Affinity: Web Based Rapid Exploration Tool for Multispectral Neural Connectivity

David Caldwell[*]
Dept. of
Bioengineering
CSNE
University of
Washington
Seattle, USA
djcald@uw.edu

Jing (James) Wu[*]
Dept. of
Bioengineering
CSNE
University of
Washington
Seattle, USA
jiwu@uw.edu

Nick Foti
I-LABS,
University of
Washington
Seattle, USA
nfoti@uw.edu

Emily Fox
Dept. of Statistics
University of
Washington
Seattle, USA
ebfox@uw.edu

ABSTRACT

Neural connectivity data often exists in a high dimensional space, with multivariate attributes for each edge between different brain regions. Visualizing a connectivity matrix, looking for trends and patterns of interest, and subsequently dynamically manipulating and drilling down these values is a challenge for scientists dealing with data from various modalities such as magnetoencephalography and electrocorticography. There exists a need for a lightweight, open-source, easy to use visualization tool to allow for the rapid exploration of these connectivity matrices. Here we present a client-side visualization tool written entirely in HTML/CSS/JS that allows for the in-browser manipulation of user defined files for the exploration of brain connectivity. End-user feedback from domain experts suggests that this will be a valuable tool for revealing underlying trends in connectivity data, and presenting these results to others in the field.

Author Keywords
Neural Connectivity; Interactive Visualization

ACM Classification Keywords
H.5.m.

INTRODUCTION
In this paper we present an interactive visualization exploration tool specifically tailored to large numbers of pairwise neural connectivity dynamics. We have used this tool to analyze connections from various regions of the brain to one another based off of magnetoencephalography (MEG) and electrocorticography (ECoG) data.

This is a challenging problem because cortical connectivity visualizations are largely exploratory and requires the encoding of a large number of multivariate attributes that exist for each pairwise edge, as each edge can be its own high-dimensional correlation matrix. These multiple metrics for each connection require an investment of user time to navigate and interpret. Unlike social networks or aggregate flow for which directed graphs are often designed, neural connectivity are highly nonstationary through time, and the available toolbox of descriptive statistics are in a large state of flux. This adds a large number of potential encoding dimensions for a visualization, and a critical need for user interactivity to represent these additional dimensions to lead to useful knowledge.

Our end users, Dr. Emily Fox and her postdoctoral fellow Nick Foti, expressed interest in visualizing an inverse correlation matrix of pairwise recordings sorted by spectral density, representing connection strengths between different regions at different frequencies in magnetoencephalography (MEG) data, and the ability to display more summary statistics based upon selected frequencies and connection regions. Additionally, our own lab with Dr. Ojemann is also very interested in applying the same techniques to pairwise phase-amplitude and phase-phase locking relationships in our own electrocorticography (ECoG) datasets. This represents a unique opportunity to apply and extend existing interactive visualization techniques - such as chord diagrams, dynamic querying and selection, and interactivity, to exploratory neuroscience.

RELATED WORK
Visualizing neural connectivity is a problem faced by the neuroscience field, for which many innovative solutions have been explored and implemented. However, many have drawbacks which preclude their utility for our intended users of neuroscientists in collaborating laboratories at the University of Washington. Here we will outline some of the

existing tools, and reveal the need for a novel connectivity visualization tool.

Many examples can be found in a centralized repository named "The Neuroimaging Informatics Tools and Resources Clearinghouse" (NITRC) [1], founded by the National Institutes of Health (NIH), which hosts hundreds of tools (mostly for MRI data) allowing for the analysis and visualization of Neuroimaging data. A subset of tools for MEG and ECoG data function as MATLAB plugins, which subsequently require package download, a MATLAB license, and user interface with a powerful but complicated software suite. We see from this a need for an open source, browser based visualization toolkit that allows more rapid exploration and visualization.

Other examples from the literature will be explored below.

Anatomically Localized Graphs

Salvador et al, 2005, demonstrated static graphs of frequency dependent functional connectivity through undirected graphs based off of fMRI data. An example image is seen below (Figure . Of note in the image below is the labeling of various anatomic locations in free space, with a web of connections in red and green between the two of them. Green edges represent bilaterally symmetric pairs (corresponding left and right regions of the same anatomical structure). Red represent asymmetric connections. While this visualization demonstrates connectivity between numerous nodes and captures the connectivity of the system, we found it difficult to interpret quickly and sought a more intuitive way to represent visualization.
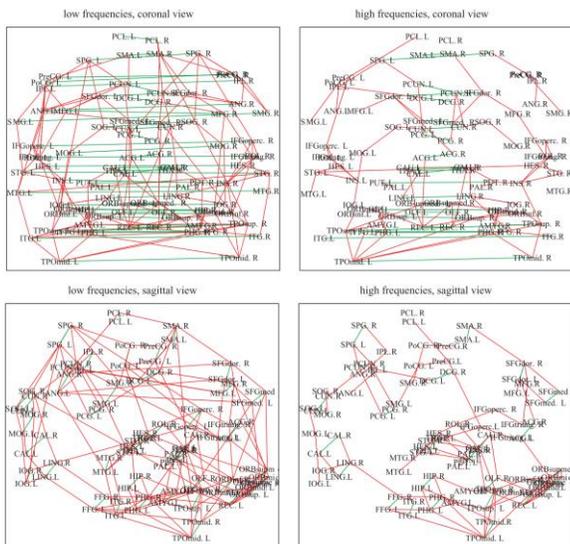
Figure 1. Neural Connectivity from Salvador et al, 2005. Nodes are brain regions, and edges between them are markers of connection strength [2].

Alper et al. (2013) [3] demonstrate different methods used in the past for demonstrating both functional connectivity (correlations in activity between various regions) and anatomic connectivity (fiber tract density between regions).

These visualizations range from clearly demonstrating brain structure, to more abstract demonstrating the brain with nodes and edges, and all the way to combining nodes and edges and anatomical landmarks. Matrix visualizations are also commonly used for visualizing connectivity between regions.

While this type of visualization preserves anatomical locations, we feel that for our visualization it may compromise the clarity and simplicity of quickly being able to identify major connectivity regions, as relations cannot be quickly discerned without interactive manual rotation.

METHODS

Our visualization technical architecture is outlined in the schematic shown in Figure 2. Briefly, a correlation matrix (3D matrix of connectivity strengths at various sensor locations and frequency regimes) are loaded into a client-side Javascript-enabled page in any modern desktop or mobile browser via drag and drop or a browsing menu. The correlation matrices are nested arrays in Javascript Object Notation (JSON) format is loaded using HTML5 standards-compatible file handing routines. These datasets can be generated by conversion from either a python numpy data file or a MATLAB data file, using an included python script by the end-user. These two data formats represent the vast majority of the userbase in the neuroscience discipline in storing pairwise connectivity data.

We chose a visualization technique particularly suited for representing pairwise connections with minimal occlusion, which is the d3.js [4] implementation of chord diagrams based upon Krzywinski's *circos* [5]. In order for the implementation to be usable the interface must be rapid and responsive when selecting and visualizing more information about pairwise connections.
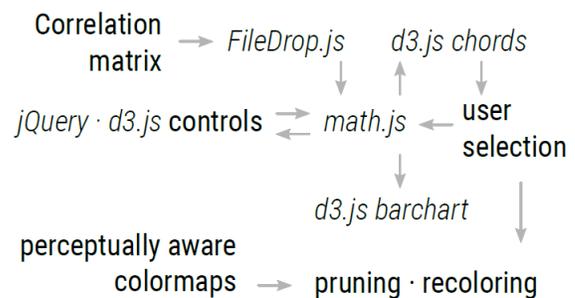
Figure 2. Schematic overview of approach to the visualization

The user is able to select a frequency range of interest, a statistical measure of interest (absolute value or magnitude and phase information if present), the inclusion or exclusion of self-connections, and then presses the "re-slice matrix and render" button to submit the query and update the chord diagram. This uses math.js in order to rapidly subselect and compute an average matrix slice across the selected

frequency regimes. The responsive controls use Javascript d3.js and jQuery controller frameworks in order to create cross-platform, responsive controls. In particular, the d3.js implementation of d3.svg.arc d3.layout.chords are heavily called upon in order to rapidly reshape the view. The user-selected threshold cut-off value in real time dynamically prunes connections below a certain strength from the diagram. A callback with active tweening of the arc and chord shapes with d3 transitions is used in order convey a sense of smooth but rapid transition in order not to lose the sense of overall context as a large number of chords fade from view with increasing minimum threshold, as shown in Figure 3.
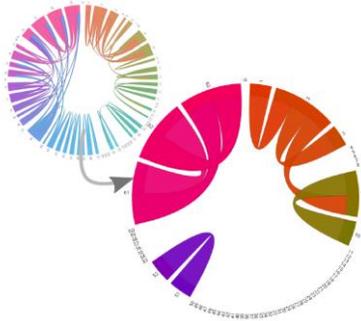
Figure 3. Example screenshot illustrating rapid fade and tween pruning using a single slider in order to select for maximally salient chords and remove occlusion. Note that regions with similar colors intuitively conveys a local group of intra-region connections while dissimilar colors convey cross-cortical connections.

The user then has the option to color the chord diagram based off of anatomic locations from a file (if present), or in a constant luminance scheme based off sequential order. As the color representation of nominal categories is extremely important [6] and spatial information must be accurately conveyed, in this case, we tailored a CIE L*a*b colorscale that fades in two directions, as well as a L*c*h colorscale that transitions in color unidirectionally, as shown in Figure 4.
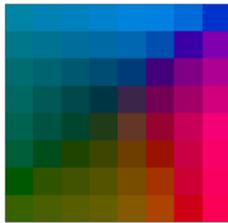
Figure 4. A modified L*a*b colorspace controlled for luminance for coloring arcs representing spatial locations around the chord diagram in order to give the user rapid assessment of sensor proximity.

Once the chord diagram has been setup for visualization, the user can hover over a node or a chord to display connection strengths for that node, or between the two nodes specified by a given chord. An additional user option for coloring via mean phase angle of the connection uses a divergent color palette from ColorBrewer (colorbrewer.org).

Upon clicking a chord, a bar graph will dynamically update which displays connection strengths across all frequencies for the two nodes of interest. By hovering over any bar in the chart, the connectivity strength will be displayed.
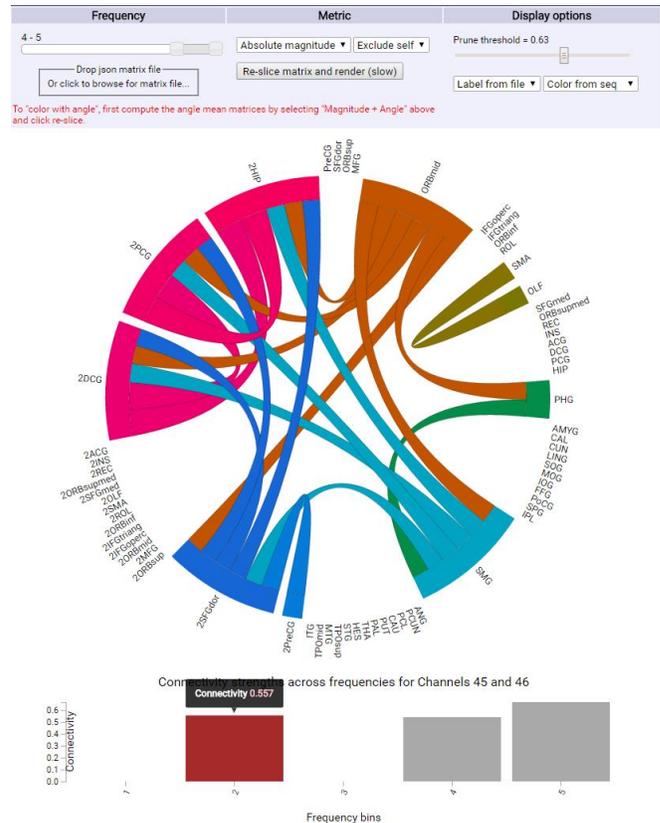
Figure 5. Example screenshot from the visualization highlighting the use of subselection, thresholding, dynamic bar graph generation, and hover feature.

RESULTS

The screenshot in Figure 5 provides an illustration of the use case of our visualization tool, as well as insights rapidly revealed by exploratory data analysis. First, a frequency range subselecting high frequencies was selected, with self-connections excluded (as these often do not add additional valuable information, depending on the data type). A threshold was set to prune out weaker connections, allowing for decluttering of the visualization space and emphasis of the most strongly represented components.

Example anatomic labeling was provided for this dataset, allowing for the user to assess connectivity between various regions by a convention familiar to neuroscientists. The user then clicked on a chord connecting two regions, which

subsequently displayed the connectivity values across all frequencies for those two locations.

This revealed the interesting insight that there was a small range of intermediate frequencies which had significant connections, which would not have been apparent from first glance at either a connectivity matrix or the chord diagram alone. Subsequent hover functionality revealed the exact strength of connectivity, allowing for the user to obtain a quantitative metric of the desired connection between two regions for a given frequency.

Our results for test matrices of size 65x64x64 and 5x64x64 reveal rerendering and math.js processing within 500 ms to a second depending on the user's' machine, which is currently acceptable performance for our users. Code profiling revealed the bottlenecks to be in the math.js processing and computations on matrices, so in the event of matrices orders of magnitude larger than our current ones, we may need to consider alternative approaches as discussed in the future directions.

DISCUSSION
Feedback provided by one of the end users, Dr. Nick Foti, illustrated his satisfaction with tool as a method to explore connectivities between various brain regions.

Furthermore, we extended our visualization tool to datasets within our own research group, which demonstrated the utility of this novel tool for analyzing both MEG and ECoG data, independent of the number of sensor channels or frequencies desired.

One example of data analysis performed with our laboratory data revealed the utility of our tool. We subselected a high frequency range, and thresholded all of the small valued connectivities. What jumped out in the chord diagram were high frequency hubs of cortical connectivity, which were motor regions of the cortex. This is consistent with our prior work and literature that motor regions are highly correlated in the high frequency band. One example chord went from a posterior parietal region (representative of premotor or motor planning functionality) to the motor cortex, illustrating our tool's ability to rapidly assess and interpret connectivities. By clicking on these chords and other, we were able to quickly assess if the connections were limited to these higher frequencies, or if they spread across the frequency range.

Another potential other users from the ECoG research sphere, Dr. Kurt Weaver of Radiology, had the opportunity to view the visualization tool and commented that he liked having both the ability to see global connectivity patterns in a given frequency band while also seeing the connectivity between two regions across all frequencies. In addition, Dr. Weaver pointed out that in comparison to existing tools for visualizing connectivity, many of which are MATLAB plugins or packages, our visualization functions as a standalone browser application. This provides a lesser barrier to entry to exploring connectivity.

One of the PhD members of our laboratory, Kaitlyn Casimo, commented that this visualization tool was the most effective way she currently has to rapidly assess connectivity patterns across regions of interest. Her PhD work focuses on connectivity in ECoG data, and subsequent changes in connectivity from task learning.

Since the visualization tool does not require any additional software packages, this is a significant advantage over existing tools which may require MATLAB or other software on the end user's computer.

FUTURE WORK
Possible future work could include the inclusion of 3D rendered anatomic image of the brain, either custom rendered for each subject, or a general one based off of anatomic atlases such as the Talairach/MNI brain. This brain, upon highlighting of the chord diagram, could dynamically update to draw attention to different physical locations on the brain representing the connectivity of two regions. In this way, the abstract labels and ordering of the nodes on the chord diagram could be better visualized by the user by mapping to a more intuitive visual and spatial representation. A possible implementation of this could be performed in WebGL.

An additional dimension of visualization could involve plotting temporal variability. Currently our implementation focuses on aggregate statistics over an entire time window. Oftentimes signals of interest occur at different times during the highly nonstationary neural recordings. This can be of critical importance for various data modalities, ranging from MEG to ECoG.

Discussion with our collaborator at the poster session revealed interest in a polar plot of phase angles between two regions, in order to reveal information along the unplotted dimension in the chord diagram, much like we implemented for the bar graph representing correlation magnitude.

We currently have implemented an equal luminance color scheme to nominally encode sensor locations based off of a grid, but based off of user feedback, this could be modified to better allow for localization.

Another additional component could include a dynamically generated summary table, able to be exported as a text file, which shows the strengths of connectivity between various regions, channels of interest or hubs of connectivity, and other metrics that could be useful for reporting in a scientific publication.

Additional processing functionality could be added in through the use of web based python script, implemented through flask, which is an in-browser python framework. This functionality could be important for matrices containing millions of elements, which are much larger than the sample datasets we have been exploring. The use of flask would require hosting on a web service different than github pages, which only supports static webpages. For truly large datasets,

we can envision the use of an implementation based on the imMens [8] architecture for querying data.

REFERENCES

1. NITRC: Welcome. 2016. Retrieved June 2016 from https://www.nitrc.org/

2. Salvador, Raymond, et al. "Undirected graphs of frequency-dependent functional connectivity in whole brain networks." *Philosophical Transactions of the Royal Society of London B: Biological Sciences* 360.1457 (2005): 937-946.

3. Alper, Basak, et al. "Weighted graph comparison techniques for brain connectivity analysis." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013.

4. Bostock, Mike. "Data-Driven Documents (d3. js), a visualization framework for internet browsers running JavaScrip t." (2012).

5. Krzywinski, Martin, et al. "Circos: an information aesthetic for comparative genomics." *Genome research* 19.9 (2009): 1639-1645.

6. Heer, Jeffrey, and Michael Bostock. "Crowdsourcing graphical perception: using mechanical turk to assess visualization design." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2010.

7. Liu, Zhicheng, Biye Jiang, and Jeffrey Heer. "imMens: Real-time Visual Querying of Big Data." *Computer Graphics Forum*. Vol. 32. No. 3pt4. Blackwell Publishing Ltd, 2013.

8. Liu Z, Jiang B, Heer J. ImMens: Real-time visual querying of big data. *Comput Graph Forum*. 2013;32(3 PART4):421-430. doi:10.1111/cgf.12129.