), 15–34.

[HMM00] HERMAN I., MARSHALL M. S., MELANÇON G.: Density functions for visual attributes and effective partitioning in graph visualization. In *Proc. IEEE Symposium on Information Vizualization* (2000), pp. 49–56.

[HRA*95] HALLIDAY D. M., ROSENBERG J. R., AMJAD A. M., BREEZE P., CONWAY B. A., FARMER S. F.: A framework for the analysis of mixed time series point process data - theory and application to the study of physiological tremor, single motor unit discharges and electromyograms. *Prog. Biophys. Mol. Bio. 64*, 2/3 (1995), 237–278.

[Jac02] JACCARD P.: Lois de distribution florale dans la zone alpine. *Bull. Soc. Vaud. Sci. Nat. 38* (1902), 69–130.

[KBS97] KAMINSKI M., BLINOWSKA K., SZELENBERGER W.: Topographic analysis of coherence and propagation of EEG activity during sleep and wakefulness. *Electroen. Clin. Neuro. 102* (1997), 216–227.

[Lev72] LEVI G.: A note on the derivation of maximal common subgraphs of two directed or undirected graphs. *Calcolo 9* (1972), 341–354.

[McG82] MCGREGOR J.: Backtrack search algorithms and the maximal common subgraph problem. *Software-Practice and Experience 12* (1982), 23–34.

[MKR*05] MORMANN F., KREUZ T., RIEKE C., ANDRZEJAK R. G., KRASKOV A., DAVID P., ELGER C. E., LEHNERTZ K.: On the predictability of epileptic seizures. *Clin Neurophysiol 116* (2005), 569–597.

[MSvdHdJ06] MAURITS N. M., SCHEERINGA R., VAN DER HOEVEN J. H., DE JONG R.: EEG coherence obtained from an auditory oddball task increases with age. *J Clin Neurophysiol 23*, 5 (2006), 395–403.

[RS09] RUBINOV M., SPORNS O.: Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage In Press, Corrected Proof* (2009), –.

[SLL*09] SHU N., LIU Y., LI J., LI Y., YU C., JIANG T.: Altered anatomical network in early blindness revealed by diffusion tensor tractography. *PLoS One 4*, 9 (Sep 28 2009), e7228.

[SPR*98] SARNTHEIN J., PETSCHE H., RAPPELSBERGER P., SHAW G. L., VON STEIN A.: Synchronization between prefrontal and posterior association cortex during human working memory. *P Natl Acad Sci USA 95*, 12 (1998), 7092–6.

[SR07] STAM C. J., REIJNEVELD J. C.: Graph theoretical analysis of complex networks in the brain. *Nonlinear Biomedical Physics 1* (2007).

[SRSP99] STEIN A. V., RAPPELSBERGER P., SARNTHEIN J., PETSCHE H.: Synchronization between temporal and parietal cortex during multimodal object processing in man. *Cereb Cortex 9* (1999), 137–150.

[tCMR08] TEN CAAT M., MAURITS N. M., ROERDINK J. B. T. M.: Data-driven visualization and group analysis of multichannel EEG coherence with functional units. *IEEE Trans. Visual. Comput. Graphics 14*, 4 (2008), 756–771.

[TF79] TSAI W.-H., FU K.-S.: Error-correcting isomorphisms of attributed relational graphs for pattern analysis. *IEEE Trans. Syst., Man, Cybern. 9*, 12 (1979), 757–768.

[Ull76] ULLMANN J. R.: An algorithm for subgraph isomorphism. *J. ACM 23*, 1 (1976), 31–42.

[WCG03] WONG N., CARPENDALE S., GREENBERG S.: EdgeLens: An interactive method for managing edge congestion in graphs. In *Proc. IEEE Symposium on Information Visualization* (2003), pp. 51–58.

[WFH19] WANG Y.-K., FAN K.-C., HORNG J.-T.: Genetic-based search for error-correcting graph isomorphism. *IEEE Trans. Syst., Man, Cybern. B 27*, 4 (19), 588–597.

[WH88] WILSON R., HANCOCK E.: *Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems*, vol. 10. North Holland, 1988, ch. Graph matching by discrete relaxation, pp. 165–176.

[WZC94] WANG I., ZHANG K., , CHIRN G.: The approximate graph matching problem. In *Proc. of 12th ICPR* (1994), pp. 284–288.

[XO90] XU L., OJA E.: *Improved simulated annealing, Boltzmann machine, and attributed graph matching*, vol. 412 of *Lecture Notes in Computer Science*. Springer Verlag, 1990, pp. 151–161.