Coclustering Based Parcellation of Human Brain Cortex Using Diffusion Tensor MRI

Cui Lin[1], Shiyong Lu[1], Danqing Wu[1], Jing Hua[1], and Otto Muzik[2]

[1] Department of Computer Science, Wayne State University
{ cuilin, shiyong, dqwu, jinghua } @wayne.edu
[2] PET Center at Children's Hospital of Michigan, Radiology at Wayne State University
otto@pet.wayne.edu

Abstract. The fundamental goal of computational neuroscience is to discover anatomical features that reflect the functional organization of the brain. Investigations of the physical connections between neuronal structures and measurements of brain activity in vivo have given rise to the concepts of anatomical and functional connectivity, which have been useful for our understanding of brain mechanisms and their plasticity. However, at present there is no generally accepted computational framework for the quantitative assessment of cortical connectivity. In this paper, we present accurate analytical and modeling tools that can reveal anatomical connectivity pattern and facilitate the interpretation of high-level knowledge regarding brain functions are strongly demanded. We also present a coclustering algorithm, called Business model based Coclustering Algorithm (BCA), which allows an automated and reproducible assessment of the connectivity pattern between different cortical areas based on Diffusion Tensor Imaging (DTI) data. The proposed BCA algorithm not only partitions the cortical mantel into well-defined clusters, but at the same time maximizes the connection strength between these clusters. Moreover, the BCA algorithm is computationally robust and allows both outlier detection as well as operator-independent determination of the number of clusters. We applied the BCA algorithm to human DTI datasets and show good performance in detecting anatomical connectivity patterns in the human brain.

1 Introduction

With ever-improving imaging technologies, the complexity and scale of brain imaging data has continued to grow at an explosive pace. Recent advances in imaging technologies, especially that of Diffusion Tensor Imaging [1–3], have allowed an increased understanding of normal and abnormal brain structure and function [4, 3]. It is well understood that normal brain function is dependent on the interactions between specialized functional areas of the brain which process information within local and global networks. Perhaps the most promising approach to parcelate the cerebral cortex into such distinct functional areas originates from the notion that functionally discrete areas of the cortical mantel are characterized by cortico-cortical connectivity patterns, which represent functionally integrated neural subsystems and determine the region's functional properties [5] and also allow their anatomical delineation and mapping.

At present, no generally accepted parcellation scheme exists for the human cortex, although circumstantial evidence points to a distinct arrangement of functional territories within the cortex. As illustrated in Figure 1, most cortical voxels in one region are strongly connected to a particular region of the cortex and the connections to any other regions are relatively weaker. For example, most voxels on the top of the cortex congregating in cortical region C_2 are connected to voxels of cortical region C_1, with only few connections to other cortical regions. Therefore, in order to perform an accurate in-vivo analysis of the cortico-cortical connectivity, what is needed is a partitioning procedure that not only simultaneously partitions voxels into groups, but also identifies the corresponding strong connectivities between the two classes of groups.

Traditional clustering algorithms [6–9] are suboptimal in incorporating anatomical constraints and as a result will fail to identify accurately the corresponding connectivity between cortical regions. Moreover, our focus in this paper is to assess the neural connections within a hemisphere (intra-hemispheric connections) sine these connections are relatively weak compared to the connections between the left and right hemisphere, but represent crucial neural pathways which are abnormal in neurological disease. Consequently, we consider only clusters which connect cortical areas within one hemisphere.

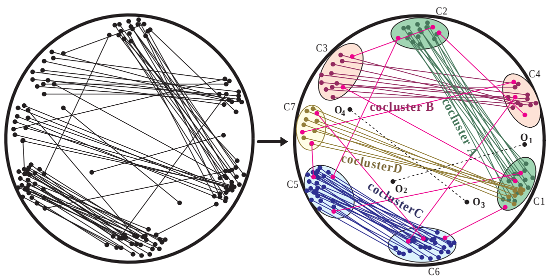

Fig. 1. The coclustering process

The main contributions of this paper are:

1. We are the first to propose a new coclustering model for defining cortico-cortical connectivity analysis as a computational problem.
2. Our BCA coclustering algorithm is able to define functional cortical areas based on cortico-cortical fiber tract connections taking into account anatomical constraints. In contrast to traditional clustering paradigms, the BCA algorithm is not only able to partition image voxels within the cortical mantel into well-defined clusters, but also is able to maximize the connectivity strength between such clusters. Moreover, the BCA method is able to identify outliers as well as the number of cortical clusters with high efficiency.
3. The application of the BCA algorithm to human DTI dataset allows automated and reproducible assessment of the connectivity patterns in the human brain.

Organization. The rest of the paper is organized as follows: Section 3 formalizes the coclustering model for the cortico-cortical connectivity analysis. Section 4 proposes our coclustering algorithm, BCA, in order to assess fiber tract connectivity between remote cortical areas. Section 5 presents 3-D visualization of the obtained results followed by a discussion of the application in patient groups. Finally, Section 6 concludes the paper and comments on future work.

2 Background and Related Work

The cerebral cortex sends connections (efferents) and receives connections (afferents) from many subcortical structures, but the largest part of the connections arriving at the cerebral cortex comes from the cerebral cortex itself. Assessing connectivity patterns of cortico-cortical fiber tracts is important for our understanding of the mechanisms involved in human brain functions and might provide clues towards the identification and characterization of many neurological diseases.

Recently, Diffusion Tensor Imaging (DTI) tractography has been shown to produce results that are consistent with known pathways formed by major white matter fiber tracts in the human brain [10, 2], although limitations in data acquisition and processing algorithms [11] related to clinical constraints produce data which cannot resolve crossing or intersecting fibers. DTI is based upon the ability of MRI to evaluate in vivo the direction and magnitude of water diffusion in tissues [2]. These attributes of in vivo water diffusion depend upon microscopic tissue architecture [12]. Therefore, changes in these parameters serve as markers for changes in tissue micro-architecture. The principal eigenvector obtained from DTI provides information about the preferential direction of water diffusion in each imaging voxel. This direction corresponds to the direction of the nerve fiber bundles, which predominantly constitute the given voxel. Hence, different nerve fiber bundles can be identified and used to assess the integrity of white matter tracts throughout the brain [13, 12]. Despite some success in delineating functional cortical areas using DTI, a systematic framework allowing functional parcellation of the neocortex based on quantitative assessment of fiber tract connectivity has not yet been produced, and the relationship among cortical territories, fiber tracts, and neuronal connections remains controversial. Consequently, there is a need to further develop advanced clustering algorithms that allow better characterization of brain connectivity patterns and as a result improve our understanding of process interactions in a complex biological system.

Traditional partitioning relocation clustering algorithms, such as the K-means [6], K-medoids [7] are simple and efficient, however, their final results may be overly sensitive to the initial cluster set and the presence of outliers. In addition, it is difficult to implement when no information exists about the likely cluster number. Hierarchical clustering algorithms [8, 14] do not require the number of clusters K as input, but they require a termination condition. In addition, they do not support reclassification of objects to new clusters. Density-based algorithms [9, 15, 16] have good performance with respect to noise handling and one-scan efficiency, but are suboptimal for the cortico-cortical problem, as they do not consider the connectivity strength between clusters, hence fail to identify accurately the corresponding strongest connectivity between cortical regions.

Even though our first coclustering algorithm GCA [17] was effective in the analysis of thalamo-cortical connectivity, it is not directly applicable to the cortico-cortical connectivity problem as each fiber connects to two different cortical voxels. Direct application of the GCA algorithm to cortico-cortical connectivity analysis might lead to the following undesirable results: (1) the same voxel can be classified simultaneously into several clusters, (2) two end voxels of a fiber tract might be classified into the same cluster, and (3) two partitions of voxels given by GCA might be inconsistent and the resolution of this inconsistency is not obvious.

3 The Coclustering Model

In this section, we present our coclustering model, which models the cortico-cortical connectivity problem. The structure of the cerebral cortex and its cortical connections is initialized as a graph $G(V, F)$, as illustrated in Figure 1, where V represents all cortical voxels, and F represents all of the cortico-cortical connections between each other.

Definition 1 (Outlier)**.** *Given a partition C of $G(V, F)$, $C = \{C_1, C_2, \cdots, C_K\}$, an outlier is defined as:*

$$o = \{v | v \in V, \forall C_i \in C, v \notin C_i\}$$

Definition 2 (Connection strength $\theta(C_i, C_j)$)**.** *Given a cortical cluster C_i and C_j, the connection strength between C_i and C_j is defined as:*

$$\theta(C_i, C_j) = \frac{N_{ij}}{|C_i|}$$

where N_{ij} equals to the total number of connections between C_i and C_j.

Definition 3 (Spouse cluster)**.** *Given a partition P of $G(V, F)$, $P = \{C_1, C_2, ..., C_k\}$, $SP(C_i) = \{C_j | \forall C_k \in P, \theta(C_i, C_j) \geq \theta(C_i, C_k)\}$*

Definition 4 (Cocluster set)**.** *$< C_i, C_j >$ is called a* cocluster set *iff C_j is the spouse cluster of C_i.*

Example 1. In Figure 1, C_2 is C_1's spouse cluster, they form a cocluster set $coclusterA < C_1, C_2 >$, $< C_7, C_1 >$ forms another cocluster set $coclusterD$. worth mentioning, the two elements in a cocluster set are not commutative. thus, $< C_i, C_j >$ does not necessarily implies $< C_j, C_i >$. For instance, $< C_1, C_7 >$ is an invalid cocluster set, because only C_2 can be identified as C_1's spouse cluster.

The goal of our coclustering procedures is to partition objects into groups while minimizing the cross-connectivity costs between those groups. More specifically, the coclustering procedures will separate objects into K groups so that (1) similar objects are within the same group, while dissimilar objects are in different groups, (2) there is a one-to-one correspondence /one-to-many correspondence between one cortical cluster to another / other clusters; and (3) the total cross-connectivity cost between each cluster and its non-spouse cluster is minimized.

To achieve the above goals, we define several notions. First, we define the centroid of a cluster and its Within-Cluster Variation (WCV) to quantify the similarity of objects within one cortical cluster.

Definition 5 (Centroid)**.** *Given a cortical cluster C_k, its centroid $\overrightarrow{\mu_k}$ is defined as:*

$$\overrightarrow{\mu_k} = \frac{\sum_{\overrightarrow{X_n} \in C_k} \overrightarrow{X_n}}{|C_k|}$$

where $|C_k|$ represents the number of cortical voxels in cluster C_k.

Definition 6 (WCV)**.** *We define Within-Cluster Variation of cortical cluster C_k as:*

$$WCV(C_k) = \sum_{\overrightarrow{X_n} \in C_k} d(\overrightarrow{X_n}, \overrightarrow{\mu_k})$$

where $d(\overrightarrow{X_n}, \overrightarrow{\mu_k})$ is the Euclidean distance between the cortical voxel $\overrightarrow{X_n}$ and the centroid $\overrightarrow{\mu_k}$ of cortical cluster C_k.

Second, we define the Total Within-Cluster Variation(*TWCV*) to quantify the quantity of a particular partitioning.

Definition 7 (TWCV)**.** *The Total Within-Cluster Variation of a cortical partition ($C_1, \cdots, C_K$) is defined as*

$$\begin{aligned} & TWCV(C_1, \cdots, C_K) \\ = & \sum_{k=1}^{K} WCV(C_k) \\ = & \sum_{k=1}^{K} \sum_{\overrightarrow{X_n} \in C_k} \sum_{d=1}^{D} (X_{n_d} - \mu_{k_d})^2 \\ = & \sum_{k=1}^{K} \sum_{d=1}^{D} {X_{n_d}}^2 - \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{d=1}^{D} (SCF_{k_d})^2 \end{aligned}$$

where SCF_{k_d} is the sum of the dth feature of all voxels in C_k.

Third, in order to minimize the cross-connectivity cost, for each cortical cluster, we define the set of cortical voxels that are connected to it as its *shaded cortical cluster*.

Definition 8 (Shaded cluster)**.** *Given a cortical partition ($C_1, \cdots, C_K$), the shaded cluster SC_k ($k = 1, \cdots, K$) is defined as:*

$$SC_k = \{sc | sc \in C, \forall v \in C_k, \exists v' \in C, (v', v) \in F, sc \cap C_k = \emptyset\}$$

Example 2. In Figure 2, all the cortical voxels that are connected to voxels in cortical cluster C_2 forms the shaded cluster SC_1, while all voxels that are connected to the voxles in cortical cluster C_1 forms the shaded cluster SC_2.

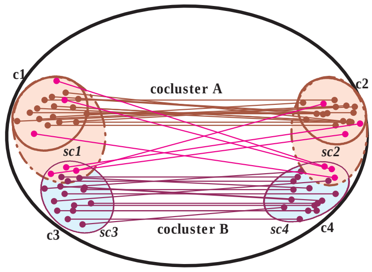

Fig. 2. Shaded clusters

In an ideal coclustering, as $CoclusterB$, a shaded cluster should *coincide* with the corresponding spouse cluster. However, this is not always the case in general. The cross-connectivity cost can be characterized by the disagreement between shaded clusters and spouse clusters and quantified by the Within-Cluster Variance of shaded clusters with respect to their corresponding spouse clusters, called *Shaded Within-Cluster Variation(SWCV)*, that is defined as follows.

Definition 9 (SWCV)**.** *The Shaded Within-Cluster Variation(*SWCV*) of cortical cluster SC_k is defined as:*

$$SWCV(SC_k) = \sum_{\overrightarrow{X'_n} \in SC_k} d(\overrightarrow{X'_n}, \overrightarrow{\mu_k})$$

Note that, instead of using the centroid of SC_k , the centroid of the C_k is used to calculate $SWCV(SC_k)$. The intuition is that, in an ideal partitioning, the shaded partition $SC_1, \cdots, SC_K$ should mostly coincide with $C_1 \cdots C_K$.

Definition 10 (STWCV)**.** *The Shaded Total Within-Cluster Variation ($STWCV$) of cortical partition ($SC_1, \cdots, SC_K$) is defined as:*

$$\begin{aligned} STWCV(SC_1, \cdots, SC_K) &= \sum_{k=1}^{K} SWCV(SC_k) \\ &= \sum_{k=1}^{K} \sum_{\overrightarrow{X'_n} \in SC_k} \sum_{d=1}^{D} (X'_{n_d} - \mu_{k_d})^2 \end{aligned}$$

The variance in distances between voxels is partitioned into variance attributable to differences among distance within clusters and to differences among clusters. $WCV(C_k)$ measures the variability within the cluster C_i, while we introduce $BCV(C_k)$as a measure of the variability between cortical clusters.

Statement of the problem. Finally, the coclustering problem can be formally stated as follows: given a Graph $G = (V, F)$ and a distance metric d for nodes between v_i and $v_j (i \neq j)$, coclustering is required to partition V into K clusters and cocluster sets, as well as a set of outliers, formulated as $\{< C_1, SP(C_1) >, \cdots, < C_K, SP(C_K) >$

$, O\}$, such that the connection strength of each cluster is maximized and the following objective function $OTWCV$ is minimized:

$$OTWCV(C_1, C_2, \cdots, C_K) = \sum_{k=1}^{K} TWCV(C_k) + STWCV(SC_k)$$

4 The BCA Algorithm

In this section, we propose our coclustering algorithm, *Business model based Coclustering Algorithm (BCA)*, to solve the coclustering problem.

BCA starts with the density-based initialization, and produces a better solution from the current solution by applying the following three phases, viz. *Split*, *Transfer*, and *Merge* sequentially. Three procedures can run iteratively to produce one solution after another until a termination condition is reached. During each iteration, the current solution S_i is associated with the figure of merits that include a function of $OTWCV$ and the connection strength.

The goal of our algorithm's initialization is to not only partition cortical voxels into cocluster sets, but also to minimize the distance variance within one cluster while maximizing each cluster's connection strength.

4.1 Density-based initialization

The goal of our density-based initialization is to have an initial clustering of the cortical voxels based on the following working hypothesis provided by our domain experts: voxels within one functional cortical region should be close to each other and each functional cortical region should contain at least one dense subregion. The initialization procedure is described by Algorithm *Initialize* in Figure 3. The algorithm takes a cortico-cortical connectivity graph G and two parameters ε and δ as input and produces an initial coclustering as output. In addition, in the output, a set of voxels O will be identified as *outliers* that will not be classified into any functional cortical region. While ε is the maximum radius of a voxel's neighborhood, δ is the minimum number of voxels within the ε-neighborhood of a voxel for the voxel to be a core voxel. We first introduce the following notions.

Definition 11 (ε-neighborhood and core voxel)**.** *Given a cortico-cortical connectivity graph $G(V, F)$, the ε-neighborhood of a voxel $v \in V$, denoted $N_\varepsilon(v)$, is defined by $N_\varepsilon(v) = \{u \in V \mid dist(v, u) \leq \varepsilon\}$. We call v a core voxel iff $|N_\varepsilon(v)| \geq \delta$.*

Definition 12 (Distance-reachable)**.** *A voxel u is directly distance-reachable from a voxel v w.r.t. ε and δ if $u \in N_\varepsilon(v)$ and u is distance-reachable from v if there is a chain of voxels $v_1, \cdots, v_n$, such that $v_1 = v, v_n = u$, and v_k is directly distance-reachable from v_{k-1} for $k = 2, \cdots, n$.*

As shown in Figure 3, Algorithm *Initialize* firstly calculates $N{*}N$ *distance-connection matrix* M to store the Euclidean distances between each pair of cortical voxels (line 5).

```
(1)  Algorithm: Initialize
(2)  Input: Cortico-cortical connectivity graph G(V, F), maximum radius ε, and minimum number of voxels δ
(3)  Output: initial coclustering {< C_1, SP(C_1) >, ··· , < C_K, SP(C_K) >, O}
(4)  Begin
(5)      Calculate from G the distance-connection matrix M to store dist(u, v) for all u, v ∈ V;
(6)      k = 0;
(7)      For each voxel v ∈ V do
(8)          If v is classified then
(9)              Process the next voxel;
(10)         Else /* v is not classified */
(11)             If v is a core voxel then
(12)                 k := k + 1;
(13)                 Collect all voxels distance-reachable from v and assign them to C_k
(14)             Else
(15)                 Process the next voxel;
(16)             End If
(17)         End If
(18)     End For
(19)     Collect all unclassified voxels and assign them to O ;
(20)     Identify SP(C_k)(k = 1...K) according to Definition 3.
(21) End Algorithm
```

Fig. 3. Algorithm Initialize

We define $dist(u, v) = +\infty$ iff u and v belong to different hemispheres of the brain to implement the constraint that each resulting cluster will not span across different hemispheres. The algorithm will iteratively consider each voxel v (lines 7 - 18). If v is an unclassified voxel, then a new cluster is formed by all the voxels that are distance-reachable from v; otherwise, either v is already classified (lines 8-9), or it is a non-core voxel (lines 14-15), the processing of v will be skipped. After the iteration completes, all the unclassified voxels will be assigned to a set O as outliers. Finally, for each identified cluster C_k ($k = 1, \cdots, K$), its spouse cluster $SP(C_k)$ will be identified according to Definition 3 (line 20) to produce the initial coclustering result.

Since the analysis performed in the initialization procedure focuses on region density and distances between cortical voxels rather than their connectivity, the BCA algorithm further applies operators Split, Transfer, and Merge iteratively to improve the coclustering result by minimizing its OTWCV value.

4.2 Split

The split operator attempts to split a cluster into two clusters when such a split will improve the result of coclustering that is characterized by the following split condition.

Definition 13 (Split condition)**.** *Given a coclustering* $CO = \{< C_1, SP(C_1) >, \cdots, < C_i, SP(C_i) >, \cdots, < C_K, SP(C_K) >\}$ *and a cluster* $C_i \in CO$*, let* C_{i1} *be the set of voxels in* C_i *that are connected to* $SP(C_i)$*,* C_{i2} *be* $C_i - C_{i1}$*, and* $CO' = \{< C_1, SP(C_1) >, \cdots, < C_{i1}, SP(C_{i1}) >, < C_{i2}, SP(C_{i2}) >, \cdots, < C_K, SP(C_K) > \}$*, then we say that* C_i *satisfies the split condition iff*
1) $|C_{i1}| >= \delta$ *and* $|C_{i2}| >= \delta$*;*
2) $OTWCV(CO') \leq OTWCV(CO)$*;*
3) $\theta(C_1, SP(C_1)) \leq \theta(C_{i2}, SP(C_{i2}))$*.*

Algorithm: Split
Input: $CO = \{< C_1, SP(C_1) >, \cdots, < C_K, SP(C_K) >\}$
Output: a new version of CO in which no more cluster satisfies the split condition
Begin
  While there exists a cluster $C_i \in CO$ satisfying the split condition **do**
    Split C_i into C_{i1} and C_{i2};
    Recalculate the spouse cluster for each cluster in CO according to Definition 3;
  End while
End Algorithm

Fig. 4. Algorithm Split

Intuitively, the split condition ensures that after a split, 1) the number of voxels in each new cluster is still greater than or equal to δ, 2) the OTWCV value for the new coclustering will not increase, and 3) the connection strengths of the two new clusters C_{i1} and C_{i2} will be no less than the connection strength of the original cluster C_i. This is always true for C_{i1}, and thus we only need to require $\theta(C_1, SP(C_1)) \leq \theta(C_{i2}, SP(C_{i2}))$ in the above definition of the split condition.

Algorithm *Split* is sketched in Figure 4. Basically, it iteratively splits the colustering result until no more cluster satisfies the above defined split condition.

4.3 Transfer

The transfer operator attempts to reassign each voxel to a new cluster in order to improve the result of coclustering that is characterized by the following transfer condition.

Definition 14 (Transfer Condition)**.** *Given a coclustering* $CO = \{< C_1, SP(C_1) >, \cdots, < C_i, SP(C_i) >, \cdots, < C_j, SP(C_j) >, \cdots, < C_K, SP(C_K) >\}$*, let* $v \in C_i$ *for some* C_i *in CO,* C_j *be the cluster to whose centroid* v *is the closest, after transferring* v *from* C_i *to* C_j*,* C_i *becomes* C'_i*,* C_j *becomes* C'_j*, and CO becomes* $CO' = \{< C_1, SP(C_1) >, \cdots, < C'_i, SP(C'_i) >, \cdots, < C'_j, SP(C'_j) >, \cdots, < C_K, SP(C_K) >\}$*, we say that* v *satisfies the transfer condition iff*
1) $|C'_i| >= \delta$*;*
2) $OTWCV(CO') \leq OTWCV(CO)$*;*
3) $\theta(C_i, SP(C_i)) \leq \theta(C'_i, SP(C'_i))$ *and* $\theta(C_j, SP(C_j)) \leq \theta(C'_j, SP(C'_j))$.

Algorithm: Transfer
Input: $CO = \{< C_1, SP(C_1) >, \cdots, < C_K, SP(C_K) >\}$
Output: new version of CO in which no more voxel satisfying the transfer condition
Begin
  While there exists a voxel $v \in C_i$ satisfying the transfer condition **do**
    Transfer v from C_i to C_j where C_j is the cluster to whose centroid v is the closest;
    Recalculate the spouse cluster for each cluster in CO according to Definition 3;
    End while
End Algorithm

Fig. 5. Algorithm Transfer

Intuitively, the transfer condition ensures that after a transfer, 1) C_i still contains at least δ voxels, 2) the OTWCV value for the new coclustering will not increase, and 2) the connection strengths of the two affected clusters will not decrease. Algorithm *Transfer* is sketched in Figure 5. Basically, it attempts to assign each voxel to a new cluster if it satisfies the transfer condition. The procedure terminates when no more voxel satisfies the above defined transfer condition.

4.4 Merge

Finally, the merge operator attempts to merge two clusters if such a merge will improve the result of coclustering that is characterized by the following merge condition.

Definition 15 (Merge Condition)**.** *Given a coclustering* $CO = \{< C_1, SP(C_1) >, \cdots, < C_i, SP(C_i) >, \cdots, < C_j, SP(C_j) >, \cdots, < C_K, SP(C_K) >\}$*, and two clusters* $C_i, C_j \in CO$*, we merge* C_i *and* C_j *into* C_m *and derive a new coclustering* $CO' = \{< C_1, SP(C_1) >, \cdots, < C_m, SP(C_m) >, \cdots, < C_K, SP(C_K) >\}$*. We say* C_i *and* C_j *satisfy the merge condition iff*
1) $OTWCV(CO') \leq OTWCV(CO)$*;*
2) $\theta(C_i, SP(C_i)) \leq \theta(C_m, SP(C_m))$ *and* $\theta(C_j, SP(C_j)) \leq \theta(C_m, SP(C_m))$*.*

Algorithm: Merge
Input: $CO = \{< C_1, SP(C_1) >, \cdots, < C_K, SP(C_K) >\}$
Output: new version of CO in which no more cluster satisfying the merge condition
Begin
  While there exists $C_i, C_j \in CO$ satisfying the merge condition **do**
    Merge C_i and C_j into C_m;
    Recalculate the spouse cluster for each cluster in CO according to Definition 3;
  End while
End Algorithm

Fig. 6. Algorithm Merge

Intuitively, the merge condition ensures that after a merge, 1) the OTWCV value for the new coclustering will not increase, and 2) the connection strength of the new merged cluster is no less than the connection strengths of the two original clusters. Algorithm *Merge* is sketched in Figure 6. Basically, it merges two clusters into one if the two clusters satisfy the above defined merge condition. The algorithm terminates when no more pair of clusters satisfy the above defined merge condition.

5 3-D Visualization of the BCA results

All fiber tracts calculated from DTI data were rendered in relation to the cortical mesh obtained from conformal brain surface mapping [18], as shown in Figure 7-(a)-Top. It can be seen that there is a large number of fiber tracts connecting cortical areas. BCA was performed based on the spatial relationship of voxels on the cortical surface and

Figure 7-(a)-Bottom exhibits clustered cortical fibers in frontal and lateral view. Figure 7-(b) shows the results of our BCA in a representative subject. Well-know anatomical fiber tracts in the brain are reproduced such as the colossal fibers (pink) which connect the two hemispheres and the forceps minor of the corpus callosum (yellow) connecting the left and right side of the frontal cortex. Moreover, the intra-hemispheric connections of the arcuate fasciculus connecting Broca's and Wernicke's cortical areas can be appreciated.

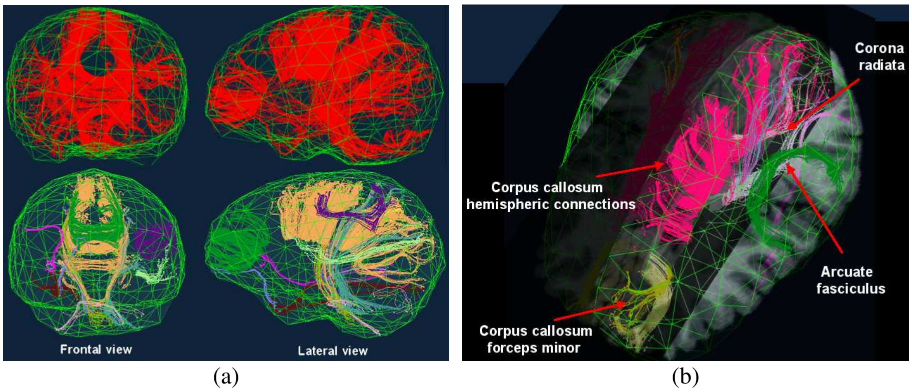

Fig. 7. (a)-Top: frontal and lateral view of cortico-cortical fibers before coclustering; (a)-Bottom: frontal and lateral view of clustered cortico-cortical connectivity; (b) Zoom in view of some specific coritco-cortical clusters.

These results indicate that the developed algorithm is consistent with brain anatomy and that it allows automated segmentation of the cortex based on DTI-derived cortical connections within the brain. We therefore believe that our algorithm is well suited to provide an efficient framework for further analysis including the quantitative assessment of cortico-cortical connectivity.

6 Conclusions and Future Work

In this paper, we defined the coclustering problem and we applied this approach to the analysis of cortico-cortical connections in the brain. Our approach represents an efficient mathematical framework that is computationally robust and is able to be used for quantitative analysis of cortico-cortical fiber tracts. This in turn might be relevant for the identification of secondary epileptic foci in patients with intractable epilepsy and might impact their clinical management.

Although the coclustering problem was initially motivated by the need of cortico-cortical connectivity analysis, we expect that it will have a wide range of applications. In the future, we plan to apply our BCA also to the analysis of thalamo-cortical connectivity and the segmentation of thalamic nuclei.

Acknowledgment

This research was partially supported by the Michigan Technology Tri-Corridor basic research grant MTTC05-135/GR686.

References

1. D. Le Bihan, E. Breton, D. Lallemand, P. Grenier, E. Cabanis and M. Laval-Jeantet. MR imaging of intravoxel incoherent motions: application to diffusion and perfusion in neurologic disorders. *Radiology*, 161:401–407, 1986.
2. D. Le Bihan. Looking into the functional architecture of the brain with diffusion MRI. *Nature Rev Neurosci*, 4:469–480, 2003.
3. P.J. Basser and C. Pierpaoli. Microstructural and physiological features of tissues elucidated by quantitative diffusion-tensor MRI. *J Magn Reson*, Series B:209 – 219, 1996.
4. A.W. Toga P.M. Thompson. A framework for computational anatomy. *Computing and Visualization in Science*, 5:13–34, 2002.
5. R.E. Passingham , K.E. Stephan and R. Kotter. The anatomical basis of functional localization in the cortex. *Nat Rev Neurosci*, 3:606–616, 2002.
6. J. Han and M. Kamber. *Data Mining*, pages 349–353. I-55860-489-8. Morgan Kaufmann Publishers, 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205, USA, 2001.
7. R.T. Ng and J. Han. Efficient and effective clustering methods for spatial data mining. *20th Int. Conf. on Very Large DataBases*, pages 144–155, 1994.
8. S. Guha, R. Rastogi and K. Shim. Cure: An efficient clustering algorithm for large databases. *ACM SIGMOD international conference on management of data*, pages 73–84, 1998.
9. M. Ester, H.P. Kriegel, J. Sander and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. *2nd Int. Conf. on Knowledge Discovery and Data Mining*, pages 226–231, 1996.
10. T.E. Conturo , N.F. Lori, T.S. Cull, E. Akbudak, A.Z. Snyder, J.S. Shimony, R.C. Mckinstry, H. Burton and M.E. Raichle. Tracking neuronal fiber pathways in the living human brain. *Natl. Acad. Sci. USA*, 96:10422–10427, 1999.
11. R.F. Dougherty, M. Ben-Shachar, R. Bammer, A.A. Brewer and B.A. Wandell. Functional organization of human occipital-callosal fiber tracts. *Natl. Acad. Sci. USA*, 102:7350–7355, 2005.
12. P.J. Basser , S. Pajevic, C. Pierpaoli, J. Duda and A. Aldroubi. In vivo fiber tractography using DT-MRI data. *Magn Reson Med*, 4:625–32, 2000.
13. S. Mori, B.J. Crain, V.P. Chacko and P.C.M. van Zijl. Three dimensional tracking of axonal projection in the brain by magnetic resonance imaging. *Ann. Neurol.*, 45:265–269, 1999.
14. G. Karypis, E. Han and V. Kumar. Chameleon: A hierarchical clustering algorithm using dynamic modeling. *Computer*, 32(8):68–75, 1999.
15. M. Ankerst, M.M. Breunig, H. Kriegel and J. Sander. Optics: ordering points to identify the clustering structure. *ACM SIGMOD international conference on management of data*, pages 49 – 60, 1999.
16. A. Hinneburg and D.A. Keim. An efficient approach to clustering in large multimedia databases with noise. In *Knowledge Discovery and Data Mining*, pages 58–65, 1998.
17. C. Lin, S. Lu, X. Liang, and J. Hua. GCA: a Coclustering Algorithm for Thalamo-Cortico-Thalamic Connectivity Analysis. *ICDM Workshop on Data Mining in Bioinformatics*, pages 163–168, December 2006.
18. G. Zou, J. Hua, X. Gu, and O. Muzik. An Approach for Intersubject Analysis of 3D Brain Images based on Conformal Geometry. *IEEE Int. Conf. on Image Processing*, pages 1193 – 1196, 2006.