

UC Santa Barbara

UC Santa Barbara Previously Published Works

Title

A Computational Geometry Approach for Modeling Neuronal Fiber Pathways

Permalink

<https://escholarship.org/uc/item/004972w6>

Authors

Shailja, S
Zhang, Angela
Manjunath, BS

Publication Date

2021

DOI

10.1007/978-3-030-87237-3_17

Peer reviewed



HHS Public Access

Author manuscript

Med Image Comput Comput Assist Interv. Author manuscript; available in PMC 2021 November 01.

Published in final edited form as:

Med Image Comput Comput Assist Interv. 2021 ; 12908: 175–185. doi:10.1007/978-3-030-87237-3_17.

A Computational Geometry Approach for Modeling Neuronal Fiber Pathways

S. Shailja, Angela Zhang, B. S. Manjunath

University of California, Santa Barbara, CA 93117, USA

Abstract

We propose a novel and efficient algorithm to model high-level topological structures of neuronal fibers. Tractography constructs complex neuronal fibers in three dimensions that exhibit the geometry of white matter pathways in the brain. However, most tractography analysis methods are time consuming and intractable. We develop a computational geometry-based tractography representation that aims to simplify the connectivity of white matter fibers. Given the trajectories of neuronal fiber pathways, we model the evolution of trajectories that encodes geometrically significant events and calculate their point correspondence in the 3D brain space. Trajectory inter-distance is used as a parameter to control the granularity of the model that allows local or global representation of the tractogram. Using diffusion MRI data from Alzheimer's patient study, we extract tractography features from our model for distinguishing the Alzheimer's subject from the normal control. Software implementation of our algorithm is available on GitHub (<https://github.com/UCSB-VRL/ReebGraph>).

Keywords

Computational geometry; Computational pathology; Reeb graph; Trajectories; Brain fibers; Connectome

1 Introduction

Diffusion MRI (dMRI) tractography [2] constructs morphological 3D neuronal fibers represented by 3D images called tractograms. In recent years, analysis of fibers in dMRI tractography data has received wide interest due to its potential applications in computational pathology, surgery, and studies of diseases, such as brain tumors [3,10], Alzheimer's [4], and schizophrenia [13]. Tractography datasets are huge and complex consisting of millions of fibers arising and terminating at different functional regions of the brain. Computational analysis of these fibers is challenging owing to their complex topological structures in three dimensions. Tractography produces white matter pathways that can be deduced as spatial trajectories represented by a sequence of 3D coordinates. To model the geometry of these trajectories, we utilize the concept of Reeb graphs [17] that have been successfully used in a wide variety of applications in computational geometry and graphics, such as shape matching, topological data analysis, simplification,

and segmentation. We assume that the groups of trajectories that are spatially close to each other share similar properties. Therefore, we compute a model to encode the arising & ending and the merging & splitting behavior for groups of trajectories (as shown in Fig. 1) along with their point correspondence. With these computations in place, we develop a finite state machine that can be used to query the state of any trajectory or its shared groups. The resulting model has tunable granularity that can be used to derive models with the desired level of geometrical details or abstract properties.

2 Related Work

Brain tractography datasets are constructed from the dMRI of an individual's brain [23,24]. One way to analyze the fiber tracts is to generate a connectivity matrix that provides a compact description of pairwise connectivity of regions of interest (ROI) derived from anatomical or computational brain atlases. For example, the connectivity matrices can be used to compute multiple graph theory-based metrics to distinguish between the brains of healthy children and those with recent traumatic brain injury [21]. However, such methods overlook the geometrical characteristics within a region of interest. A number of inter-fiber distance-based approaches have been used to analyze the fibers [1,5,8,11,12,26] for clustering and segmentation but have some limitations. For example, one needs prior information about the number of clusters to be segmented in [8]. More sophisticated methods produce high-dimensional representations that are not efficient [20,25]. Due to the complex nature of tractography algorithms, another way to compare bundles is by using tract profiling techniques that quantifies diffusion measures along each fiber tract [22]. Notably, researchers in [7] introduced a representation that is sparse and can be integrated with learning methods for further study. However, their approach leads to possible loss of critical points of fibers (due to polynomial fitting) and ignores multi-fiber tractography. Our design addresses this by sequentially processing the group behavior emerging due to events of individual trajectories. Our method builds on previous work on time-dependent trajectory analysis using a Reeb graph. A deterministic algorithm for Reeb graph computation in $O(n \log n)$ time is shown in [14]. Reeb graph can be used to model the trajectory grouping structure defined by time as a parameter [6]. For tractography analysis, the concept of “bundling” and “unbundling” structure of trajectory data to compute a sparse graph is proposed in [18]. They show graph representation of brain tractography but do not present the algorithm or proofs for the computation, focusing instead on the novel problem definition.

3 Preliminaries

In the three dimensional Euclidean space \mathbb{R}^3 , we define the following terms that would help in setting up the problem in this section.

Trajectory: A trajectory T is as an ordered sequence of points in \mathbb{R}^3 . We denote a trajectory T as a sequence of points $\{p_1, p_2, \dots, p_m\}$, where m is the number of points in T and $p_i \in \mathbb{R}^3$.

ϵ -(dis)connected Points: For any pair of points p_1 and p_2 in \mathbb{R}^3 , we define $d(p_1, p_2)$ as the Euclidean distance between the two points:

$$d(p_1, p_2) = \|p_1 - p_2\|_2,$$

where $\|\cdot\|_2$ represents the Euclidean norm. Two points p_1, p_2 are ϵ -connected if $d(p_1, p_2) \leq \epsilon$. Similarly, two points p_1, p_2 are ϵ -disconnected if $d(p_1, p_2) > \epsilon$.

Appear Event: For each trajectory T , the initial point of its ordered sequence is labeled for the occurrence of the *appear event*. For example, for trajectory $T_1 = \{p_1, p_2, \dots, p_m\}$, we observe the appear event at the point p_1 .

Disappear Event: For each trajectory T , the final point of its ordered sequence is labeled for the occurrence of the *disappear event*. For example, for trajectory $T_1 = \{p_1, p_2, \dots, p_m\}$, we observe the disappear event at p_m .

Connect Events: To define *connect events* for a pair of trajectories, consider two trajectories

$$T = \{p_1, p_2, \dots, p_m\}, \quad T' = \{p'_1, p'_2, \dots, p'_m\},$$

then a connect event for the pair (T, T') is defined by (p_i, p'_j) such that $p_i \in T, p'_j \in T'$ and,

$$d(p_i, p'_j) \leq \epsilon, \quad d(p_{i-1}, p'_{j-1}) > \epsilon, \quad \text{for } i > 1 \text{ and } j > 1.$$

If there is no such pair of points, it implies that T and T' are disjoint. Moreover, if T and T' are ϵ -connected at (p_i, p'_j) and if T' and T^* are also ϵ -connected at (p'_j, p_l^*) where $p_l^* \in T^*$, then we say that T and T^* are ϵ -step connected at (p_i, p_l^*) .

Disconnect Events: Given a pair of trajectories (T, T') with a connect event at (p_i, p'_j) , we define a *disconnect event* by (p_{i+k}, p'_{j+k}) such that,

$$d(p_{i+k}, p'_{j+k}) > \epsilon, \quad d(p_{i+k-1}, p'_{j+k-1}) \leq \epsilon.$$

Max-width ϵ -Connected Trajectories: For an input \mathcal{S} , there are many possible ϵ -step connected trajectories. The maximal group of trajectories at a given step k are called *max-width ϵ -step connected* and there is no other possible set of sub-trajectories that can intersect with the maximal group at k .

Note that the trajectories estimated from dMRI tractography do not have a specific beginning or ending, as dMRI is not sensitive to the direction of connections. So, reversing the order of the points of a streamlines will produce similar results. Two events appear and disappear are used for convenience in describing the algorithm and its implementation.

3.1 Problem Formulation

We set up the following central problem for this paper:

Input: A set of trajectories $\mathcal{T} = \{T_1, T_2, \dots, T_n\}$, such that $T_i \in \mathbb{R}^3$ for all $i \in \{1, 2, \dots, n\}$ where n is the number of trajectories.

Output: A finite state machine (FSM) S that models the evolution of trajectories and their critical points of interaction with all other trajectories.

$$S = (A, O, V, \mathcal{R}),$$

$$\mathcal{R}: (S \times A) \rightarrow (S \times O),$$

where A is the set of events associated with each trajectory $T_i \in \mathcal{T}$, O is the set of outputs encoding the location information of the critical events, V is the set of states that corresponds to a group of trajectories. \mathcal{R} is the state-transition and output function. When the machine is in a current state $v \in V$ and receives an input $a \in A$ it moves to the next state specified by R and produces an output location $o \in O$ as shown in Fig. 2.

4 Reeb Graph

The central part of solving the problem as stated above is to compute \mathcal{R} —the state transition and output function. Towards that end, we compute an undirected graph \mathcal{R} called the Reeb graph. In this section, we define the Reeb graph and then proceed to develop an algorithm that can compute this graph for a set of trajectories. Formally, a Reeb graph \mathcal{R} is defined on a manifold $\mathcal{M} \in \mathbb{R}^3$ using the evolution of level sets L [9]. To adapt this definition of \mathcal{R} for the case of neuronal fiber trajectory evolution problem, we define a manifold \mathcal{M} in \mathbb{R}^3 as the union of all points in the tractogram. The set of points of trajectories at step k is the level set of k . The connected components in the level set of k correspond to the max-width ϵ -connected trajectories at step k . Unlike previous studies [6], here, any number of trajectories can become ϵ -(dis)connected at the same location. Reeb graph \mathcal{R} describes the evolution of the connected components over sequential steps. At every step k , the changes in connected components (states of FSM) are represented by vertices in \mathcal{R} .

4.1 Computing the Reeb Graph

In Sect. 3, for a given trajectory, we defined appear and disappear events. For a pair of trajectories, we defined connect and disconnect events. These events ($a \in A$) describe the branching structure of the trajectories. To compute the Reeb graph, we process these events sequentially. We maintain a graph $G = (V, E')$ where the vertices represent the set of trajectories. G is a graph that changes with steps representing the connect and disconnect relations between different trajectories. At each step k , we insert new nodes at appear events and delete nodes at disappear events. At connect events, we insert edges in G and at disconnect events, we delete edges. At each step k , an edge (T_1, T_2) in G shows that T_1 and T_2 are directly connected. Therefore the max-width ϵ -connected trajectories correspond to the connected components in G at step k .

Initialization: We spend $\mathcal{O}(N^2)$ time to store the appear, disappear, connect, and disconnect events for all pair of the trajectories. We store a mapping M from the current components in G to the corresponding vertices in \mathcal{R} . We start from one of the trajectories and add other trajectories of interest on the way of following its points sequentially. We maintain a data structure to flag the points for which the events are already processed and store their mappings to the vertices of the Reeb graph in D . Note that although the computational time is $\mathcal{O}(N^2)$, this step is massively parallelizable.

Split and Merge: To handle a disconnect event of trajectories T_1 and T_2 at step k , we delete the edge (T_1, T_2) from G_k . Similarly, for a connect event of trajectories T_1 and T_2 at step k , we add the edge (T_1, T_2) to G_k . We do this for all the connect and disconnect events as shown in Fig. 3 for trajectory T_1 . For the disconnect event, we query G_{k-1} to get the connected component C consisting of trajectories T_1 and T_2 and locate the corresponding u in \mathcal{R} . We query G_k to get the connected components C_1 and C_2 consisting of trajectories T_1 and T_2 , respectively. $C_1 = C_2$ implies that the trajectories T_1 and T_2 are still ϵ -step connected. If $C_1 \neq C_2$, we add a new split vertex v to \mathcal{R} and a new edge (u, v) and update M accordingly.

Computing \mathcal{R} from G : We query G_k and G_{k-1} to get the connected components at step k and $k-1$ respectively. For each connected component C_c in G_k , if C_c is present in the connected components of G_{k-1} , then we do not modify \mathcal{R} . This implies that no such event occurred in the trajectories of C_c which could result in any critical points. Otherwise, using M , we locate the corresponding nodes in \mathcal{R} for the connected components in G_{k-1} , we call it previous connected components. The corresponding nodes in \mathcal{R} for the connected components in G_k are called present connected components. For each component in present connected components, we add a node v in \mathcal{R} if not already present in the previous connected component and assign the location $(o \in O)$ as the coordinates of one of the points in the connected components. If that is the case, we also add an edge (u, v) , where u is the node corresponding to previous connected component C_1 and v is the node corresponding to present connected component C_2 , if $|C_1 \cap C_2| > 0$. Finally, we update M accordingly.

At next step $k+1$, if we encounter the point of a trajectory T for which the events have already been processed, we query D to locate the vertex u and v in \mathcal{R} for p_k and p_{k+1} respectively. We add an edge (u, v) to \mathcal{R} as shown in Fig. 4 and delete the node corresponding to trajectory T in G and update M .

Theorem 1.—*For a given set of trajectories, $\mathcal{F} = \{T_1, T_2, \dots, T_n\}$ with a total of N points, the Reeb graph \mathcal{R} of \mathcal{F} can be computed in $\mathcal{O}(N \log N)$ time.*

Proof.: It is possible to compute the connected components of a graph $G(V', E')$ with $|V'|$ vertices and $|E'|$ edges using Breadth First Search (BFS) or Depth First Search (DFS) in $\mathcal{O}(N^2)$ time. But, since we know all of the points of the given input \mathcal{F} at which any event occurs, we can use a dynamic graph connectivity approach [14] to improve the computation time. This method allows connectivity operations, inserts, and deletes, in $\mathcal{O}(\log N)$ time. In

the worst case, we modify and query the graph G to get the connected components for all the points in \mathcal{S} . Hence, the total time required for the construction of \mathcal{R} is $\mathcal{O}(N \log N)$.

5 Examples and Applications

To the best of our knowledge, there are no existing modeling methods in the literature for brain fibers that can be used to compare our method directly. To provide the proof of concept and demonstrate utility, we evaluate our proposed algorithm on real data and validate manually as illustrated in Fig. 5. To design a case study demonstrating the utility of \mathcal{R} , we randomly select 22 subjects (11 Normal and 11 Alzheimer's patient) from the publicly available Alzheimer's Disease Neuroimaging Initiative (ADNI) [15] dataset (<http://adni.loni.usc.edu/>). We evaluate the qualitative representation of critical points using our model on fibers for random ROIs. All the analytically significant points are captured by \mathcal{R} through nodes and edges, which are highly consistent across subjects. The proposed model can be employed in the existing deep learning and machine learning algorithms to provide new insights into the structure and the function of the brain. Similar to recent research works where graph theory-based features are utilized for classification tasks, we compute the total number of max-width ϵ -connected groups that is $|E|$ and the aggregate of significant points on fibers that is $|V|$. We also calculate network properties such as clustering, centrality, modularity, and efficiency of \mathcal{R} . We choose two ROIs: Posterior Cingulate Gyrus and Middle Occipital Gyrus from the left hemisphere based on the Automated Anatomical Labelling (AAL) atlas [16,19] and compute tractography consisting of 1000 fibers in each ROI for each subject. We used Q-Space Diffeomorphic Reconstruction as implemented in DSI Studio [23] to compute the fibers. In Fig. 6, we show the distribution of a set of properties that can be used to facilitate comparisons between Alzheimer's and normal subjects. By comparing the p-value for the ROIs shown in Fig. 6, we can conclude that Posterior Cingulate Gyrus (lesser p-value) is a more significant ROI than Middle Occipital Gyrus. This is in accordance with the study [16] that highlights the relevant ROIs for Alzheimer's disease. The average run time of our implementation for examples consisting of 132,000 points on average was 42 s on Intel Core CPU 4 GHz processor with 32 GB RAM.

6 Conclusion

Our paper proposes the study of the spatial evolution of neuronal trajectories including the algorithmic analysis. We also demonstrate how our proposed reduced graph encodes the critical points of the pathways. Point correspondence of the critical coordinates in the 3D brain space calculated in our algorithm is an essential requirement of the tract-orientated quantitative analysis which is overlooked in the previous works. This aids in localizing and underpinning the points of interest in white matter tracts. Through our preliminary experiments, we show a set of properties of the Reeb graph that can be used to distinguish between the Alzheimer's patients and control subjects. In future, we plan to utilize graph-theoretic concepts to analyze the Reeb graph models of white matter fibers. We intend to further evaluate the reproducibility of our approach on additional datasets. Integrating graph-theoretic features of the Reeb graph with data-driven learning approaches can greatly improve our understanding of various human disease pathways.

Acknowledgments

Supported by NSF award: SSI # 1664172 and NIH award # 5R01NS103774-02.

References

1. Andersson M, Gudmundsson J, Laube P, Wolle T: Reporting leaders and followers among trajectories of moving point objects. *GeoInformatica* 12(4), 497–528 (2008)
2. Basser PJ, Pajevic S, Pierpaoli C, Duda J, Aldroubi A: In vivo fiber tractography using DT-MRI data. *Magn. Reson. Med* 44(4), 625–632 (2000) [PubMed: 11025519]
3. Berman JI, Berger MS, Mukherjee P, Henry RG: Diffusion-tensor imaging–guided tracking of fibers of the pyramidal tract combined with intraoperative cortical stimulation mapping in patients with gliomas. *J. Neurosurg* 101(1), 66–72 (2004) [PubMed: 15255253]
4. Bozzali M, et al. : White matter damage in Alzheimer’s disease assessed in vivo using diffusion tensor magnetic resonance imaging. *J. Neurol. Neurosurg. Psychiatry* 72(6), 742–746 (2002) [PubMed: 12023417]
5. Brun A, Knutsson H, Park H-J, Shenton ME, Westin C-F: Clustering fiber traces using normalized cuts. In: Barillot C, Haynor DR, Hellier P (eds.) MICCAI 2004. LNCS, vol. 3216, pp. 368–375. Springer, Heidelberg (2004). 10.1007/978-3-540-30135-6_45
6. Buchin K, Buchin M, van Kreveld M, Speckmann B, Staals F: Trajectory grouping structure. In: Dehne F, Solis-Oba R, Sack J-R (eds.) WADS 2013. LNCS, vol. 8037, pp. 219–230. Springer, Heidelberg (2013). 10.1007/978-3-642-40104-6_19
7. Cabeen RP, Toga AW, Laidlaw DH: Tractography processing with the sparse closest point transform. *Neuroinformatics*, 1–12 (2020) [PubMed: 30982183]
8. Dodero L, Vascon S, Murino V, Bifone A, Gozzi A, Sona D: Automated multi-subject fiber clustering of mouse brain using dominant sets. *Front. Neuroinform* 8, 87 (2015) [PubMed: 25628561]
9. Doraiswamy H, Natarajan V: Efficient algorithms for computing Reeb graphs. *Comput. Geom* 42(6–7), 606–616 (2009)
10. Kao PY, Shailja S, Jiang J, Zhang A, Khan A, Chen JW, Manjunath B: Corrigendum: improving patch-based convolutional neural networks for MRI brain tumor segmentation by leveraging location information. *Front. Neurosci* 14 (2020)
11. Mobergs B, Vilanova A, Van Wijk JJ: Evaluation of fiber clustering methods for diffusion tensor imaging. In: VIS 05. IEEE Visualization, pp. 65–72. IEEE (2005)
12. O’Donnell LJ, Schultz T: Statistical and machine learning methods for neuroimaging: examples, challenges, and extensions to diffusion imaging data. In: Hotz I, Schultz T (eds.) Visualization and Processing of Higher Order Descriptors for Multi-Valued Data. MV, pp. 299–319. Springer, Cham (2015). 10.1007/978-3-319-15090-1_15
13. Park HJ, et al. : White matter hemisphere asymmetries in healthy subjects and in schizophrenia: a diffusion tensor MRI study. *Neuroimage* 23(1), 213–223 (2004) [PubMed: 15325368]
14. Parsa S: A deterministic $\alpha \log(m)$ time algorithm for the Reeb graph. *Discrete Comput. Geom* 49(4), 864–878 (2013)
15. Petersen RC, et al. : Alzheimer’s disease neuroimaging initiative (ADNI): clinical characterization. *Neurology* 74(3), 201–209 (2009). 10.1212/wnl.0b013e3181cb3e25 [PubMed: 20042704]
16. Rondina JM, et al. : Selecting the most relevant brain regions to discriminate Alzheimer’s disease patients from healthy controls using multiple kernel learning: a comparison across functional and structural imaging modalities and atlases. *NeuroImage Clin* 17, 628–641 (2018). 10.1016/j.nicl.2017.10.026 [PubMed: 29234599]
17. Shinagawa Y, Kunii TL, Kergosien YL: Surface coding based on Morse theory. *IEEE Comput. Graph. Appl* 11(5), 66–78 (1991)
18. Sun J, Cieslak M, Grafton S, Suri S: A reeb graph approach to tractography. In: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 1–4 (2015)

19. Tzourio-Mazoyer N, et al. : Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *NeuroImage* 15(1), 273–289 (2002). 10.1006/nimg.2001.0978 [PubMed: 11771995]
20. Wang Q, Yap P-T, Wu G, Shen D: Fiber modeling and clustering based on neuroanatomical features. In: Fichtinger G, Martel A, Peters T (eds.) *MICCAI 2011*. LNCS, vol. 6892, pp. 17–24. Springer, Heidelberg (2011). 10.1007/978-3-642-23629-7_3
21. Watson CG, DeMaster D, Ewing-Cobbs L: Graph theory analysis of DTI tractography in children with traumatic injury. *NeuroImage Clin* 21, 101673 (2019). 10.1016/j.nicl.2019.101673 [PubMed: 30660661]
22. Yeatman JD, Dougherty RF, Myall NJ, Wandell BA, Feldman HM: Tract profiles of white matter properties: automating fiber-tract quantification. *PLoS ONE* 7(11), e49790 (2012) [PubMed: 23166771]
23. Yeh FC, Tseng WYI: NTU-90: a high angular resolution brain atlas constructed by Q-space diffeomorphic reconstruction. *NeuroImage* 58(1), 91–99 (2011). 10.1016/j.neuroimage.2011.06.021 [PubMed: 21704171]
24. Yeh FC, Verstynen TD, Wang Y, Fernández-Miranda JC, Tseng WYI: Deterministic diffusion fiber tracking improved by quantitative anisotropy. *PLoS ONE* 8(11), e80713 (2013). 10.1371/journal.pone.0080713 [PubMed: 24348913]
25. Yendiki A, Panneck P, Srinivasan P, Stevens A, Zöllei L, Augustinack J, Wang R, Salat D, Ehrlich S, Behrens T, et al. : Automated probabilistic reconstruction of white-matter pathways in health and disease using an atlas of the underlying anatomy. *Front. Neuroinform* 5, 23 (2011) [PubMed: 22016733]
26. Zhang S, Correia S, Laidlaw DH: Identifying white-matter fiber bundles in DTI data using an automated proximity-based fiber-clustering method. *IEEE Trans. Vis. Comput. Graph* 14(5), 1044–1053 (2008) [PubMed: 18599916]

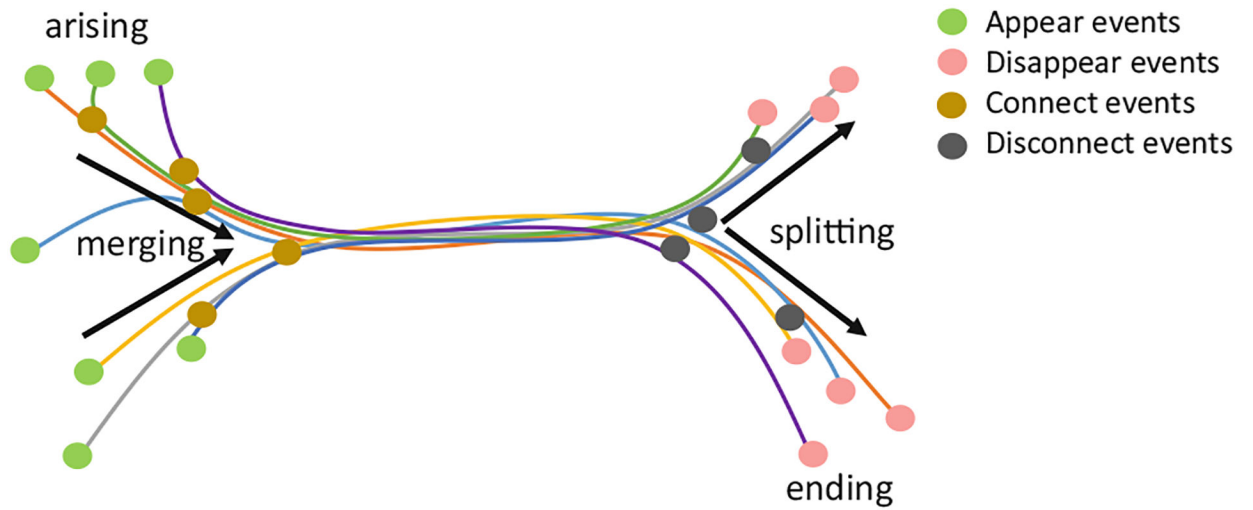


Fig. 1. A basic example of a set of trajectories displaying the arising, merging, splitting, and ending behaviour. These qualitative behaviors of group of trajectories emerge due to the events (appear, connect, disconnect, and disappear) of individual trajectory. Events of trajectories are used to define behavior of the group of trajectories.

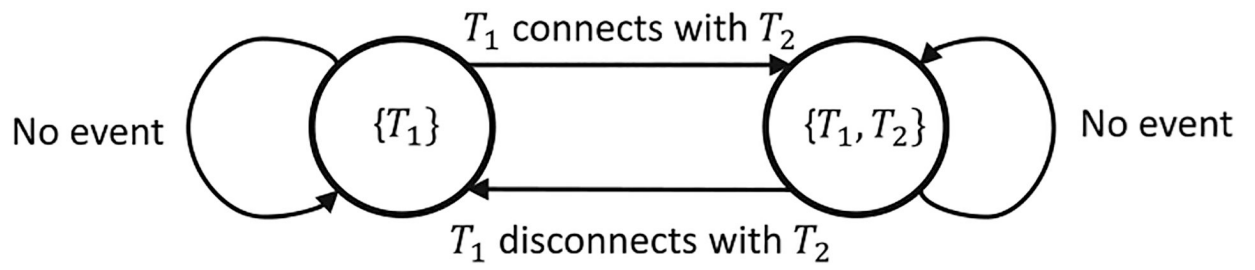


Fig. 2.

An example of state diagram for trajectory T_1 . T_1 is either directly (dis)connected with T_2 or ϵ -step (dis)connected.

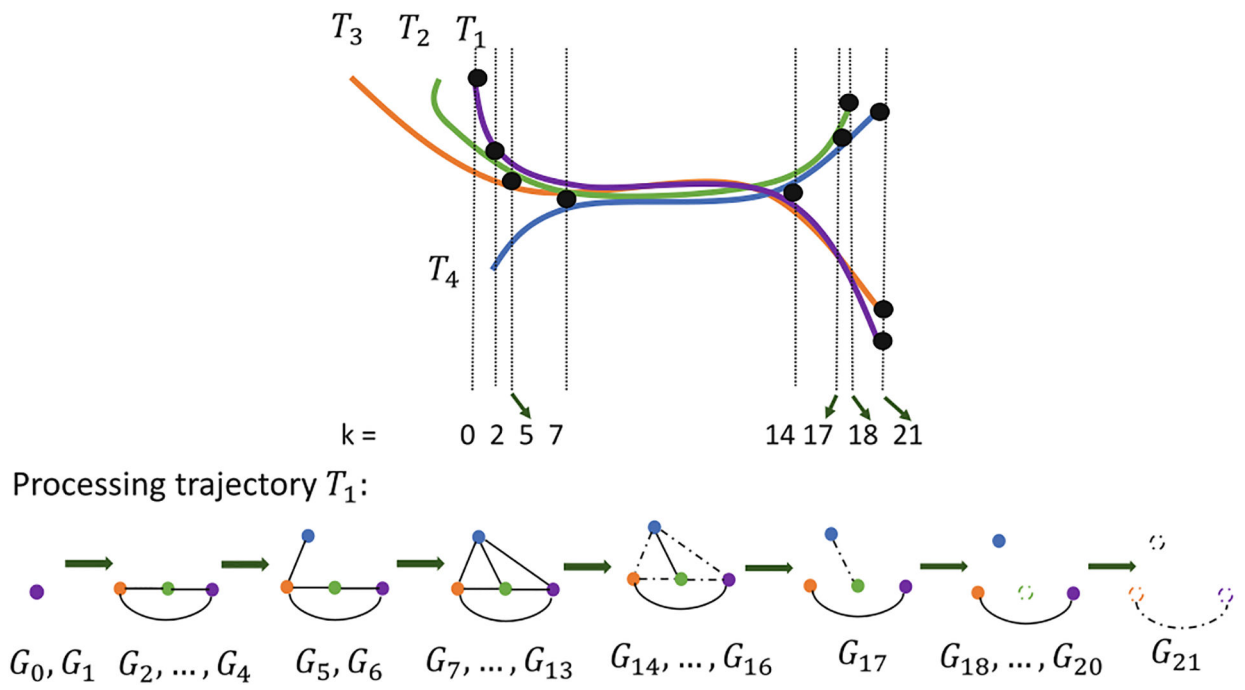


Fig. 3.

The figure shows an input $\mathcal{S} = \{T_1, T_2, T_3, T_4\}$ exhibiting arising, merging, splitting, and ending behaviour. After processing the sequence of points in T_1 , at $k = 0, 2, 5, 7, 14, 17, 18, 21$ steps, we modify the G_k respectively. Connect, disconnect, appear, and disappear events associated with T_1 are marked by black circles. Delete node and edge queries are represented by dashed circle and dashed line in G_k .

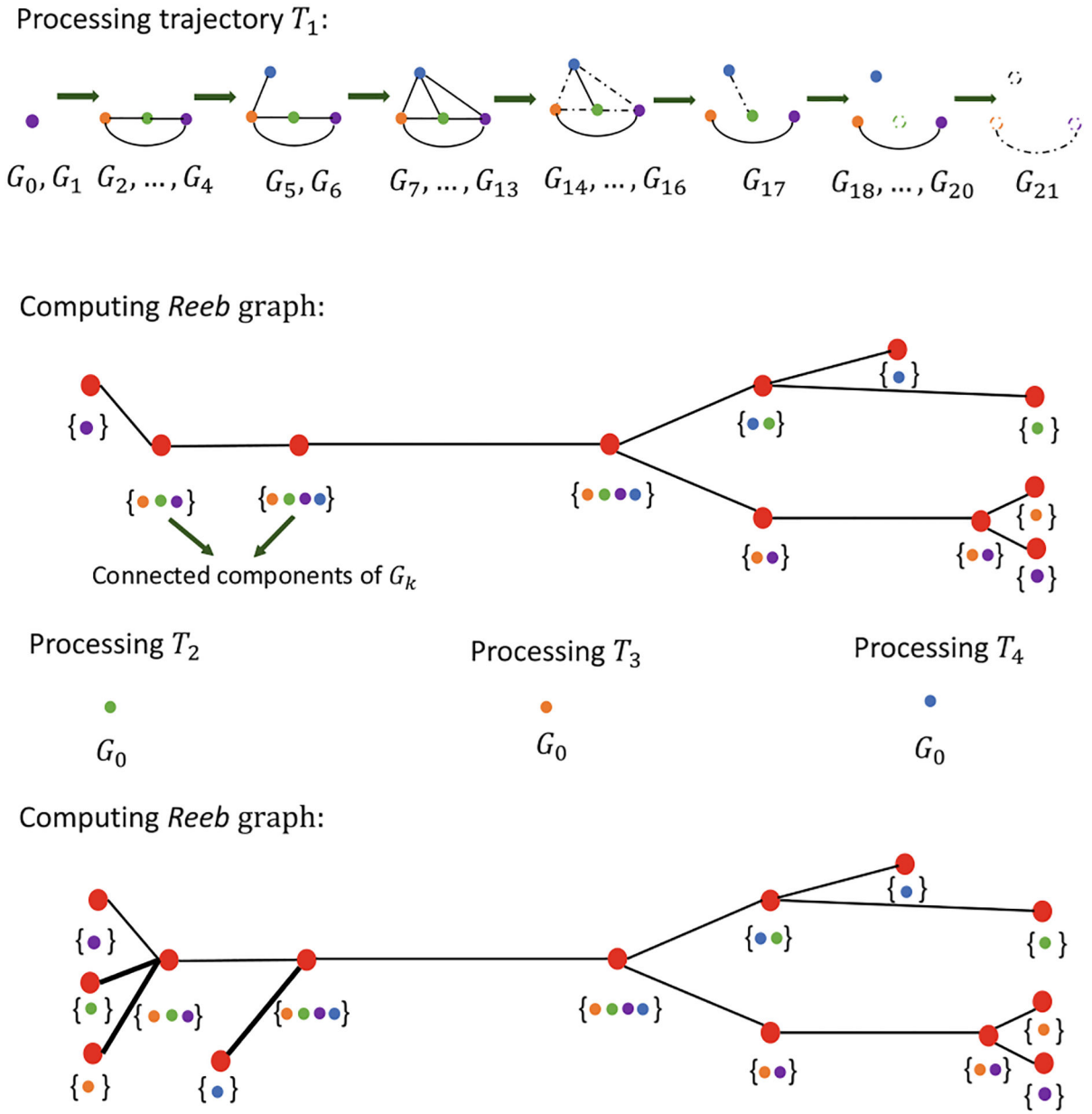


Fig. 4. Continuing the same example from Fig. 3, we show representation of \mathcal{R} from G_k . Edges of \mathcal{R} encodes the maximal group of trajectories and vertices of \mathcal{R} records the significant points of the trajectories.

Author Manuscript

Author Manuscript

Author Manuscript

Author Manuscript

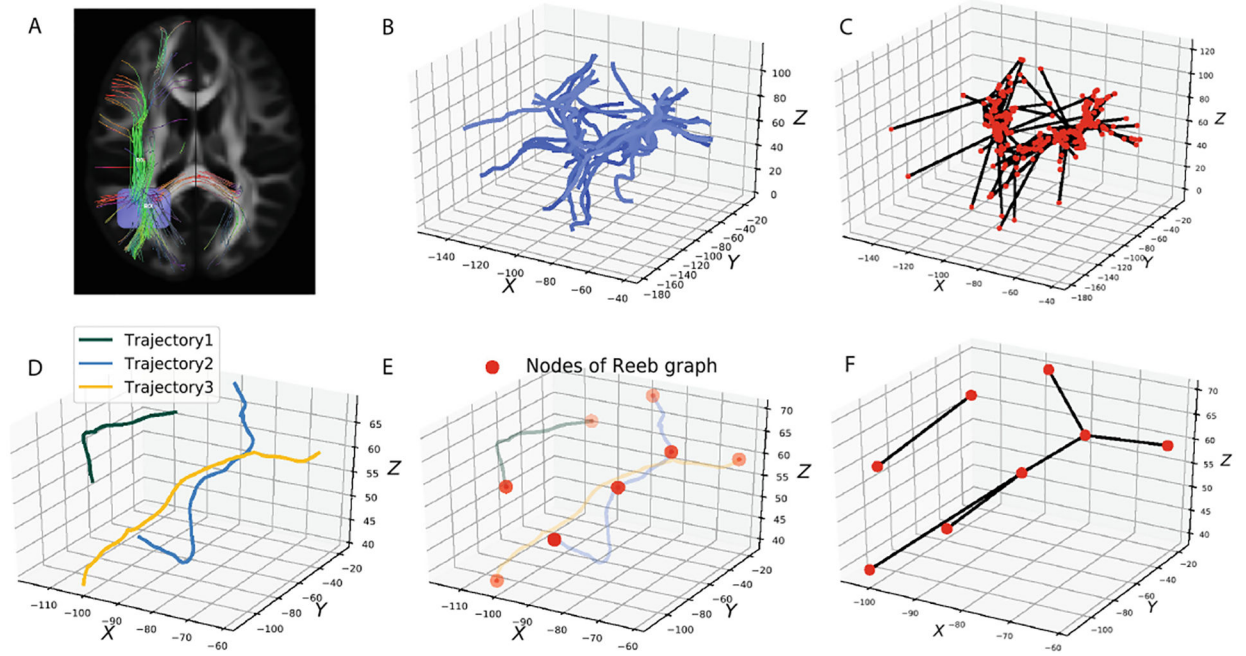


Fig. 5.

(A) shows the fiber tracts created by DSI Studio (<http://dsi-studio.labsolver.org/Manual/Fiber-Tracking>) for an example ROI. (B) shows the example white matter fiber tracts in 3D. (C) exhibits the corresponding \mathcal{R} for the example fibers. (D) shows three fibers from B to form a qualitative impression of our proposed algorithm. (E) indicates the nodes of \mathcal{R} overlapped on the trajectories. (F) represents the proposed grouping structure with the vertices and edges.

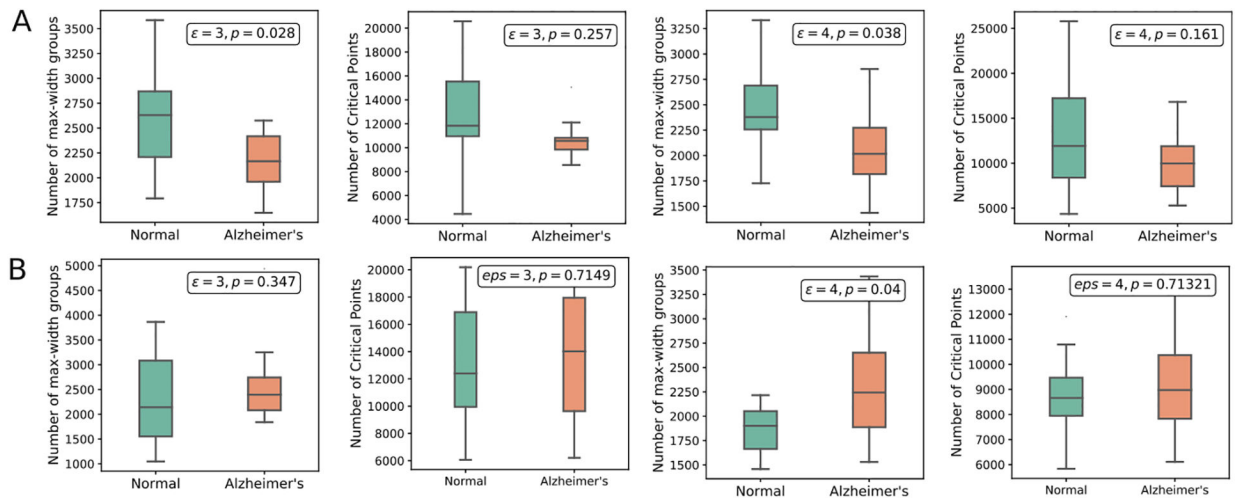


Fig. 6. Statistical analysis results for different values of ϵ showing the comparison between normal and Alzheimer's subjects across properties of \mathcal{R} for ROI A) Posterior Cingulate Gyrus and B) Middle Occipital Gyrus