BRAIN CONNECTOME NETWORK PROPERTIES VISUALIZATION

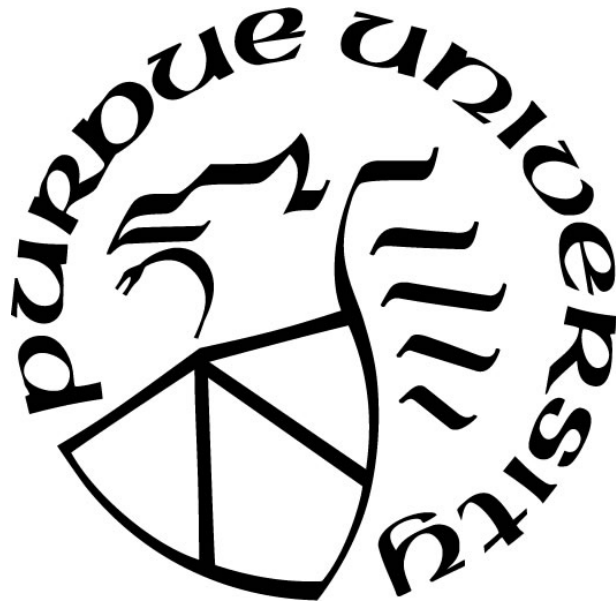by

Chenfeng Zhang

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science

Department of Computer Science

Indianapolis, Indiana

December 2018

THE PURDUE UNIVERSITY GRADUATE SCHOOL

STATEMENT OF COMMITTEE APPROVAL

Dr. Shiaofen Fang, Chair

     Department of Computer & Information Science

Dr. Mihran Tuceryan

     Department of Computer & Information Science

Dr. Snehasis Mukhopadhyay

     Department of Computer & Information Science

Approved by:

     Dr. Shiaofen Fang

       Head of the Graduate Program

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Author: Zhang, Chenfeng. MS
Institution: Purdue University
Degree Received: December 2018
Title: Brian Connectome Network Properties Visualization
Committee Chair: Shiaofen Fang

Brain connectome network visualization could help the neurologists inspect the brain structure easily and quickly. In the thesis, the model of the brain connectome network is visualized in both three dimensions (3D) environment and two dimensions (2D) environment. One is named "Brain Explorer for Connectomic Analysis" (BECA) developed by the previous research already. It could present the 3D model of brain structure with region of interests (ROIs) in different colors [5]. The other is mainly for the information visualization of brain connectome in 2D. It adopts the force-directed layout to visualize the network. However, the brain network visualization could not bring the user intuitively ideas about brain structure. Sometimes, with the increasing scales of ROIs (nodes), the visualization would bring more visual clutter for readers [3]. So, brain connectome network properties visualization becomes a useful complement to brain network visualization. For a better understanding of the effect of Alzheimer's disease on the brain nerves, the thesis introduces several methods about the brain graph properties visualization. There are the five selected graph properties discussed in the thesis. The degree and closeness are node properties. The shortest path, maximum flow, and clique are edge properties. Except for clique, the other properties are visualized in both 3D and 2D. The clique is visualized only in 2D. For the clique, a new hypergraph visualization method is proposed with three different algorithms. Instead of using an extra node to present a clique, the thesis uses a "belt" to connect all nodes within the same clique. The methods of node connections are based on the traveling salesman problem (TSP) and Law of cosines. In

addition, the thesis also applies the result of the clique to adjust the force-directed layout of brain graph in 2D to dramatically eliminate the visual clutter. Therefore, with the support of the graph properties visualization, the brain connectome network visualization tools become more flexible.

CHAPTER 1 INTRODUCTION

Data visualization is now widely used in our modern life. Not only professionals but also ordinary people are getting used to using such an intuitive and fast way to receive information. For example, most credit card companies have adopted a pie chart to show the categories of expense instead of listing each spending record only. From the perspective of biology, the brain could process the graph much faster than words or spreadsheet [1]. In other words, the graph could help people receive more information. So, data visualization could boost both information delivery and receive. Data visualization development could be considered into two parts: method study and tool development. Data visualization method is about the way how the information could be presented. It mainly focuses on how to describe the information precisely, and how the reader could find useful information easily and quickly. The method includes all kinds of visualization techniques is based on different data types [2]. The other one is data visualization tool. It is an application of data visualization. With the development of big data, data visualization tool should possess proper data visualization methods to provide users with various ways to manipulate different data types, like video or image files. It is an interactive way to custom visualization ways according to user's needs. Besides, some data visualization tools could even process real-time or near real-time data streaming. Those convenient data visualization tools could fully meet the user's requirements to some extent. Therefore, with the increasing needs of analysis and dissemination of information, data visualization research become more and more important.

Although the graph seems to work better compared to words, the node-link network sometimes still causes confusion and visual clutter for readers [3]. For example, there is a map with hundreds of cities. Each pair of cities is connected by a line if there is a direct path between them. It is a very normal map prototype. However, the user could not immediately find the city which has the most

direct connections to others, if he wants to visit a convenient city. Besides, in the social network application, a user would be interested to know if his friends also know each other. By inspecting the graph properties, the above requirements could be satisfied easily. In other words, the visualization of graph property could be a wise complement to a graph.

Brain connectome network is a study about the mapping of neural connections in the brain by means of graph theory. Brain connectome visualization could show the pattern of the dataset and help people get information quickly and easily. Currently, there are several brain connectome network visualization tools. For example, BrainNet could visualize brain network from different levels in 3D.

However, if the network is getting more complex, or if the number of nodes increases, the visualization result sometimes could cause confusion. For example, it is hard to tell the connectivity of one node when the edges are mixed together. Or how the nodes of the hypergraph are impacted by the network change.

The thesis is about how to add properties visualization as a supplement for graph visualization. It discusses the brain connectome network properties visualization and related tools based on the previous research work. Generally, brain connectome is the spatial topology of nervous connection based on neuroimaging data [4]. The definition of nodes and edges in brain network are very open according to different needs [4]. In this paper, the dataset is downloaded from Alzheimer's disease Neuroimaging Initiative (ADNI) website (www.adni-info.org) The ADNI started in 2004 which has been focusing on the research of Alzheimer's disease. The website provides four dataset types: clinical, genetic, magnetic resonance imaging (MRI) image, positron emission tomography (PET) image and biospecimen. According to the previous research, the dataset is from the MRI and (Diffusion Tensor Imaging) DTI scans including 134 subjects. These subject data samples are

divided into three groups: healthy control (61 samples), mild cognitive impairment (50 samples) and early Alzheimer's disease (23 samples) [5].

To build the brain network using the dataset, nodes and edges have been defined in previous research work [5]. The node is defined as the regions of interest (ROI). The ROI is from anatomical parcellation [5]. The scales of ROI in the dataset are 83,129 and 324. The edge weight is defined as the density of the fibers between the ROIs [5]. With the nodes and edges, the brain network could be presented as an undirected weight graph. The adjacency matrix of the graph is stored as Comma Separated Values (CSV) files. For each ROI scale, there are 134 CSV files. In each scale, the CSV files are divided into three categories. Although each node (ROI) has its weight in ADNI dataset in the connectome matrix, the thesis only focuses on studying the relations between the ROIs. Therefore, the diagonal elements of the matrix are set to zero.

Graph drawing is about how to visualize the graph. There are serval ways to automatically decide where to place the nodes like concentric circles, etc. [6]. In the thesis, I use two ways to visualize the graph. One is based on the scientific brain visualization tool. This tool provides a volume rendering of the brain outline. Therefore, the locations of ROIs in the brain are fixed. The other is the force-directed graph drawing method in 2D. The force-directed graph layout is based on the spring model which dynamically set up a spring balanced system [6]. In the layout, the location of the node could be set with an initial value. Then the layout will adjust the nodes to reach balance. The spring force between each pair nodes is defined as the graph weight in the system.

However, brain graph drawing is not enough for the scientists to find the inside features of nerve connections. To better understand the damage of Alzheimer's disease to the brain connectome, the thesis chose five graph properties as the complementary for the graph. The degree and closeness are node properties. The shortest path, maximum flow, and clique are edge properties. Except for

clique, the rest properties are visualized in both 3D and 2D. As for the clique, a new hypergraph visualization method is come up with three algorithms in 2D. Instead of using an extra node to present a clique in the hypergraph, the thesis uses a "belt" to connect all nodes within the same clique. The nodes connection methods are based on Law of cosines calculations and TSP. In addition, the thesis also applies the result of the clique to adjust the force-directed layout of brain graph in 2D to dramatically eliminate the visual clutter. With these properties, the neurologists could compare the brain network connections between normal and abnormal brain.

Several software development tools are used in the thesis. The previously developed brain connectome network visualization tool BECA adopted VTK C++ library for volume rendering and Qt platform for GUI design. Because VTK library is widely for scientific visualization and medical image area (https://www.vtk.org/). The 2D information visualization chose MATLAB GUI. MATLAB is very powerful in numerical computation. Besides MATLAB GUI could provide various tools for tool design(https://www.mathworks.com/help/matlab/index.html).

In this thesis, chapter 2 includes the related work of current brain connectome network visualization and hypergraph visualization methods. Chapter 3 introduces the basic concepts of graph theory. In chapter 4, I focus on the four graph properties visualization in both 2D and 3D tools. Chapter 5 is about the information visualization tool design using MATLAB GUI. I tried three algorithms for the clique visualization: distance algorithm, degree change algorithm and center of gravity algorithm. By introducing a dummy point in the distance algorithm, I convert the problems to the classic TSP. Theoretically, the TSP is an NP-hard problem. The complexity of the Dynamic Programming method is $O(n^2*2^n)$ [7]. Although the complexity of TSP could be in the interval $O(n^4)$, $O(n^3*2^n)$) [8], it is still too much time for a user interface tool. So, the approximation algorithm Cross-Entropy (CE) is used to reduce the running time of TSP. Besides,

the chapter also comes up with a method about improving the visual clutter of force-directed layout with cliques. The last chapter provides the conclusion and future work.

CHAPTER 2 RELATED WORK

2.1 Brain Connectome and Visualization Tool Development

The neurobiological research has been working on brain connectome network establishment for a long time. The relations among anatomical connections and the neuronal elements are still the focus [9]. With the application of graph theory in the brain functional connectivity research, the analysis of the human brain connectome network has been brought to a new stage [10] [11]. To understand the brain connectome in an intuitive way, more and more such visualization tools are developed to help the neurologists to learn and investigate brain structure, function and connectivity in graph theoretical methods [5] [12] [13].

In the standing point of data visualization subfield, there are two categories: scientific visualization and information visualization. Scientific visualization is mainly about spatial dataset like airflow simulation, etc. Obviously, the brain connectome network is a 3D model which should use the scientific visualization method. However, information visualization, which is about non-spatial dataset is also indispensable. Because other than the location of the dataset in the space, the neurologists are also interested in the property of data elements themselves [14]. In other words, 3D visualization could provide the structure and connection of brain network. The 2D graph could disclose the features and patterns of the data elements. Therefore, the scientific visualization and information visualization are both very important for the brain connectome network visualization.

 Currently, there are a lot of interactive visualization tools according to different brain research needs. For example, BrainNet Viewer could draw a ball-stick model in 3D using the user-defined brain surface file, node, and edge file and volume file [12]. It constructs the brain network according to the anatomical reference directly. With the toolbox of the application, the

visualization result could be changed by different parameters. Another is the Connectome visualization Utility (CVU). Except for constructing brain network according to user's input, CVU could also provide more various visualization methods. To receive better visualization result, it provides selected nodes highlighted or isolated display method, matrix and circle views and network statistics, and so on. [13]. The CVU tool applies more visualization methods in presenting brain connectome network. Therefore, in the paper "Visualizing the human connectome", Daniel S. Margulies summarized the visualization tools into three types according to the points of focus: "Anatomical connectivity", "Functional connectivity" and "Multimodal connectivity" [15]. He also pointed out that the visualization result of a whole brain connectome network could also visual clutter sometimes for readers. So, the visualization tool is based on the types of dataset. It should follow the original rule of visualization that it could help the user get the information quickly and easily.

BECA is a visualization tool developed by the previous research group. This tool is designed based on the ADNI dataset. It is used for Alzheimer's disease research. The brain connectome network is from MRI and DTI scans based on three steps: ROI generation, DTI tractography and connectivity presentation [5]. From the dataset definition, the node in the brain connectome network is ROI. The edge weight is defined as the density of the fibers between the ROIs. By utilizing both scientific visualization and information visualization methods, the tool could help the user interactively explore the dataset easily. In another aspect, the ADNI also could provide three different categories of the dataset. They are healthy control, mild cognitive impairment and early Alzheimer's diseases. Since the current research suggests that Alzheimer's disease could have bad effects on brain network [16], the study of brain connectome network properties becomes necessary and critical. To better understand the data elements for each data categories and make

the user more convenient, the thesis adds the function of 3D graph properties visualization in BECA. To reduce the cluttered results and improve the readability of visualization, the thesis also selects several critical graph properties and adopt several algorithms to achieve a useful complementary for the brain connectome network visualization in 2D. The 2D visualization tool is developed as a separate GUI based on MATLAB. Therefore, the user could make an easier comparison of the network among the subjects from different categories.

2.2 Graph Layout

With the development of brain connectome research, more and more graph-based brain datasets are collected to interpret the brain functional connectivity. So, the visualization problem for such kind of dataset becomes the graph drawing problem. The brain connectome network is a spatial model which could be visualized using the brain surface files or volume files in the tools mentioned in 2.1.

Based on the ADNI dataset, the thesis adopts both spatial and non-spatial visualization. The 3D graph visualization has achieved on previous research work of BECA. It has applied the iso-surface extraction method from MRI voxel to generate the brain network [5]. The 2D graph visualization puts more focus on revealing the patterns and clusters of dataset elements. Generally, there are six common algorithms to visualize the 2D graph. They are a Force-directed layout, Group in a box, Radial, Arc, Semantic Substrate, and Matrix Diagram [17]. These algorithms could solve most of the graph visualization problems. In the thesis, I chose to use the force-directed layout, which is the most popular one for the undirected weighted graph. The layout algorithm is based on the spring simulation which will dynamically reach to a balanced system at the end. The MATLAB has integrated the force-directed layout algorithm as a function

(https://www.mathworks.com/help/matlab/index.html). From the paper "Graph Drawing by Force-Directed Placement", Thomas M.J. Fruchterman summarized that the layout algorithm used two principles. One is arranging the nodes with direct path together. The other is the control the distance of the node depending on the node size and display space [18]. Although other criteria like minimized edge crossing, evenly distribute the nodes, etc. are still be considered, the main goals are simplified to reduce the visualization time.

Although the force-directed layout could draw the aesthetical symmetry graph automatically, the visualization result for a large graph with more than 100 nodes is still cluttered and unreadable. To preserve the original graph properties, some graph layout adjustment strategies like fisheye algorithm [19] are used to improve the visualization result. The thesis also introduces a clique related method to get better results.

2.3 Hypergraph and Clique Visualization

A hypergraph is used to describe the relations of nodes and edges. From the mathematical definition, a hypergraph H has a set of nodes N, and a set of edges or hyperedges E, $H = (N, E)$ [20]. The most difference between graph and hypergraph is that hyperedges could connect more than two nodes. Therefore, the relations between nodes could not be visualized simply using a straight line. Jesse Paquette has summarized four popular methods to visualize hypergraph in "Hypergraph Visualization and Enrichment Statistics: How the EGAN Paradigm Facilitates Organic Discovery from Big Data". They are Venn diagram method, edge-connection method, node-coloring method and extra nodes method (Figure 2.1) [21].
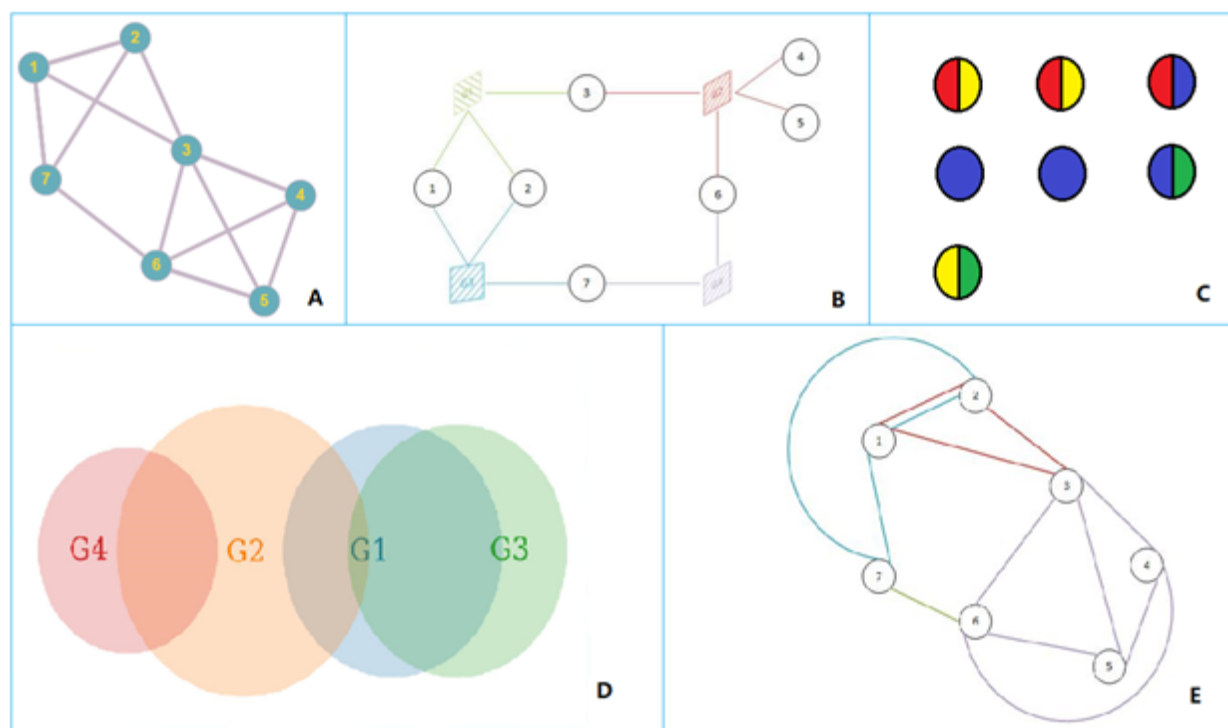
A is the original graph. B adopts the extra nodes method. C uses node-coloring method. D shows Venn Diagram. E adopts the clique method.

Figure 2.1 Hypergraph visualization methods

In graph theory, a complete graph means each node has a direct connection to the rest nodes. Clique is defined as a "maximal complete subgraph" in the graph by Luce and Perry [22]. Clique has a meaningful application in brain connectome network. Because of Alzheimer's disease, the connection of brain nerves might be damaged. Therefore, by comparing the patterns of network cliques among the normal and abnormal subjects, the factor causing the disease might have clue. Besides, with the clique visualization, the neurologists could find the abnormal of brain structure easily. So, the clique visualization could use the hypergraph visualization methods mentioned above. However, all of them have shortcomings in brain connectome visualization. Venn diagram method and node-coloring method would lose brain network information. They are just about set concepts. Edge-connection method and extra nodes method both would cause visual clutter and

confusion since the brain network has almost hundreds of nodes. In conclusion, clique analysis is critical for the brain connectome network visualization tool. A new algorithm will be introduced in Chapter 5.

2.4 Bezier Curve

Bezier Curve is a parametric curve which is also used in BECA. The Bezier curves are included in the convex hull of the control points [5] [28]. By adjusting the control points, the curve could adjust intuitively. In BECA, cubic Bezier Curves are used to replace the straight line inside of the brain to avoid visual clutter in 3D. In this thesis, it is used to connect all the nodes within the same clique in 2D.

2.5 Traveling Salesman Problem (TSP)

The hypergraph is visualized using the Bezier Curve. To find the order of node connections, there are three algorithms are discussed. One of the algorithms is about connecting several nodes in an order which has the minimum sum of the distance between nodes. Although the algorithm is about a line with different start and end nodes, it could be considered as a TSP with adding a "dummy" point.

TSP also called the shortest Hamiltonian path, is a classic problem that finding a path to visit all the cities once with minimum walk length then return to the start point. It is an NP-hard problem which could not be solved in polynomial time. There are two strategies to study the solutions. One is an exact algorithm. The other is an approximation method. Generally, the time complexity is $O(n!)$ for brute force methods. It is reduced by Held-Karp to $O(n^2*2^n)$ using Dynamic Programming method [23]. Then the time complexity is reduced to the Interval ($O(n^4)$,

O(n^3•2^n)) in 2014. Although these improvements are dramatical, the time complexity is still unfordable when the size of n gets larger. So, in some cases, the approximation TSP solution is also enough for the problem. There are many algorithms to find an optimal path, like Nearest Neighbor, Greedy, Insertion Heuristics, etc. [24] [25]. However, the solutions mentioned above are not stable enough.

Currently, the CE method is widely used which is a more reliable method for an optimal solution [26]. It is an iterative method based on two steps: random sample generation and accurate sample generation for the next iteration using a defined sample selection mechanism [27]. For CE application in TSP, the steps to generate the tour of TSP has three steps. First, randomly generate 2*n*n tours as the first iteration input. Then calculate the probability matrix for each node with the equation

$$p_{ij} = \frac{\sum_{k=1}^{N} I\{S(X_k) \le \gamma\} I\{X_k \in\}}{\sum_{k=1}^{N} I\{S(X_k) \le \gamma\}}$$

Where X_{ij} is the tour from node i to j, $S(X_k)$ is the set of all tours including node k. γ is the rarity value calculated by ρ which is "the fraction of samples to take". The iteration will be ended until $k = n - 1$. There is an extra parameter α, called smoothing parameter. It is used for reducing the variance in the P_{ij} calculation [27]. By using the approximation method, the time complexity is reduced to $O(n^3)$.

CHAPTER 3 GRAPH PROPERTIES

Graph theory is a subject about the mathematic structure: Graph [29]. In some books' definition, the graph is composed of vertex and edge. The network is composed with node and link. There are indeed no major nor minor differences between vertex and node [30]. Therefore, vertex and node are considered the same concept in the thesis.

The graph could be divided as a directed graph and undirected graph according to the properties of edges. If the edge in the graph has defined the direction from one node to the other, the graph is called a directed graph. Otherwise, it is an undirected graph. The thesis is about the brain connectome network. The definition of the edge in the network is the density of the fibers, which is a scalar quantity. So, the brain connectome network is an undirected graph.

The graph properties are about the general attributions or measures of the graph itself. Basically, it does not depend on the graph layout or graph visualization method. In the paper "Large networks and graph limits", the graph property is formally defined as "a class of graphs that is invariant under isomorphism" [31]. Graph properties are divided into two categories. One is the measures for vertices; the other is for edges. The vertices measures include vertex number, vertex degree (vertex in degree and vertex out degree), vertex independency, closeness (vertex in closeness, vertex out closeness), and so on. The edges measures contain edge number, shortest distance, edge independency, maximum flow, clique, etc. In the thesis, I chose several graph properties mainly related to the practical meaning of the brain connectome network. Therefore, five measures are picked for the brain connectome network visualization: degree, closeness, shortest distance (path), maximum flow and clique. As I mentioned above, all the measures are discussed in the undirected graph category.

In the thesis, the purpose of the data visualization is to help the neurologists to compare the Alzheimer's disease in different stages. The most obvious change of the brain structure with the disease is the value of fiber density, which is the edge weight. Therefore, visualization should present and emphasize the change patterns of edges. Firstly, node degree is the numbers of edges connecting to other nodes. In practical use, it would show which ROI has the most connections to others. The ROI with a larger degree would be the more important node. By inspecting the ROI degree, the user could compare the ROI in normal and abnormal brain structure easily. Secondly, another measure to evaluate the influence of one ROI to the rests is closeness. Closeness is the inverse sum of the distance from a current node to others:

$$c(i) = (\frac{A_i}{N-1})^2 \frac{1}{C_i}$$

Where A_i is the nodes with a connection to the current node. It is a normalized form. N is the total nodes number in the graph. C_i is the sum of distance (or edge weight) from the current node to other directed connected nodes. Therefore, if the node is isolated in a graph, the $c(i)$ value is zero (https://www.mathworks.com/help/matlab/index.html). In other words, the more influence a node has, the closer the node is to the others. Thirdly, the shortest distance (path) is find a path between two nodes with the minimum sum of weight. If the weight of edges is 1, it means to find the number of edges between the two nodes [29]. In Alzheimer's disease dataset, the abnormal brain has week connection between some pairs of nodes. So, the shortest distance value would be increased along with the reducing connections. Through the shortest distance visualization, the user could find the pattern of the variety between two nodes. Fourthly, maximum flow is about the maximum possible flow from a start node to a sink node. The definition of flow is directed graph $G = (V, E)$ with start node $s \in V$, sink node $t \in V$. The capacity is $f(v, w) \leq c(v, w) \forall v, w \in V$, where $c(v, w)$ is the capacity of the edge between v and w. The total flow is $|f| = \sum_{v \in V} f(v, t)$. The

maximum flow is to find the largest possible $|f|$ value. Generally, the maximum flow of a network might be different. But the value should be the same [29] (Figure 3.1). The max-flow min-cut theorem from Ford and Fulkerson is used to solve the maximum flow problem, which is "In any network, the value of any maximum flow is equal to the capacity of any minimum cut". Although, the definition is based on directed graph, the undirected graph could be considered as the same capacity of backwards and forwards. In this case, the flow only propagates using one of the two edges [32]. For the application of maximum flow in the Alzheimer's disease dataset visualization, the changes of the graph maximum flow could access the abnormal of fiber density. Finally, clique visualization is an overview of fiber density vary in brain connectome network. In mathematical definition, a clique is a complete subgraph where every node has a direct connection to the rest nodes. A clique should include at least two nodes. So, when all ROIs are grouped as cliques, the brain connectome network is considered as a hypergraph [21]. In the thesis, we chose three nodes as the minimum clique size to reduce the unnecessary visualization. A graph with many cliques means the connections between nodes are very strong. In the brain connectome network, the changes of clique size and total clique numbers could show how the ROI connections are weakened by the decreasing fiber density.
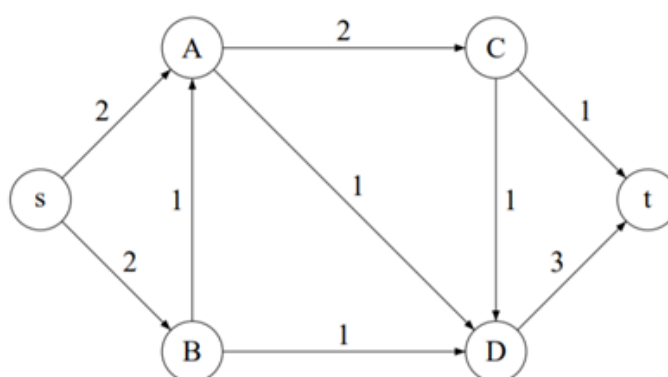
From the s node to t node the maximum flow is 4[33]

Figure 3.1 Maximum flow example

Therefore, to study the brain network by means of graph theory and make the brain connectome network visualization tool more flexible, the graph property visualization is necessary. Besides, it is known that Alzheimer's disease would cause a change in fiber density value in the brain. To visualize and emphasize the differences, the selected graph properties and measures in the thesis could reveal how the nodes (ROIs) and the edges (fiber density) would be changed by the impacts. Besides, the clique would show how the network connection is influenced by the disease.

CHAPTER 4 3D AND 2D VISUALIZATIONS

As I discussed in the previous chapters, graph properties visualization is a useful compensation for graph visualization. It could provide more information about dataset from the viewpoint of the graph. According to the type of dataset, there are two ways (3D visualization and 2D visualization) to visualize the graph properties.

4.1 Graph properties in 3D

In 3D visualization, the brain is summarized as an anatomical structure, which is easier for the user to inspect the ROI locations. In other words, the node locations are fixed. The graph visualization is mainly about how to arrange and visualize the relationship between them. According to the previous research in neuroimaging visualization, an integrated visualization tool for ADNI database is developed named BECA. BECA could explain the dataset in various ways including scientific visualization and information visualization. The 3D graph property visualization in the thesis is based on the basic brain anatomical structure visualization of BECA (Figure 4.1). The user interface of BECA is shown in Figure 4.1.A. It includes two renderers: Fiber Render and Gray Matter. The 3D brain visualization result is in Figure 4.1.B with loading Gray Matter Label Image file. To make the visualization result in the brain more flexible, the tool also provides a slider bar to adjust the opacity of the brain structure visualization result shown in Figure 4.2. The function is very important when the graph properties are also visualized inside the brain. If the user prefers to inspect more details inside of the brain, the opacity value should be reduced. If the structural visualization is the focus, then the opacity value could be increased.
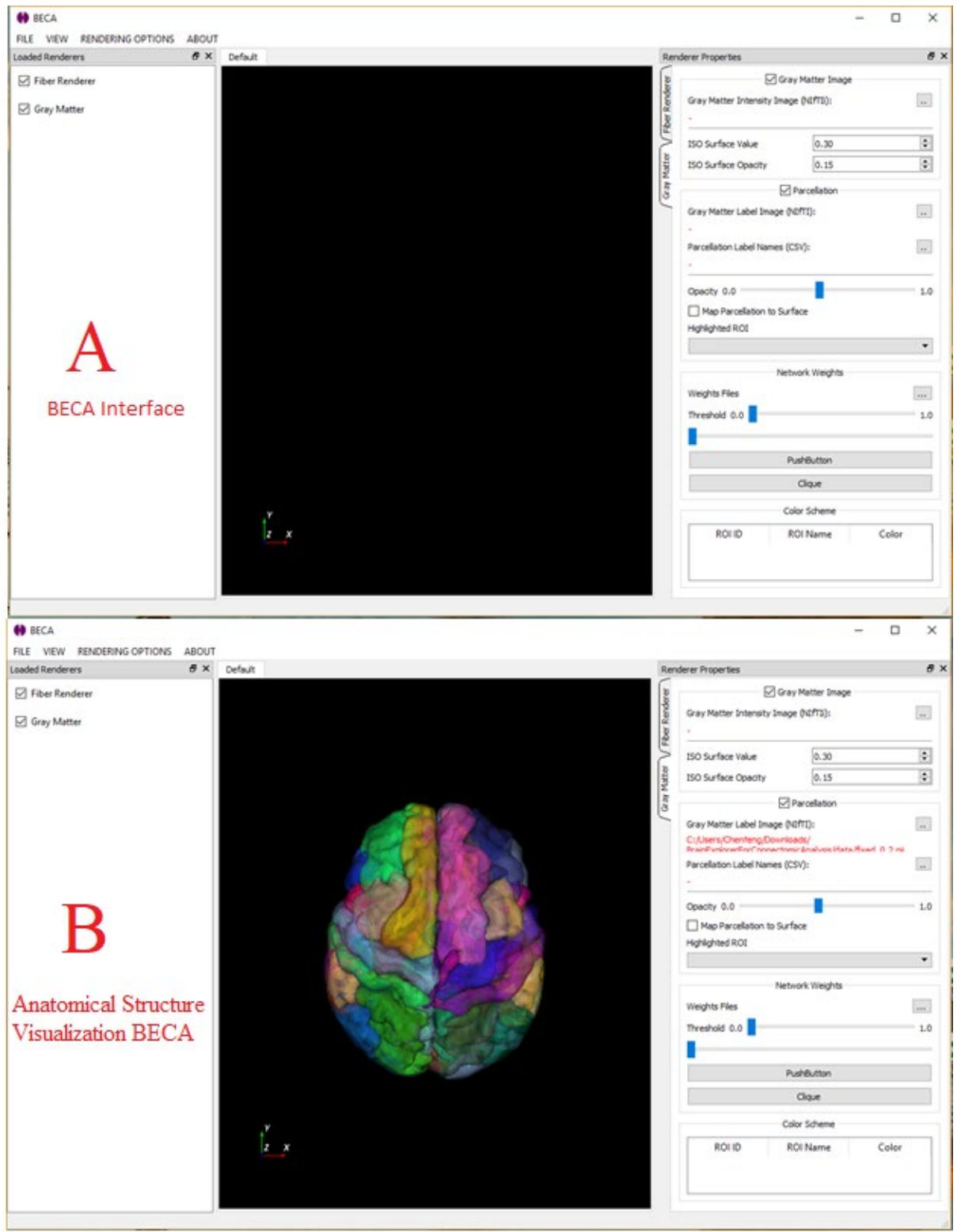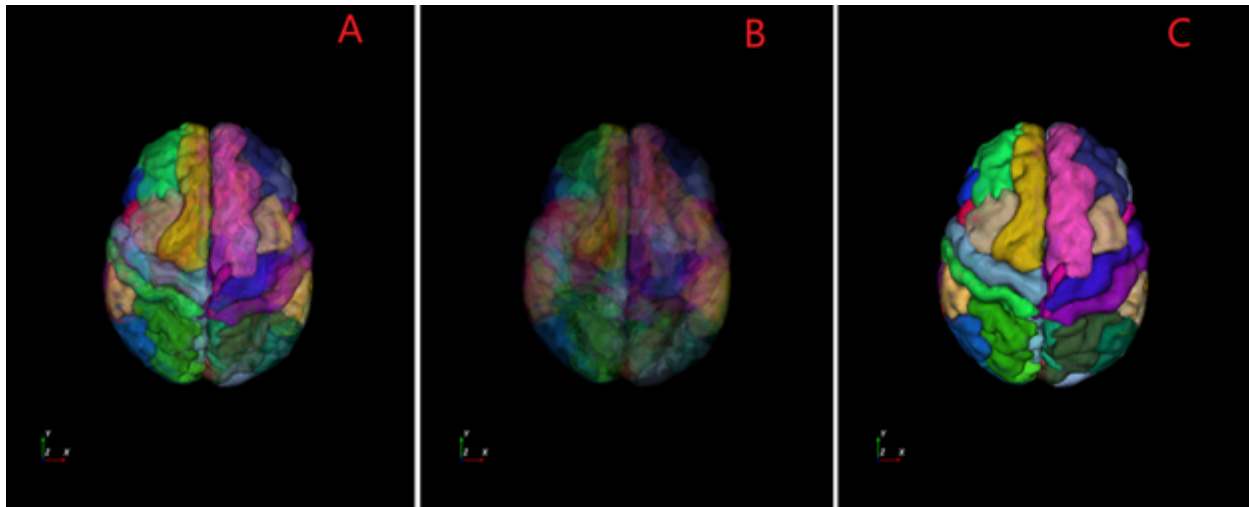
Figure 4.1 BECA

Opacity value in B is smaller than A and C. C has the biggest opacity value among the three pictures.

Figure 4.2 Comparison with different opacity

4.1.1 Degree and closeness in 3D

There are two node properties visualized: degree and closeness (Figure 4.3 with 83 ROIs). A node degree is the number of connections in an undirected graph. Closeness is the inverse sum of the distance from a current node to other connected nodes.
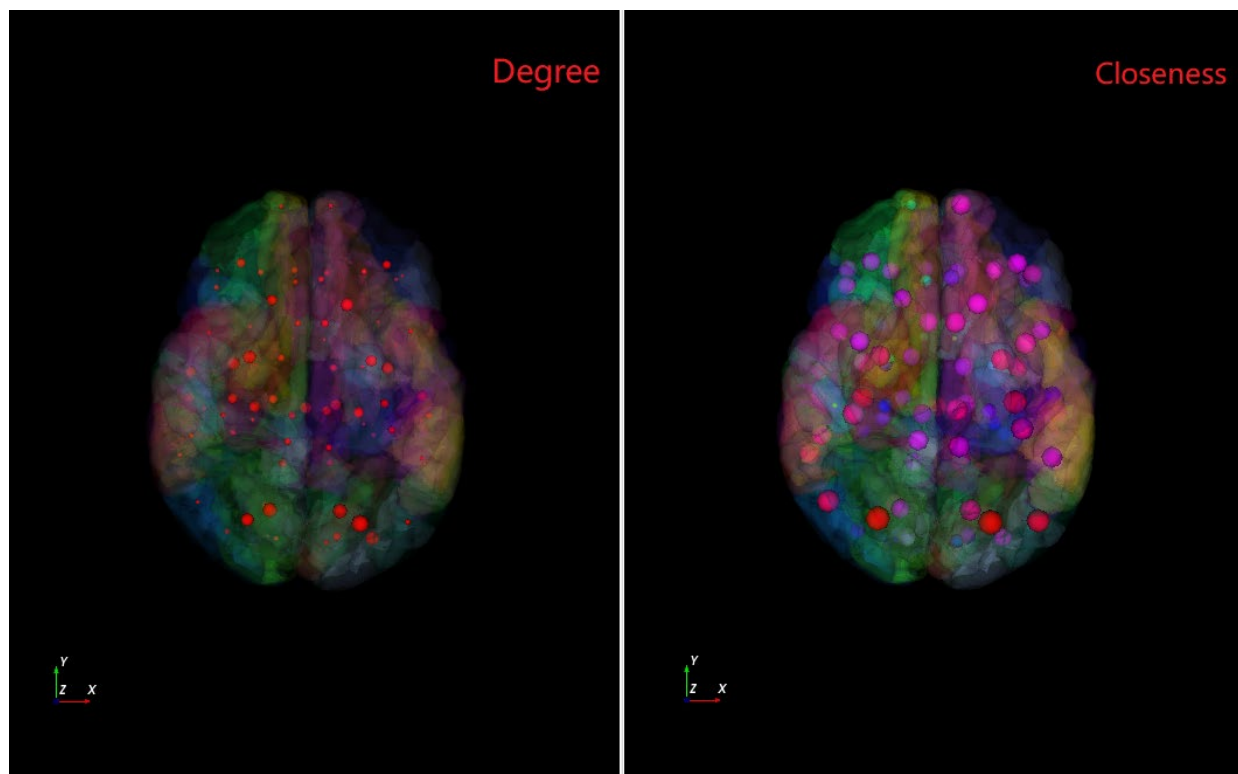
Figure 4.3 Degree and Closeness visualization

With the increase of the degree or closeness value, the node size is also increased. Besides, to stress the variance of the value changes, the color of the nodes would be also changed according to the property value. Based on the VTK function library, the function:

$$\text{Static void } vtkMath :: HSVToRGB(\text{float } h, \text{float } s, \text{float } v, \text{float } * r, \text{float } * g, \text{float } * b)$$

is used to set the node color, where h, s v and the values of r, g and b are all in the range [0,1]. A color model is a system of how people describe a color. The RGB is one of the popular models, which uses three colors: red, green and blue to generate the rest color in an image. The RGB value is necessary when the image would be shown on the screen. To transfer the degree or closeness value into RGB, the "HSVToRGB" function is called in the code. HSV is presented for hue, saturation, and value. It is also a color model. In the program, the saturation and value parameters are set to 1. The Hue value is changed according to the degree or closeness value. In Figure 4.4,

when Saturation and Value/Brightness are set to 1, the color from left to right in Hue row displays the value changes from 0 to 1. By contrast with Figure 4.3, the visualization result could be clearer. For example, in Figure 4.3, the nodes look like similar size in the closeness part. By inspecting the color, the blue node should smaller than the purple one. Therefore, the value difference could be easier to tell by applying both size and color in the degree or closeness visualization.
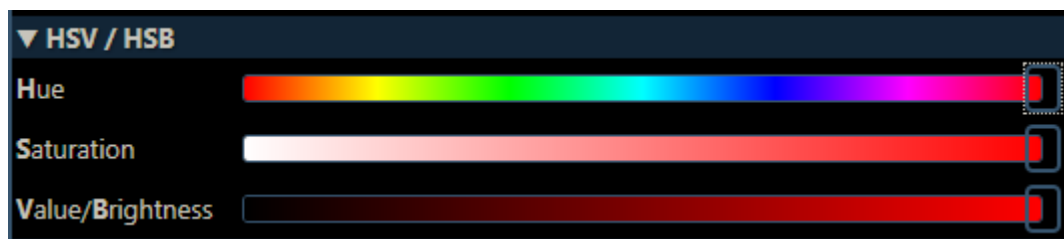

Figure 4.4 HSV chart

The above cases are the applications for studying the ROIs and fiber density pattern in the brain structure. Besides, the visualization is also helpful in comparing the difference between normal and abnormal brain dataset (Figure 4.5).

	1	2	3	4	5	6	7	8 A
1	2.6669	1.4894	0	0.7271	0.3791	0	0	
2	1.4894	1.1886	0	0	0.5013	0	6.1900e-04	
3	0	0	0	0.4324	0	0	0	
4	0.7271	0	0.4324	0.1674	0	0	0.0591	0.14
5	0.3791	0.5013	0	0	4.3132	0.1516	0.0406	0.85
6	0	0	0	0	0.1516	4.3925	0.0228	1.11
7	0	6.1900e-04	0	0.0591	0.0406	0.0228	0.6905	1.59
8	0	0	0	0.1438	0.8595	1.1195	1.5960	0.87

	1	2	3	4	5	6	7	8 B
1	1.8570	0.6726	2.4333	0.3143	0.4369	0.0057	0.1269	
2	0.6726	0	0	0	0.0951	0	0	
3	2.4333	0	0	0.6896	0	0	0	
4	0.3143	0	0.6896	0.4384	0	0	0	0.05
5	0.4369	0.0951	0	0	1.7369	0.5649	0.0397	0.00
6	0.0057	0	0	0	0.5649	1.3187	0.2906	0.45
7	0.1269	0	0	0	0.0397	0.2906	1.2187	0.94
8	0	0	0	0.0541	0.0045	0.4583	0.9481	2.32

A is normal brain data sample. B is abnormal brain data sample.

Figure 4.5 ADNI data sample

In ADNI dataset, the practical meaning for the degree is the ROI connectivity. The edge of the graph represents the density of the fibers between each pair ROI, which is the "number of tracks between them divided by the mean volume of two ROIs" [5]. If an ROI with a larger degree means that the ROI plays an important role in brain connection issue. The practical application of closeness is a little bit different from its definition. From Figure 4.5, the edge weight is reduced because of Alzheimer's disease. Then the calculation of the distance from one ROI to the others is reduced. However, the smaller edge weight means less information could be passed through. So, the closeness practical meaning is the more the closeness is, the less information the ROI could transfer. The visualization comparisons are shown in Figure 4.6, Figure 4.7 prove the above analysis. In Figure 4.6, the node is visualized according to the value of ROI degree. If we inspect carefully, some nodes are smaller in the abnormal part compared with the normal part. From Figure

4.6, it is hard to tell that the node color is changed. Then in Figure 4.7, although the size change is still not obvious. The color pattern changed dramatically that almost all ROIs turn to be red. Because the closeness value is increased. In other words, each ROI could transfer less information compared with a normal brain.
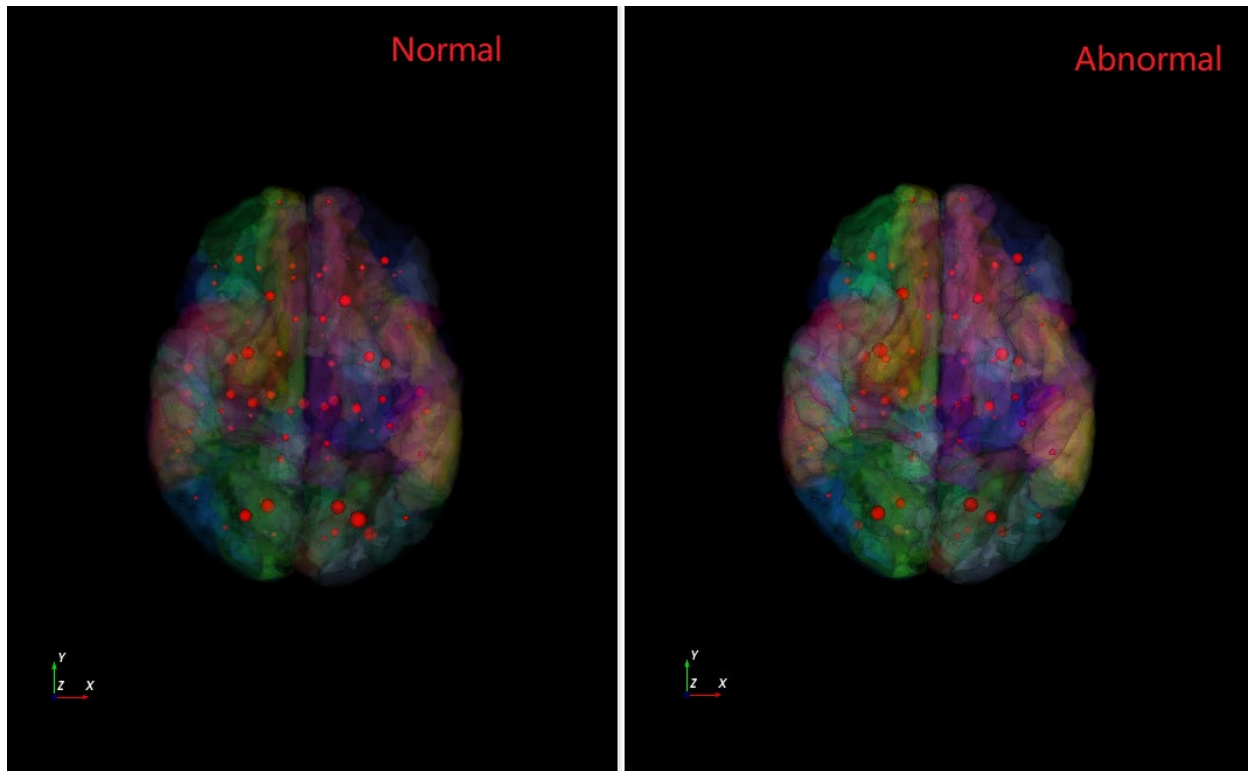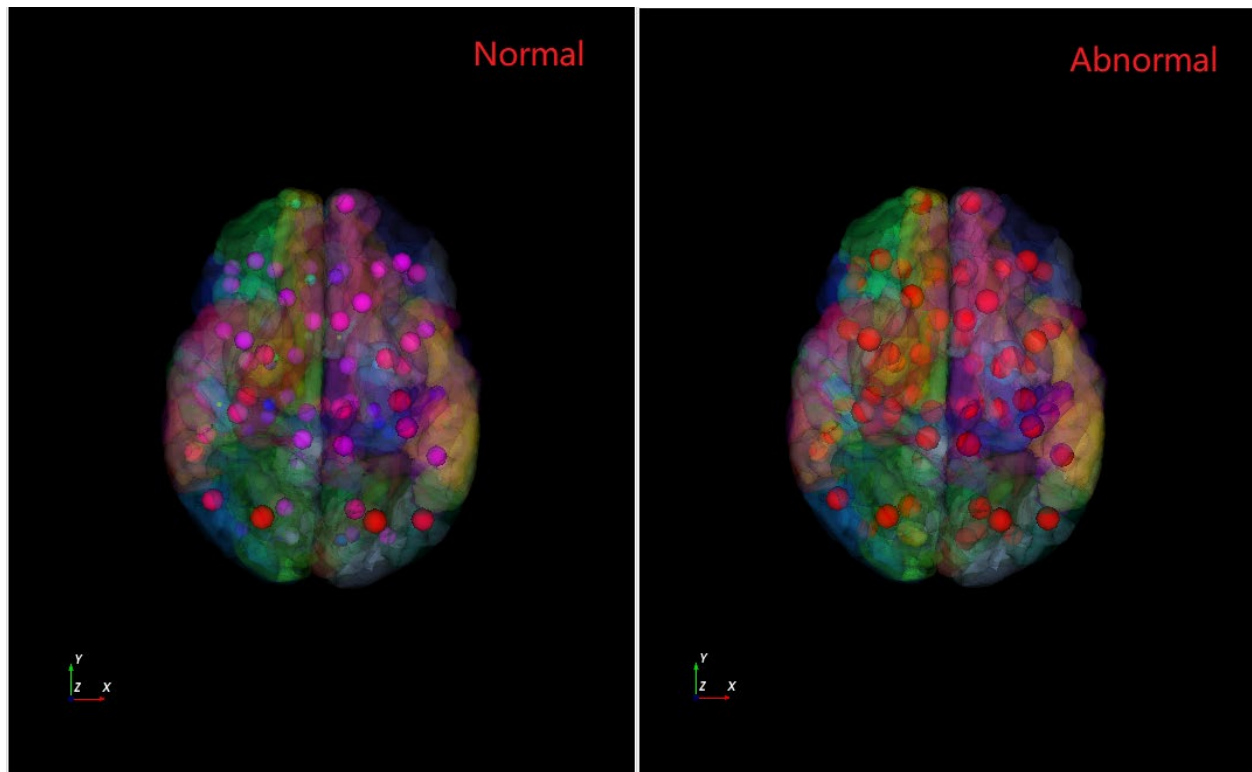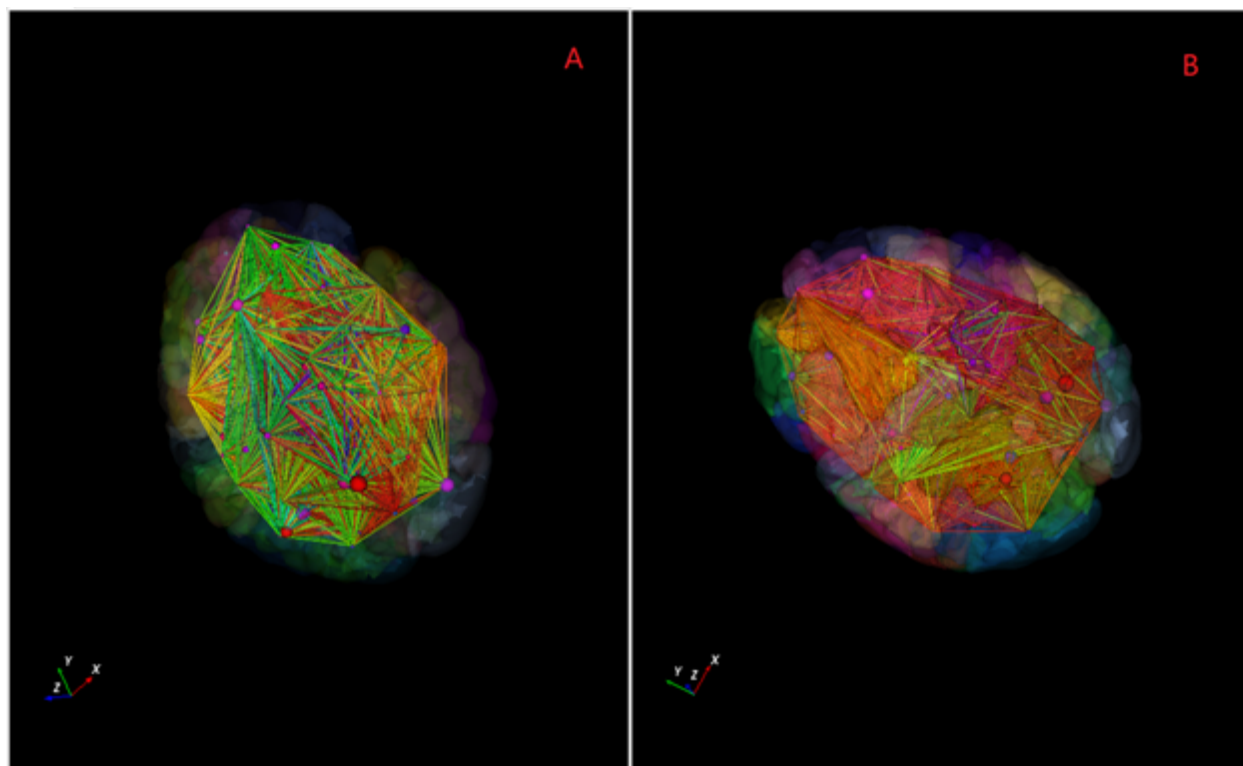

Figure 4.6 Degree visualization

Figure 4.7 Closeness visualization

In conclusion, although some of the properties visualization results are not obvious, the fiber density changes between normal and abnormal brain could be found in the final visualization by applying both color and size methods.

4.1.2 Edge properties in 3D

The edge properties like shortest path, maximum flow and clique could also be visualized in 3D to help the graph analysis (Figure 4.8).

Edge properties visualization with the nodes are visualized with closeness property. A is the maximum flow visualization. B is the shortest path visualization

Figure 4.8 Edge properties visualization

According to the example in Figure 4.8 with 83 ROIs, each node has at most 82 edges connect to the rest nodes. Obviously, in the 3D visualization, it is difficult to tell any pattern of edge properties except for the colors roughly. In clique visualization, the problems would be more severe if an extra node is introduced to present a cluster. Therefore, the edge properties in 3D visualization might need other visualization methods.

4.2 Graph properties in 2D

In 2D visualization, all the node locations are not fixed in a brain space. To reduce the vision clutter like 3D, they would be rearranged according to the visualization purposes. The force-directed graph layout is a better choice to meet the requirements, which would dynamically set up

a spring balanced system. A 2D visualization tool based on MATLAB is developed for the needs. The tool interface is shown in Figure 4.9.
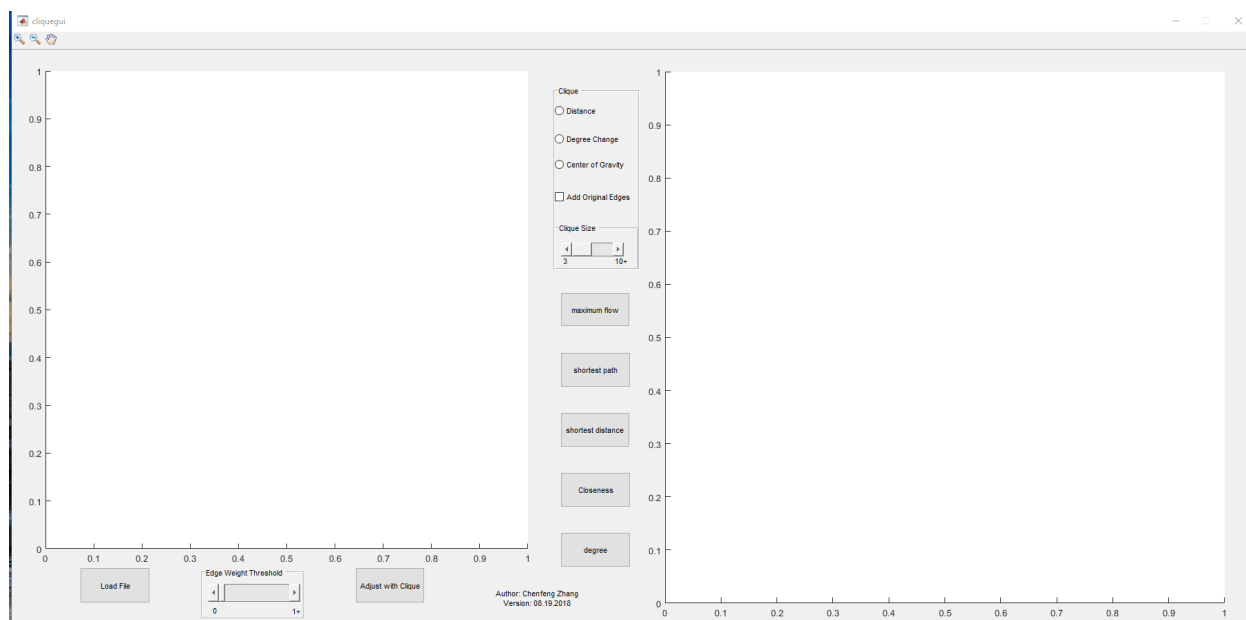

Figure 4.9 2D visualization tool user interface

The 2D visualization tool is composed of two parts: graph visualization and graph property visualization. Graph visualization is mainly at the left side in Figure 4.9 including data loading button, adjust button, threshold slider, and the 2D plot. The graph property visualization is at the right side of the interface containing the five graph properties button, three algorithms, clique size slider, and plot area.

4.2.1 Graph Layout in 2D

As mentioned in previous chapters, the force-directed layout is chosen as the graph layout method. Therefore, in MATLAB the functions:

$$h = plot(G, 'Layout', 'force', 'NodeLabelMode', 'auto');$$

$$layout(h, 'force', 'WeightEffect', 'inverse')$$

are called to display the layout in the left axis, where the edge length is inversely proportional to the edge weight (https://www.mathworks.com/help/matlab/index.html). Because the definition of edge weight is fiber density. The larger the fiber density is, the shorter length is between a pair of nodes. The layout result is shown in Figure 4.10. It is easy to find out that several nodes have stayed away from the rests. It means that these nodes have fewer connections with others. However, most nodes are still stacked together. It is difficult to find a specific node or tell the patterns of the node's relationship.
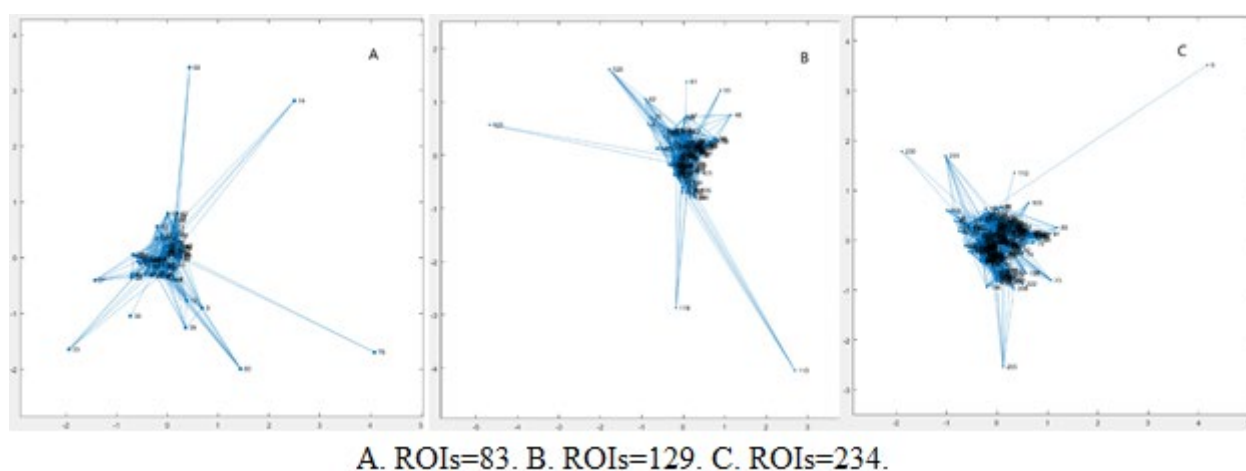

A. ROIs=83. B. ROIs=129. C. ROIs=234.

Figure 4.10 Fore-directed Layout in 2D

4.2.2 Degree and closeness in 2D

With the same definitions as 3D, degree and closeness are also visualized in the 2D tool in right side axis (Figure 4.11). The node size is used to represent the value of the degree. But most of the nodes stack together which causes vision clutter for the user.
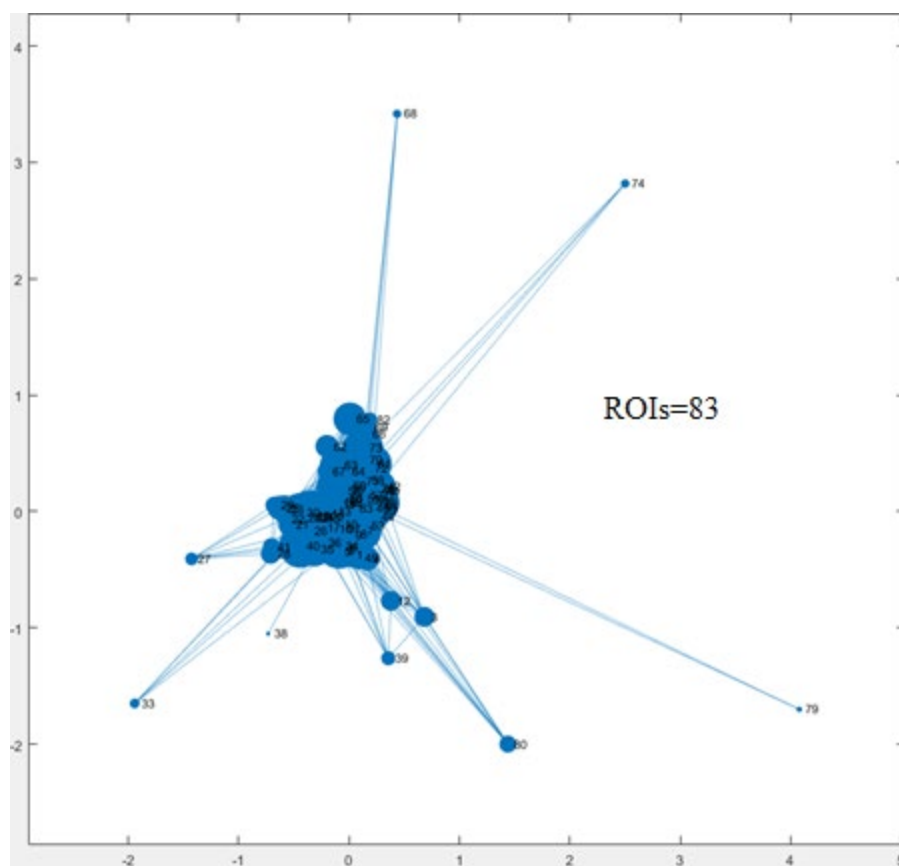
Figure 4.11 Degree visualization

4.2.3 Edge properties in 2D

As for edge properties, the 2D visualization tool provides the visualize method for maximum flow, shortest distance (path) and cliques. Besides, to solve the vision clutter problems mentioned in 4.2.1 and 4.2.1, the clique concept is applied in layout and node properties visualization.

In 3D visualization, I used a colored line to connect the ROIs to visualize the edge properties. In the 2D tool, the color map method is adopted (Figure 4.12). Both the x-axis and y-axis are the node index. Therefore, according to the color bar on the left side for each color map, it is easy to see the pattern of the two properties.
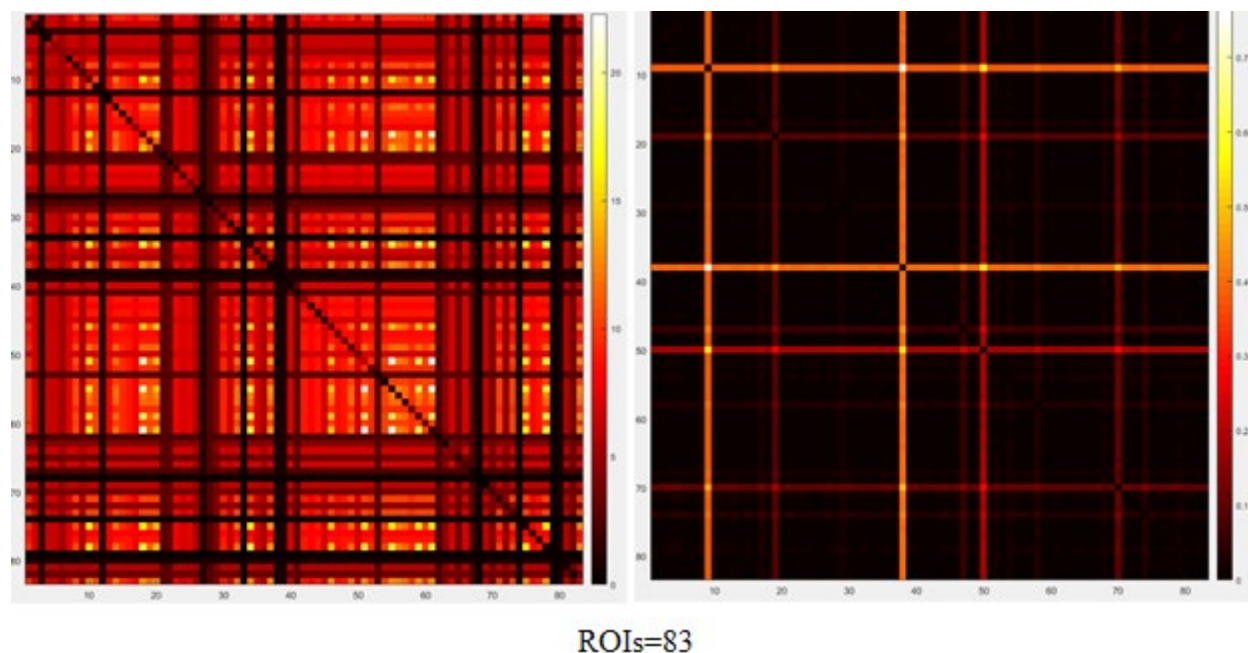
ROIs=83

Figure 4.12 Maximum flow and the shortest distance

Besides, to inspect the shortest path between the selected nodes, the 2D tool also provide the input function to customize the visualization result (Figure 4.13). In the case in Figure 4.13, when input ROI 1 to ROI 83, the tool could visualize the shortest path from 1 to 83. It is clear that when the slider increased which means some edges disconnected, one would pass more nodes to reach the other.

The clique visualization example is shown in Figure 4.14. The clique visualization applies a colored "belt" to connect all the nodes in the same clique. There are three algorithms are used in the tool. In the next chapter, clique visualization and algorithms would be discussed in detail.
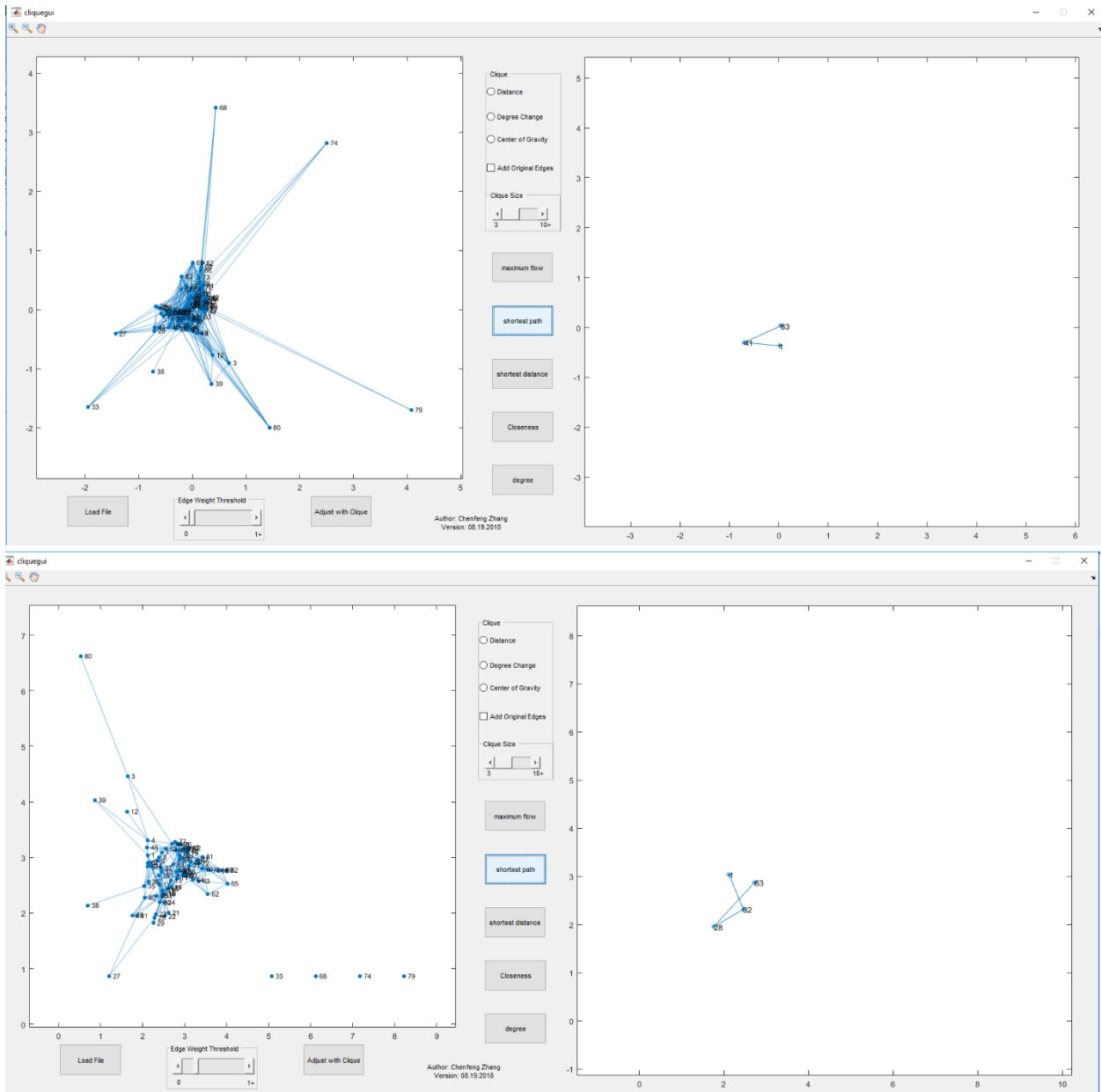
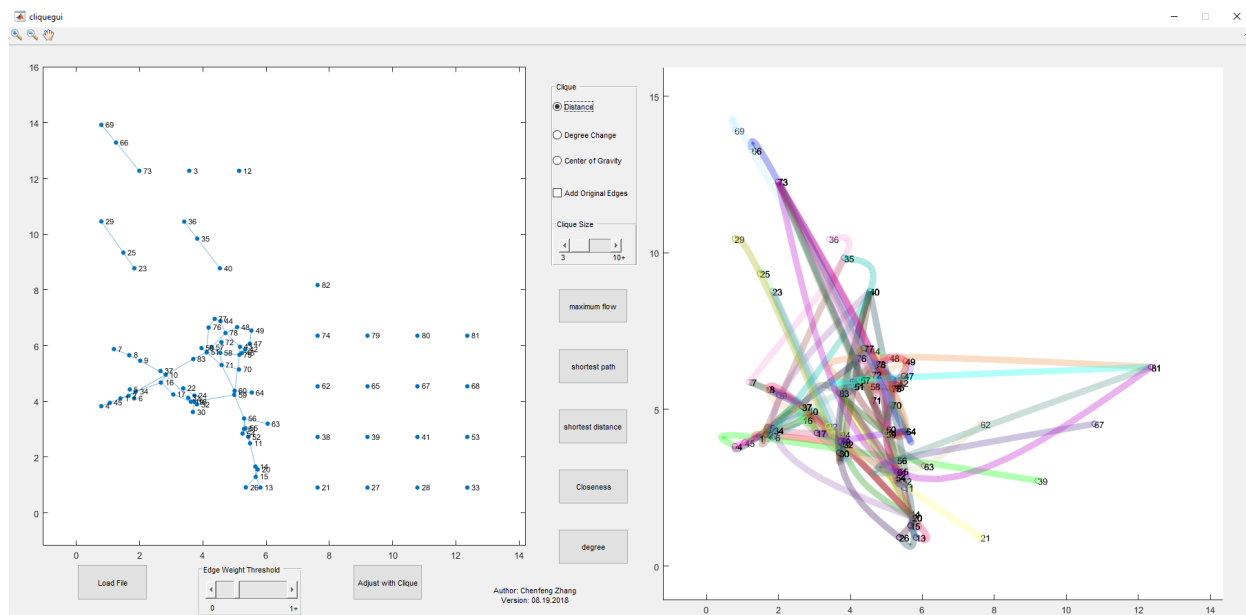Figure 4.13 Shortest path from ROI 1 to ROI 83 comparison w/o threshold

Figure 4.14 Clique visualization ROIs=83

4.2.4 Normal and abnormal dataset comparison in 2D

As the same application as BECA in 4.1, the 2D visualization tool could also be used to compare normal brain and brain with damage. In 3D visualization, the difference is only detected in closeness visualization result. In 2D, I could find more clues due to the various visualization ways. Firstly, the graph layout could display the difference easily in Figure 4.15. Even the most nodes are still crowded together, it is obvious that ROI 38 is disconnected from the others although the threshold is set to zero. The practical meaning is that the ROI could not transfer any information in the brain structure. By visualization the maximum flow and shortest path together, the influence could be clearer (Figure 4.16). In the shortest distance picture, a lot white party are displayed in ROI 38. Then in Figure 4.17, the shortest path from ROI 1 to ROI 83 also get impacted. Without any threshold, now the path has 5 nodes. Normally, it only contains 3 nodes in 4.13
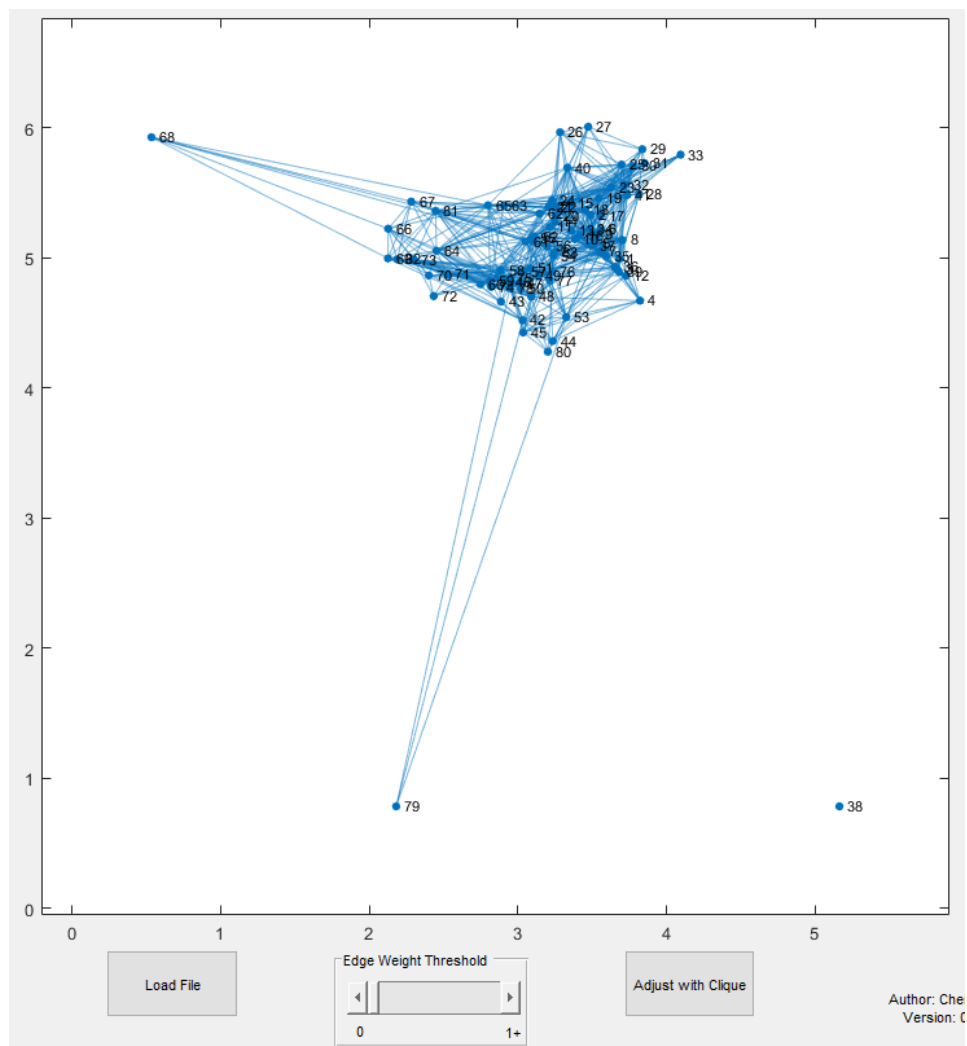
Figure 4.15 Force-directed layout with brain dataset inflicted by Alzheimer's disease
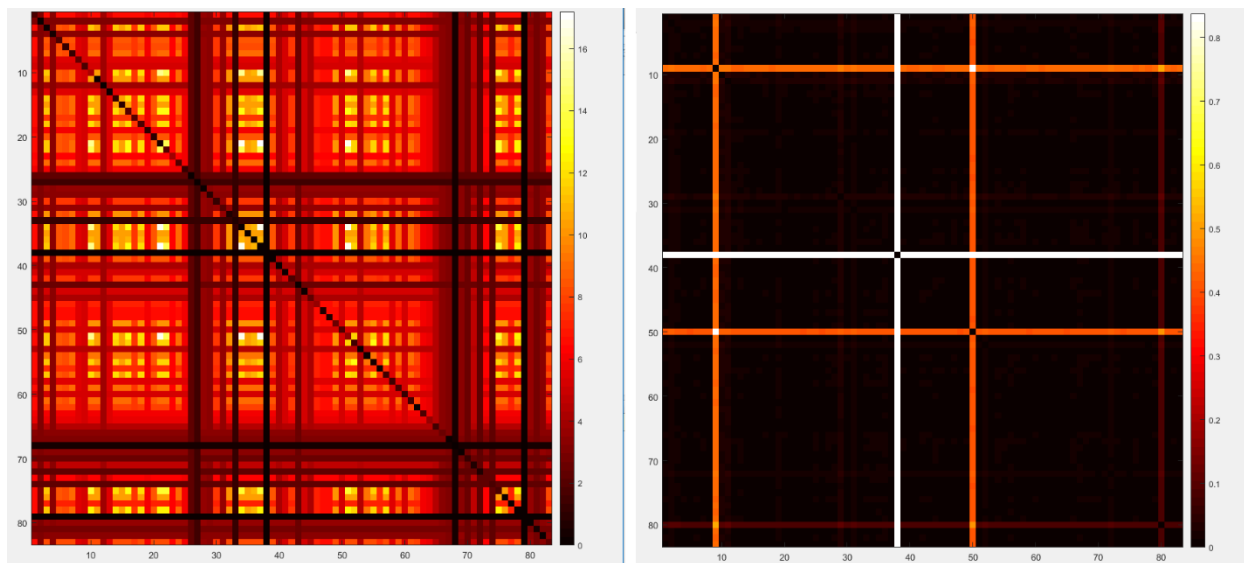
Figure 4.16 Maximum flow and shortest distance for brain dataset inflicted by Alzheimer's disease
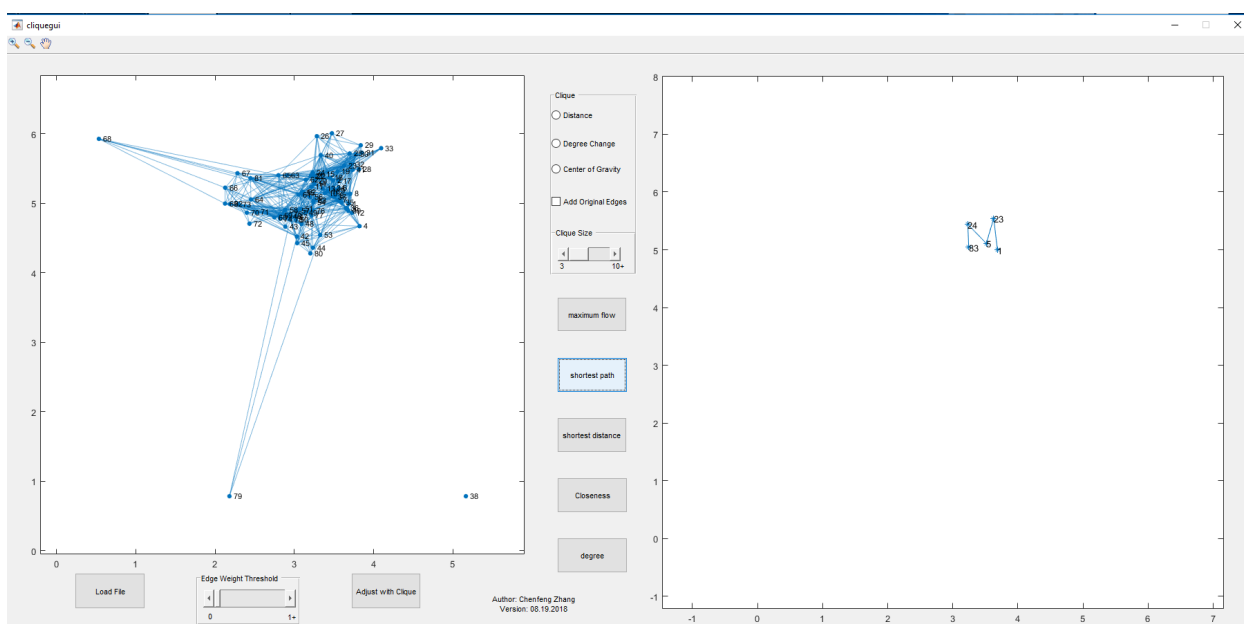

Figure 4.17 Shortest path from ROI 1 to ROI 83 using brain dataset inflicted by Alzheimer's disease

In conclusion, both 3D and 2D visualization could visualize the graph properties. In the 3D visualization, the user could inspect the visualization result with the brain anatomical structure at the same time. It provides a more intuitively way compared to the 2D result. However, most result

causes vision clutter. It is impossible to tell the pattern of edge properties. By contrast, although the 2D visualization tool also has blurry problems in graph layout visualization and node properties visualization, the edge properties visualization result could apparently reflect the data information. Therefore, in chapter 5, the clique method is introduced in 2D graph layout and node properties visualization to solve the problems.

CHAPTER 5 CLIQUE VISUALIZATION

Clique is one of the important graph properties in graph theory. In mathematics, it defines as a subset of nodes where every pair of nodes has directed connection between each other. In other words, a clique is considered as a group of nodes with strong connections. In practical use, such as social network, clique study could describe a social community where everyone knows each other. In our dataset, a clique is also useful. Due to the characteristics of Alzheimer's disease, the total clique numbers and the nodes number in one clique would be affected by the disease. In this chapter, I would explain how clique improves graph layout, how to visualize clique and the comparison between normal and abnormal brain structure from the clique perspective.

5.1 Force-directed layout adjust with cliques

According to the properties of the dataset, the force-directed layout is adopted in graph visualization in the thesis. However, from the graph visualization result in chapter 4, like Figure 4.10, most nodes are crowded together which causes visual clutter. To maintain the properties of the graph and show the clique properties, a clique is applied as an extra condition in node relations for graph layout.

In the 2D visualization tool, the button "Adjust with Clique" is used to reduce the visual clutter of graph visualization. Firstly, all the clique would be calculated with the current threshold of edge. Then all the edge weights multiples with 0.1 to increase the distances among nodes (Figure 5.1 B). Because according to the practical definition of fiber density, the smaller the edge weight is, the bigger distance of the pair of nodes should be. Besides, to emphasize the clique relations, a parameter of 2 is multiplied to the nodes within one clique (Figure 5.1 C). Therefore, in the graph

visualization, all the nodes are separated. Meanwhile, the nodes within one clique could stay close. The multiplier number is adjustable. But to separate the nodes, it should first reduce the weights, and then increase it only for cliques. The adjustment has three benefits. First is to reduce the visual clutter. Second is that the graph layout embodies the clique property. For example, it is clearly in Figure 5.1 C that node 38 does not belong to any cliques. Besides, when the edge threshold is zero, the node 44, 48, 49, 76, 77 are within one clique. In Figure 5.1 A, it is even hard to find these nodes. In Figure 5.1 B, we could find all the nodes easily (marked with red line). But due to the parameter, all the nodes within the clique are separated. To emphasize the clique relationship and prepare for clique visualization, the nodes in the clique are gathered in Figure 5.1 C (marked with red line). By contrast, in Figure 5.1 B, the nodes within the yellow line are not in the same clique. In Figure 5.1 C, the distance between them is enlarged. Last advantage of such adjustment is that the clique visualization could be improved since the node locations in graph layout would be used in clique visualization. If the nodes within one clique could stay close, the line connecting the nodes would be easier to draw smoothly. The example could be found in 5.2.
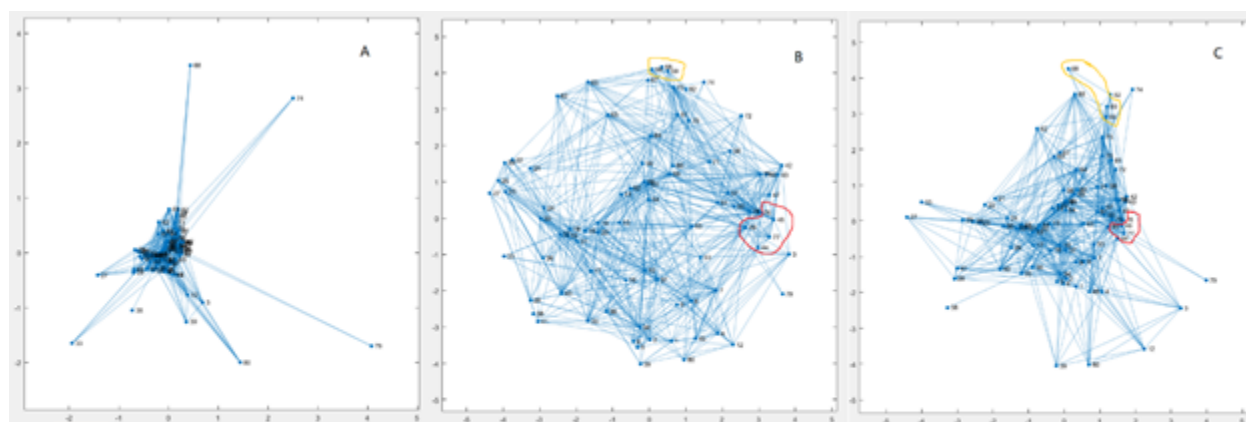
A is original data layout. B is with 0.1 where all weights reduced, so restriction among nodes are reduced too. C increases the restrictions for nodes within cliques.

Figure 5.1 Graph layout with different parameter

Moreover, the degree and closeness visualization could also be improved by using the adjust location of the graph layout (Figure 5.2).
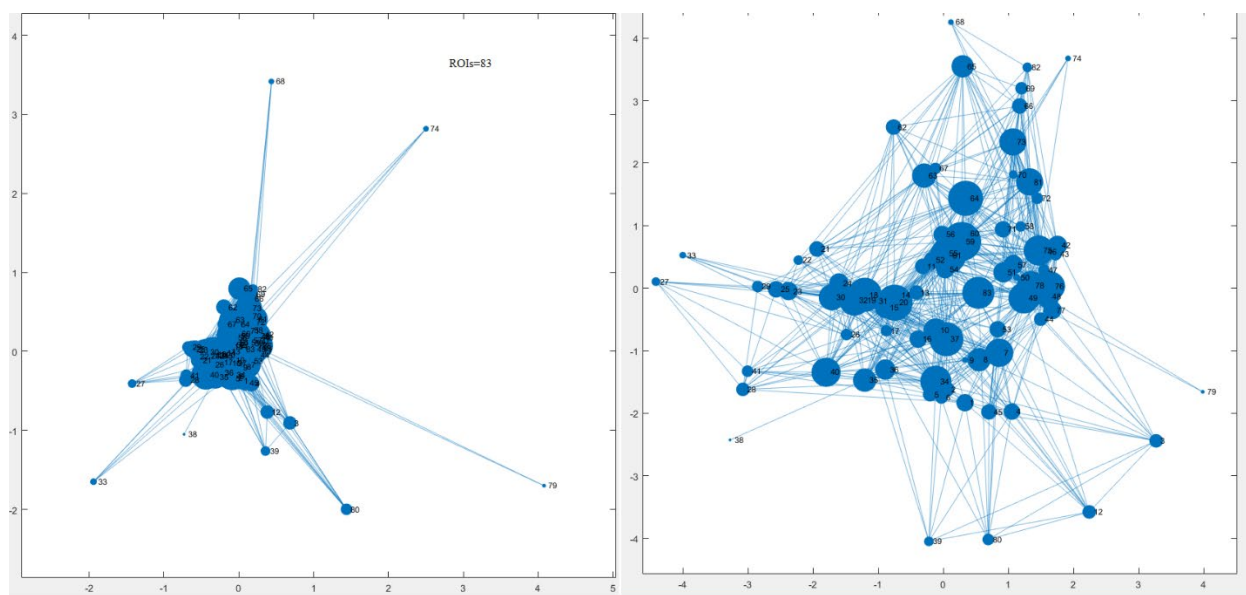

Figure 5.2 Improved degree visualization

Besides, the 2D visualization tool could also label ROI name expect for node number (Figure 5.3).

5.2 Clique visualization

As I discussed in in Chapter 2, brain connectome could be considered as a hypergraph when the ROIs are grouped as the cliques. In a clique, every node has a directed connection with others. The most intuitive way of clique visualization is using a line to connect each pair (Figure 2.1E) However with the growth of node numbers, the lines would mix together. Another way is adding extra nodes to present all the nodes within a clique. However, the introduced node would cause confusion. Besides, the newly added node also needs to connect to the nodes in the clique with lines too (Figure 2.1 B). It means more lines are needed too. To reduce the redundant lines, a new method is introduced in the thesis. By using the Bezier curve, all nodes within one clique would

be connected without extra nodes or lines (Figure 5.3). According to the definition of the Bezier

curve, the orders of node connection would infect the shape of the line. There are three methods

to decide the connection orders in the thesis, shortest distance, degree change and center of gravity.
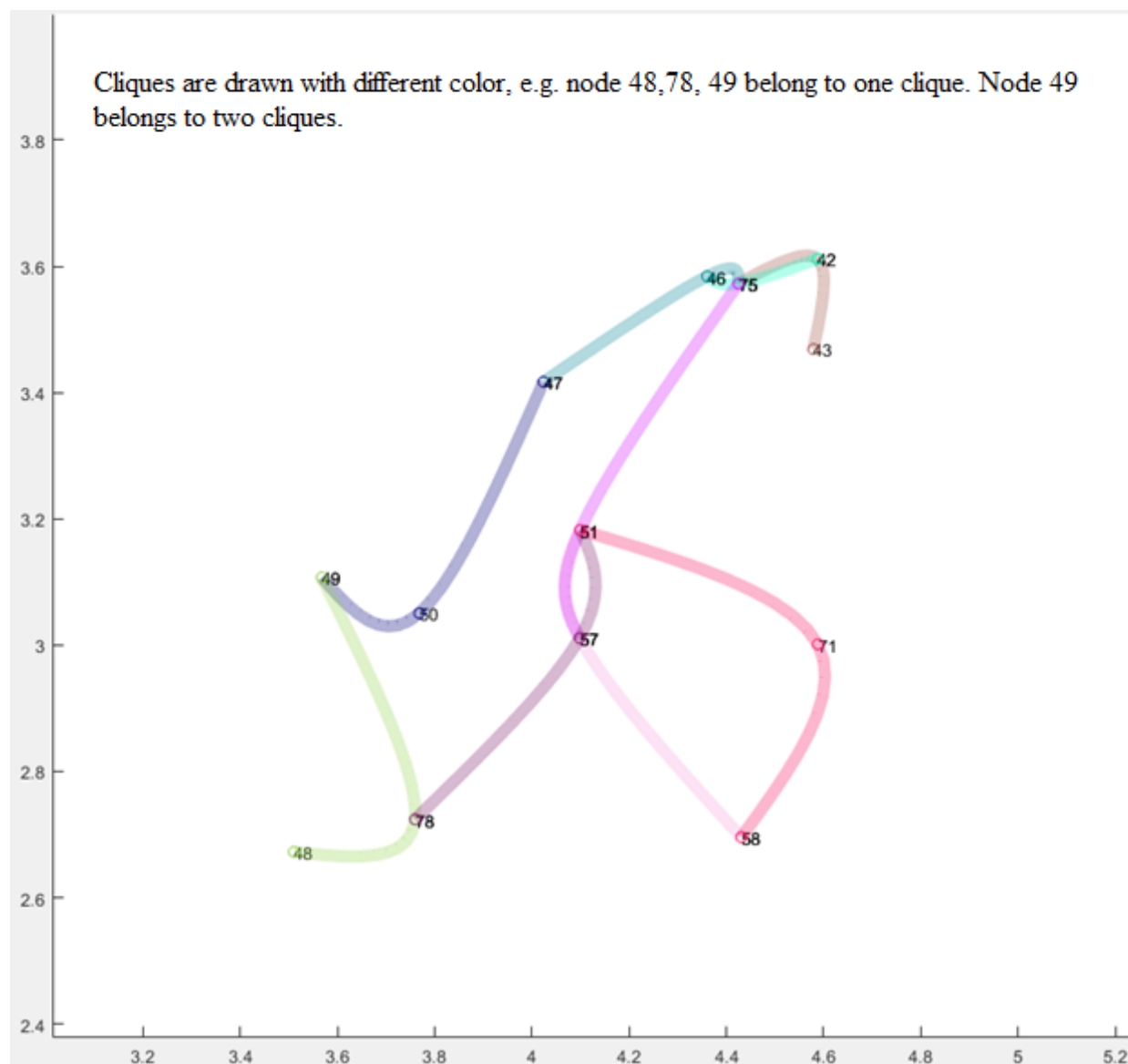

Figure 5.3 Clique visualization example

5.2.1 Shortest distance

The shortest distance means that the sum of the distance between all nodes is minimum. In that

cases, the visualization could use less line to connect all the nodes which could reduce the influence

caused by cross lines. By searching the typical problems, it is similar to the traveling salesman problem (TSP) or the shortest Hamiltonian cycle problem. TSP focuses on searching a path which starts from one node, visits all nodes once, then goes back to the beginning point. The Hamiltonian cycle is the same problem of TSP which will visit each node exactly once.

However, in the thesis, all the nodes are connected by a "belt" that the start node and end node are different. To solve this problem, a dummy point is added to convert the belt as a special case of TSP. Firstly, assume the dummy point has zero weight to the rest nodes. Then, return the solution of TSP for all nodes including the dummy point. Last, cut the circle at the dummy point, which turns the circle into a line. Now, the "belt" problem turns into searching the solution of TSP.

TSP is an NP-hard problem. The exact algorithm of TSP requires the time complexity $O(n^2 * 2^n)$. When ROIs = 83, the maximum clique size is 10, therefore the maximum time is $O(N * 100 * 2^{10})$, where N is the total clique number. In MATLAB, the algorithm calculation time is over 5 minutes. So, the response time of the algorithm is very long.

Except for the exact algorithm, there is an approximation method called CE algorithm. The heuristic function of the algorithm could find an approximation solution quickly. There are three steps of the approximation algorithm. Firstly, initialize the probability matrix $P^{(1)}$, which every node has the same probability. In other words, the sum of each row in the matrix is equal to 1. The star x_1 is initialized as 1, which represents an iteration will begin from node 1. The iteration counter k sets to 1. Secondly, according to $P^{(1)}$, obtain $P^{(k)}$ using the equation in chapter 2. Normalize the sum of rows into 1. find x_k and added to the tour. Thirdly, if k is not equal to $N - 1$, keep doing the iteration of the second step. In practical programming, the CE algorithm has a stopping criterion using stopping parameter d. If the threshold value of distance calculated by ρ is keeping the same value for d times, the algorithm would stop. The choice for sample size N and

ρ depends on the size of the problem. The experience value of N is $2 * n * n$, where n is the node number. The ρ is 0.01. Besides, to limit the variance of the probability matrix for each iteration, a smoothing parameter α is also applied in the algorithm. The time complexity of the algorithm is $O(n^3)$. The method is mainly for larger size nodes. Although, when $ROIs = 83$, most clique sizes are around 4 to 5, the performance would be dramatically improved when the ROIs is increased, or the clique size grows larger. In Figure 5.4, the right axis displays the clique using the above algorithm with clique threshold 10. The total calculation time is less than 1.5 seconds. In the visualization result, there is no cross line at all.
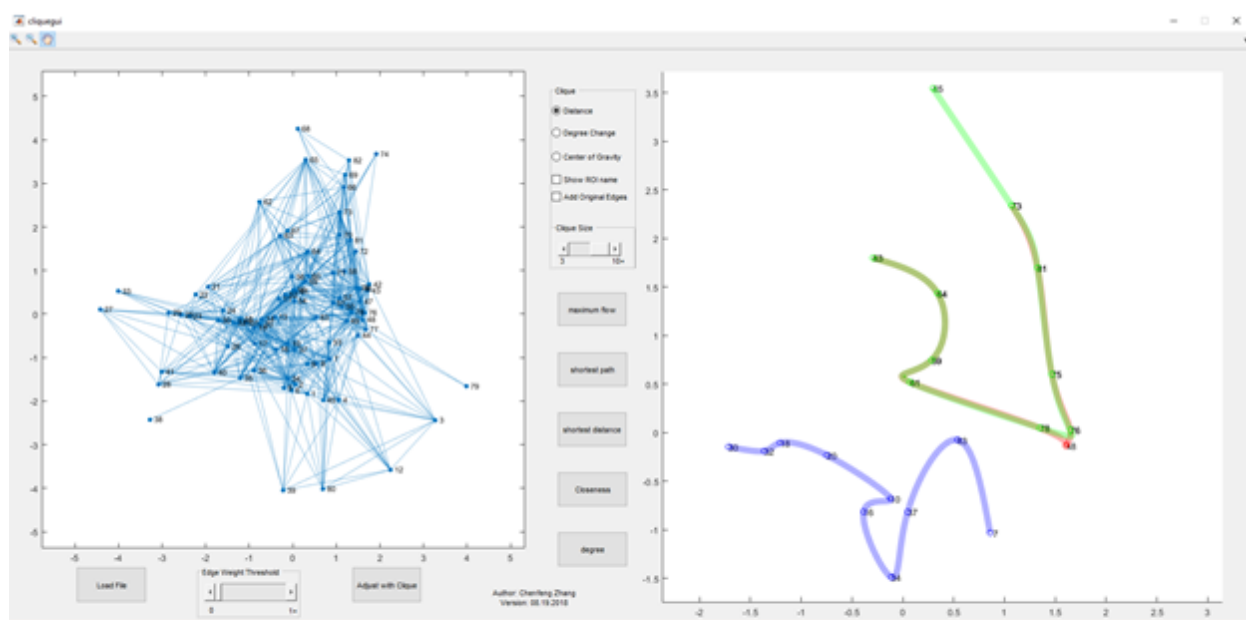
There are three cliques: (7,10,16,18,20,30,32,34,37,83),

(48,59,61,63,64,73,75,76,78,81), (59,61,63,64,65,73,75,76,78,81).

Figure 5.4 Shortest distance algorithm

5.2.2 Degree change algorithm

To make the connection curve smoother. The second way to decide the order of node is adopting minimum the degree change method. For example, in Figure 5.5, there are three nodes A, B, C.

The degree change from line CA to line AB is the angle α. The degree change from line AB to line BC is the angle β. The degree change from line BC to line CA is the angle γ. Therefore, for the node order ABCA, the degree change is $\beta + \gamma$. For the node order ABC, the degree change is β. In the thesis, the line is not return to the start node, the calculation would not include the degree change from the last node to the start node.
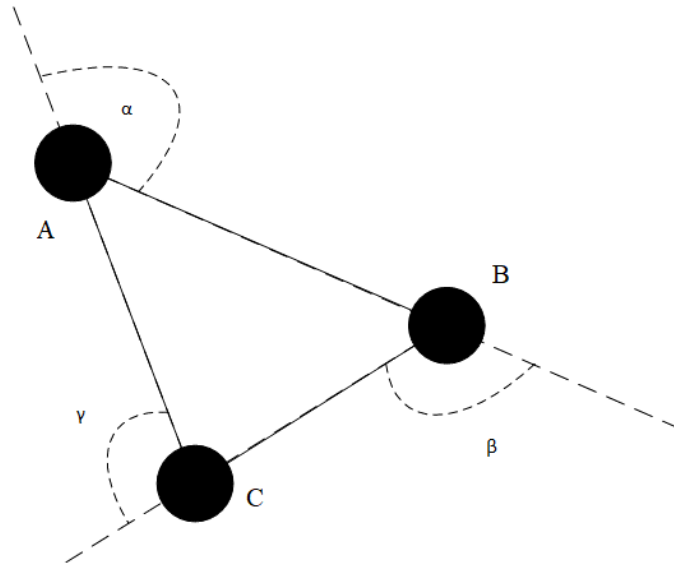

Figure 5.5 Degree change algorithm example

To calculate the minimum sum of the degree change, the thesis applied Law of Cosine:

$$a^2 = b^2 + c^2 - 2 * b * cosA$$

where a is the distance between node A and B, b is the distance between node A and C, c is the distance between node B and C. By using the equation, the $\angle CAB$ will be calculated. Then $\alpha = \pi - \angle CAB$.

There are three steps for the algorithm. Firstly, find all permutation of current nodes. Secondly, calculate the sum of degree change and compare with current minimum degree change value. Thirdly, check if all permutation has been searched. If not, return to step two to continue. The time complexity is $O(n!)$. Although the two permutations 1234 and 4321 are different, they represent

the same line. So, time complexity is $O((n-2)!)$. According to the algorithm, the nodes would be connected by smooth curves like the red curve in Figure 5.6. However, the method also has some shortcomings. For instance, the blue curve has two sharp connections in Figure 5.6. Because although the degree change for nodes 18,30,32 and nodes 37,34 16 are over 90 degree, the degree change for rest nodes are very small. In this case, the sum of the degree change are still the minimum. So, the connections could not be smooth for all nodes.
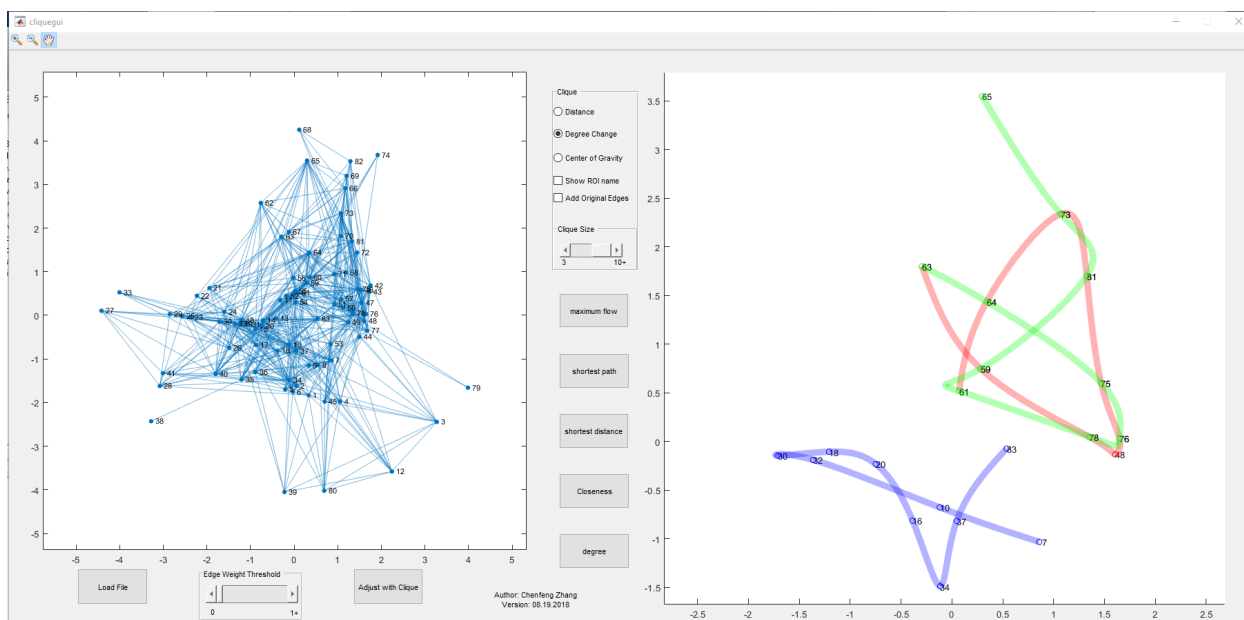

Figure 5.6 Degree change algorithm

5.2.3 Center of gravity algorithm

Center of gravity algorithm adopts the center of gravity to generate the nodes sequence for clique visualization. The core method is searching the minimum sum of degree changes among nodes and center of gravity. It also tries to connect all nodes with lines as smoother as possible. For instance, in Figure 5.7 the center of gravity for all nodes is D. The center of gravity change from A to B is α. The change from B to C is β. The change from C to A is γ. Then the sum of the three value is the center of gravity for the node order ABCA. Similar to degree change in 5.2.2, the path

in the thesis is not returned to the start node. The sum of center gravity only counts $\alpha + \beta$ for node order ABC.
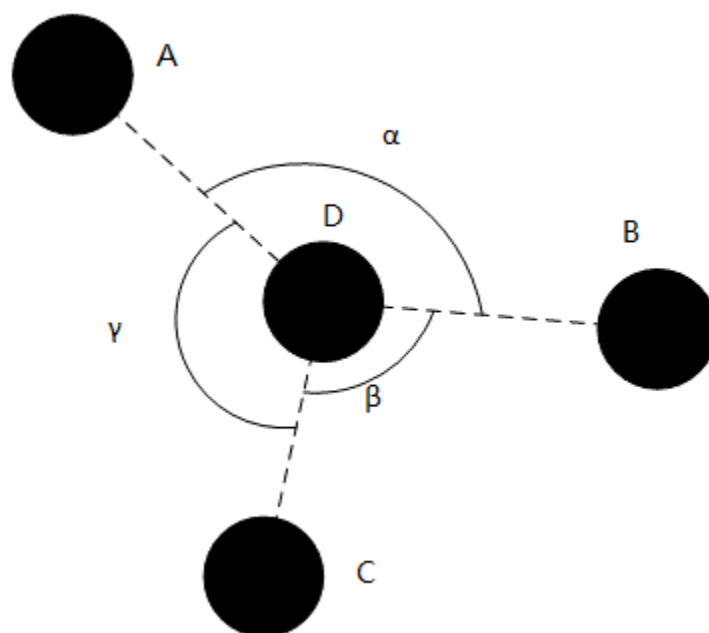

Figure 5.7 Center of gravity algorithm example

The calculation of the center of gravity changes also adopts the Law of Cosine. The coordination of D is the arithmetic mean of all nodes. Then by using Law of Cosine in triangle ΔADB in Figure 5.7, α could be calculated.

The steps of the algorithm are the same as the degree change algorithm except for calculation content. Besides, node order ABCDE is the same line same as node order EDCBA. Therefore, the time complexity for the algorithm is also $O((n-2)!)$. Figure 5.8 shows the clique visualization result of center gravity algorithm. It is clearly that all the connection tried to bend to a center point. The connection curve looks also smoother with less cross line compared with degree change algorithm. In MATLAB, the running time of Figure 5.8 is about 20 seconds. The connections among nodes 18, 30, and 32 is not as smooth as other connections. Because the center of gravity is near node 10. If the node order is 32, 30, and 18, the sum of degree changes to the center of

gravity includes overlap area compared with the current order. So, such an algorithm could not guarantee smooth curves for all connections.
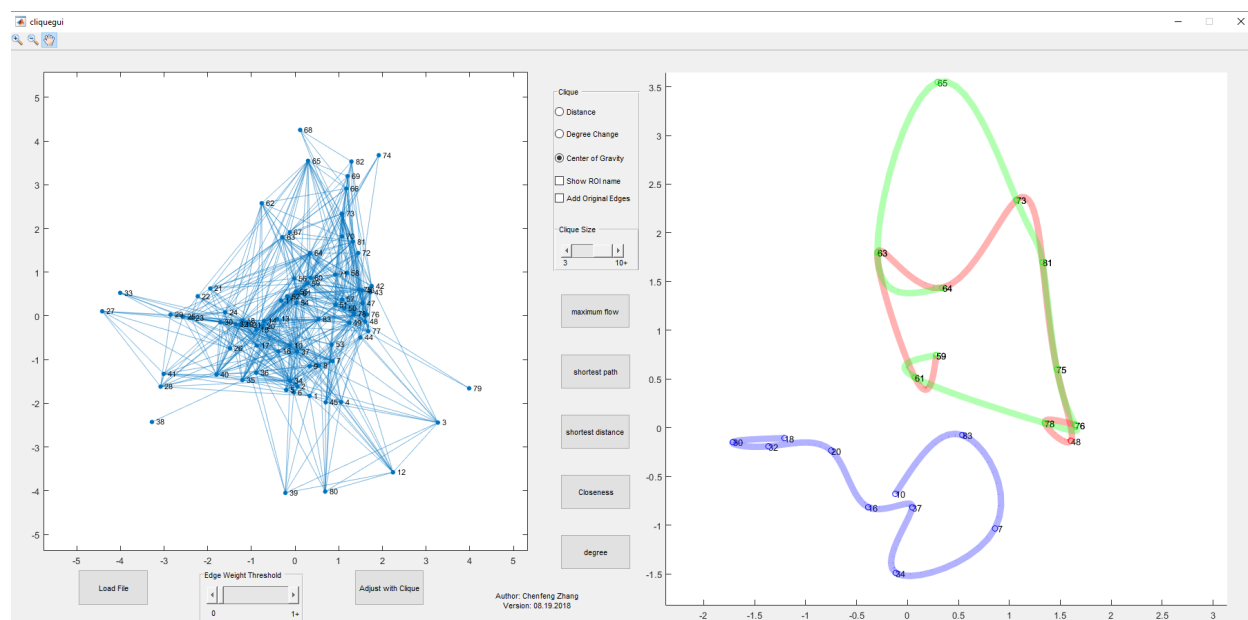

Figure 5.8 Center of gravity algorithm

5.3 Comparison using clique

According to ADNI dataset, the cliques may vary due to different subjects. However, by comparison, the clique visualization, some clue could be found. Figure 5.9 and Figure 5.10 use the same fiber density (ft = 0) and clique threshold (ct = 9). From the left graph layout in Figure 5.9, ROI 38 is disconnected from others. As for the right side of the clique visualization, the clique numbers in Figure 5.10 are more than that in Figure 5.9. Besides, some nodes are included in more than one clique in Figure 5.10. For instance, ROI 20 belong one clique in Figure 5.9. By contrast, ROI 20 is included in five cliques in Figure 5.10. It has the same situation for ROI 15, ROI 40, and so on. In other words, the information transfer for ROI 20 is getting weaker with the influence of disease. In conclusion, the connectivity of Figure 5.9 looks like reduced a lot from the perspective of the clique.
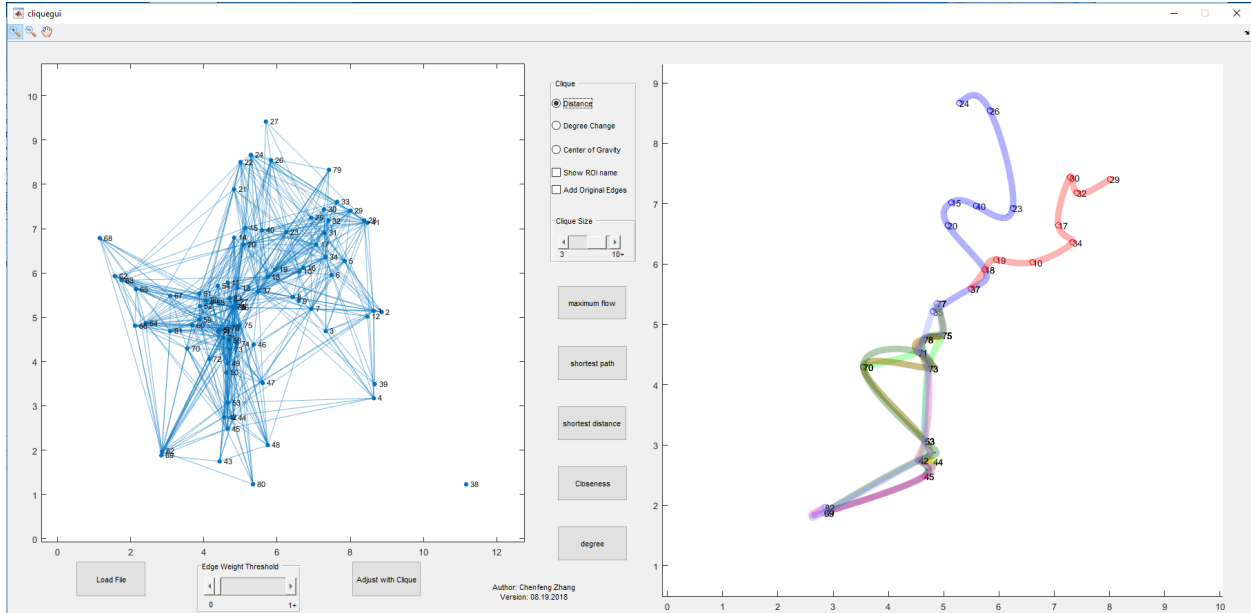
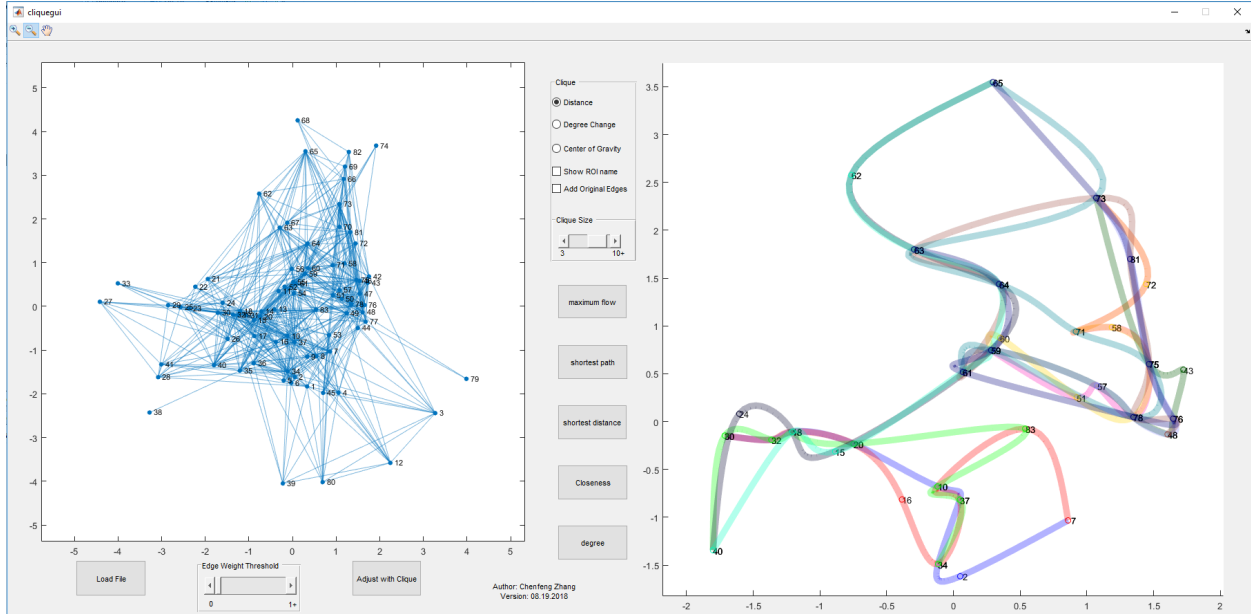Figure 5.9 ADNI dataset for the subject with Alzheimer's disease

Figure 5.10 ADNI dataset for the normal subject

5.4 Other functions in clique visualization

In clique visualization, a checkbox is added for display the original edges for ROIs (Figure 5.11). It could help the user to find the ROI with most connections with others. For example, ROI 26 has the most connection with other ROIs. It looks like that the ROI 26 is an important information transfer node.
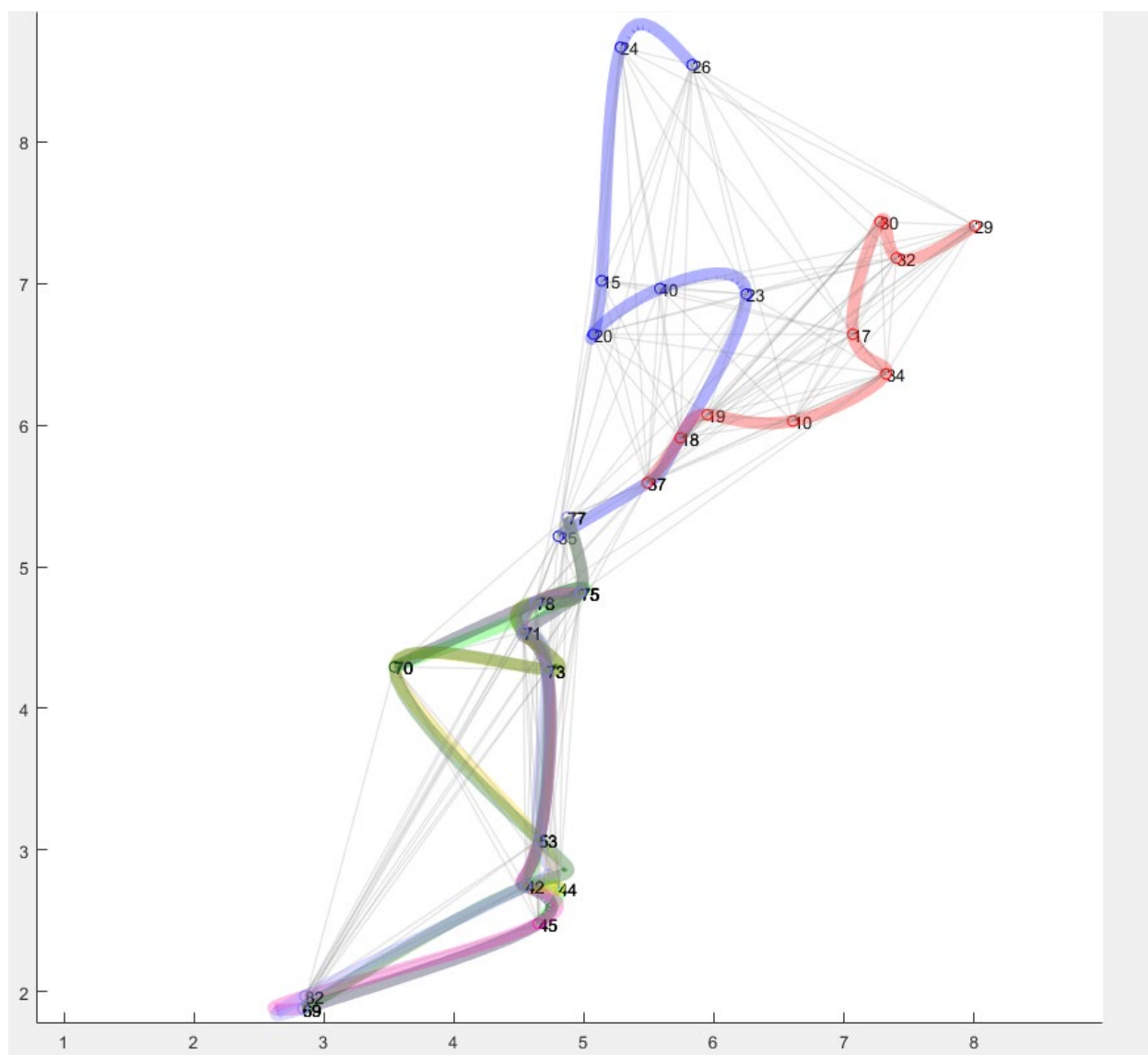

Figure 5.11 ADNI dataset for the subject with Alzheimer's disease (with original edges)

Besides, the 2D visualization tool also provides another checkbox to change the name of ROIs. Since every node has an ROI name in the anatomical parcellation. The function could switch the node labels (Figure 5.12).
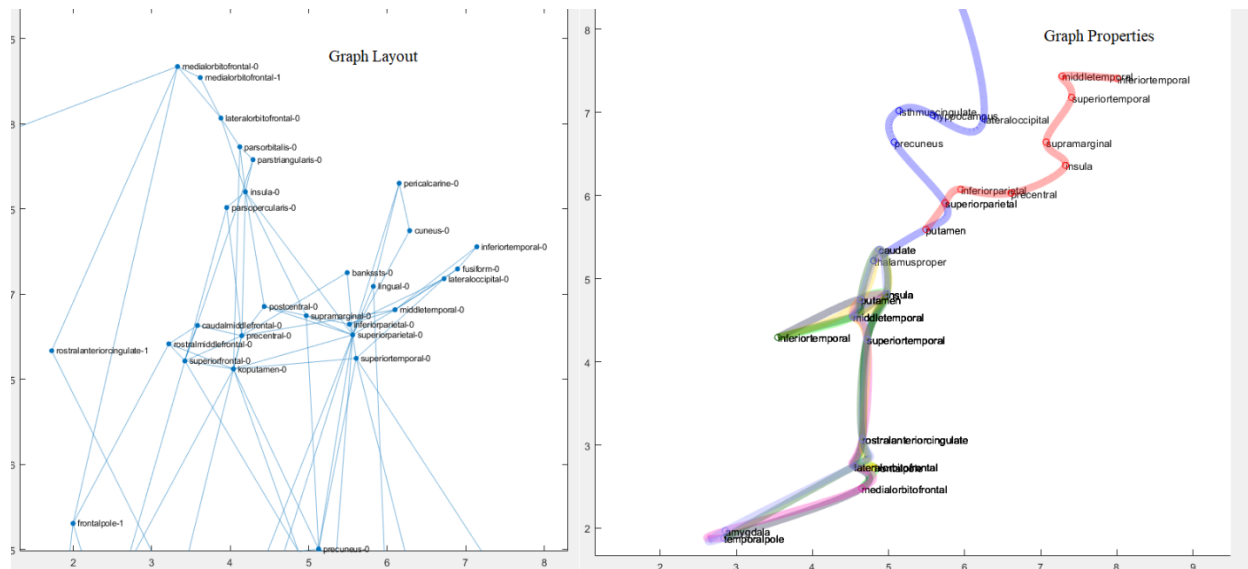

Figure 5.12 ADNI dataset for the subject with Alzheimer's disease (with ROI names)

CHAPTER 6 CONCLUSION

In the thesis, I discussed both 3D visualization and 2D visualization for brain network connectome. To help the user study and compare the properties of brain network and ADNI dataset, graph properties visualization is introduced.

For 3D visualization, the ROIs locations are fixed and visualized in the 3D brain model. It could help the user find the ROIs easily and quickly. With the degree and closeness visualization, the user could compare the normal and abnormal ROIs. However, as for other graph properties like the maximum flow and the shortest path, the visualization results cause vision clutter, which is hard to find any pattern or information.

To solve the problems in 3D, I also use 2D visualization too. The 2D visualization includes graph visualization and graph properties visualization. In 2D, the ROI locations are not fixed. By contrast, they are arranged by the force-directed layout. In other words, their locations are decided by the power (edge weight) among them. Although the method could not visualize the ROIs as intuitive as that in 3D, it could reveal the properties of the dataset itself. Except for graph visualization in 2D, the graph properties are also visualized in the 2D tool. The degree and closeness visualization methods are the same as 3D. The shortest path and maximum flow problems in 3D are solved by adopting the color map method. By following the color bar, the 2D visualization could easily find the selected value. Besides, as for the graph property-clique, I applied its concept to graph visualization to help reduce the clutter. And the force-directed layout visualization result improved a lot with the restraint of cliques. In addition, the clique visualization has three algorithms to connect the ROIs with a smoother curve and less cross line. They are distance method, degree change method, and center of gravity method. The first one is an approximation method which might not accurate enough. The last two are the exact algorithm. Since the problem is an NP-hard

problem, it means that the running time could be very long. So, by comparing the three visualization results when ROI is 83, the distance algorithm runs faster than the others. When the ROI value is increased, the size of the clique would also be increased. In that case, the implementation time becomes very important for the 2D interactive tool. Therefore, the distance algorithm is a better choice than others for the perspective of time response.

Besides, the visualization results shown in the thesis are from two subjects' dataset. Although the examples analyzed in the thesis could not include all ADNI dataset visualization situations due to the individual differences, the impact of Alzheimer's disease is the same. The 2D and 3D visualization tools are still very meaningful for ADNI dataset study. Firstly, both tools provide the file loading function for selected dataset visualization. Secondly, the visualization tools are designed for showing the influence from edge weight change.

In conclusion, comparing the 3D and 2D visualization result, 2D is not as intuitive as 3D. The ROIs' relative locations could be anywhere in 2D. But the 2D could reflect the substance of dataset. In other words, the 2D visualization tool could help to study the dataset of brain connectome, comparing and detecting the pattern of normal and abnormal brain connectivity easily. Therefore, a tool combining the advantages of 3D and 2D together might provide better visualization. Besides, to receive a general graph properties visualization comparison, the average data for the normal and abnormal subject might be visualized for further research.

REFERENCES

[1] R. Pant (2015, January 16). Visual marketing: a picture's worth 60,000 words. Business 2Community, Retrieved from https://www.business2community.com/digital-marketing/visual-marketing-pictures-worth-60000-words-01126256 (accessed November 29,2018).

[2] E. Sinar (2016, January 1). Use animation to supercharge data visualization. Retrieved from https://medium.com/@EvanSinar/use-animation-to-supercharge-data-visualization-cd905a882ad4 (accessed November 29,2018).

[3] D. Holten, Jarke J.vanWijk.: Force-directed edge bundling for graph visualization. Proceeding EuroVIs'09 Proceedings of the 11th Eurographics/IEEE-VGTC conference on Visualization,2009:983-998.

[4] E.T. Bullmore, D.S. Bassett.: Brain graphs: graphical models of the human brain connectome. Annual review of clinical psychology, 2011,7:113-40.

[5] H. Li, S. Fang, J. Goni.: Integrated Visualization of Human Brain Connectome Data. International Conference on Brain Informatics and Health,2015:295-305.

[6] T. Kamada, S. Kawai.: An algorithm for drawing general undirected graphs. Information Processing Letters 31, 1989:7-15.

[7] M. Held, R.M. Karp.: A dynamic programming approach to sequencing problems. Journal of the society for industrial and applied mathematics, 1961,10(1):196-210.

[8] Y. Li.: A new exact algorithm for traveling salesman problem with time complexity interval($O(n^4)$,) $O(n^3*2^n)$). 2014 Retrieved from https://pdfs.semanticscholar.org/444f/150affc6b0de006f2b7003dfd7d31c473f9f.pdf (accessed November 29,2018).

[9] O. Sporns, G. Tononi, R. Kotter.: The human connectome: a structural description of the human brain. PLoS Computational Biology, 2005, 1(4): e42.

[10] E. Bullmore, O. Sporns.: Complex brain networks: graph theoretical analysis of structural and functional systems. Nature Reviews Neuroscience, 2009, 10:186-198.

[11] Y. He, A. Evans.: Graph theoretical modeling of brain connectivity. Current opinion in neurology, 2010,23(4):341-350.

[12] M. Xia, J. Wang, Y. He.: BrainNet Viewer: a network visualization tool for human brain connectomics. PLoS ONE, 2013, 8(7): e68910.

[13] R.A. Laplante, L.Douw, W.Tang, S.M. Stufflebeam.: The connectome visualization utility: software for visualization of human brain networks. PloS ONE, 2014, 9(12): e113838.

[14] A.C. Telea.: Data Visualization: Principles and Practices, Second Edition. A. K. Peters/CRC Press, 2007. Print.

[15] D.S. Margulies, J. Bottger, A. Watanabe.: Visualizing the human connectome. NeuroImage. 2013, 80: 445-461.

[16] M.D. Greicius, G. Srivastava.: Default-mode network activity distinguishes Alzheimer's disease from healthy aging: evidence from functional MRI. Proceedings of the National Academy of Sciences of the United States of America. 2004, 101(13):4637-4642.

[17] R. Gove.: 6 ways to visualize graphs. 09.06.2017. Retrieved from https://www.twosixlabs.com/6-ways-visualize-graphs/ (accessed November 29,2018).

[18] T.M.J. Fruchterman, E.M. Reingold.: Graph drawing by force-directed placement. Software: Practice and Experience. 1991, 21(11):1129-1164.

[19] M.A.D. Storey, H.A. Muller.: Graph layout adjustment strategies. International Symposium on Graph Drawing. 1995, 1027:487-499.

[20] A. Bretto.: Hypergraph Theory: An introduction (Mathematical Engineering). Springer, 2013. Print.

[21] J. Paquette, T. Tokuyasu.: Hypergraph visualization and enrichment statistics: how the EGAN paradigm facilitates organic discovery from Big Data. Human Vision and Electronic ImagineXVI. 2011, 7865

[22] R.D. Alba.: A graph-theoretic definition of a sociometric clique. Journal of Mathematical Sociology.1973, 3(1):113-126.

[23] D. Johnson.: Asymptotic experimental analysis for the Held-Karp traveling salesman bound. Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms. 1996. Chapter 39.

[24] C. Nilsson.: Heuristics for the traveling salesman problem. 2003. Retrieved from: https://pdfs.semanticscholar.org/7b80/bfc1c5dd4e10ec807c6f56d0f31f8bf86bc6.pdf (accessed November 29,2018).

[25] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan.: The traveling salesman Problems: a guided tour of combinatorial optimization. Wiley. 1985. Print.

[26] S. Hamad.: The Cross Entropy method and its applications. Master of Science in Mathematics. Eastern Mediterranean University.2015

[27] P.T. De Boer, D.P. Kroese, S. Mannor.: A tutorial on the cross-entropy. Annals of Operations Research.2005, 134(1):19-67.

[28] X. Li, J. Xue.: Complex Quadratic Bezier Curve on Unit Circle. International Conference on Logistics Engineering, Management, and Computer Science. 2014, 1138-1141.

[29] R.J. Wilson.: Introduction to Graph Theory, 5 Edition. Pearson, 2012. Print.

[30] B. R. Preiss.: Data structures and algorithms with object-oriented design patterns in C#. 2002. Retrieved from:

https://github.com/arasty/books/blob/master/Data%20Structures%20and%20Algorithms%20with%20Object-Oriented%20Design%20Patterns%20in%20C%23%20-%20Bruno%20Preiss.pdf (accessed November 29,2018).

[31] L. Lovasz.: Large networks and graph limits. American Mathematical Society.2012. Print.

[32] J. Schroeder, A.L.P. Guedes.: Computing the minimum cut and maximum flow of undirected graphs. 2004. Retrieved from

https://pdfs.semanticscholar.org/56e2/fe2b1e9da4fe6edd163606215228addaf8b5.pdf (accessed November 29,2018).

[33] A. B. Tucker.: Computer science handbook, Second Edition. Chapman and Hall/CRC. 2004. Print.