

Immersive Exploration of Brain Simulation Data

Aarhus University IEEE SciVis Contest Team 2023*

1 INTRODUCTION

The contest dataset is in the domain of neuronal network simulation, even linking it back to clinical measurements. Therefore, we built our contest solution in close collaboration with two domain experts: a computational neuroscientist and a medical expert working in neuroimaging and deep brain stimulation. Joint initial data analyses quickly revealed a gap between our visualizations and the mental map of our domain experts: Notions like left and right hemispheres, or like frontal and occipital lobes are very engrained in the domain and form important positional cues for the domain experts. Yet our figures using mainly a PCA-based embedding of the 3D neuron positions did not preserve the inherent 3-dimensionality of the brain. This turned out to be a major hurdle for interpreting and comparing the data, as well as for refinding and communicating insights.

As a result of this initial experience, we developed an immersive analytics framework in which the domain experts can explore the simulation data in virtual reality (VR) without having to compromise their intuition of the brain's anatomy. The following sections give an overview of our framework (Sec.2), an account of the underlying methods (Sec.3) and its realization (Sec.4), and provide brief answers to the questions posed on the contest website (Sec.5).

2 FRAMEWORK OVERVIEW

Our framework builds on a hierarchical aggregation [4] of the neuronal network through recursive clustering of highly connected regions of the brain. The resulting cluster hierarchy forms the foundation of most aspects of our framework from the partitioning of the data in the database to the hierarchical edge bundling of the synapses and the interactive drill-down/roll-up for locally adjusting the level of detail of clusters. The design of our framework was guided by the tasks set out by the contest as well as by our collaborating domain experts, and it was further informed by known design criteria [7].

The framework offers two types of views: a 3D rendering of the neuronal network that shows a simulation step in its spatial frame of reference, and a line chart with a collection of trajectories (one per cluster) showing the full simulation in its temporal frame of reference. Interactions like selection/deselection and drill-down/roll-up of clusters are linked between views (Fig. 2). Together, these views cater towards **Task 1** set out by the contest in that they provide a spatial as well as a temporal overview of the data.

Both views can be parametrized to show any of the numerical attributes that are given or computed for the neurons and aggregated for the clusters. Different aggregation mechanisms are available

(min, avg, max) to customize the shown aggregates to the analysis task at hand. Together with interactive filtering mechanisms, this caters towards **Task 2** as it allows to show plasticity changes as captured by the number of dendrites, axons, and synapses, which are common quantifiers for neural plasticity [10].

Multiple views can be spawned to show different time steps, different simulations, different levels of detail, different numerical attributes, or different 3D perspectives side by side. Handling of multiple views is eased by interactive features like a “format painter” for copying the state of one view (perspective, selection, levels of detail) to another view to facilitate quick visual comparison without any manual adjustment (Fig. 3). These capabilities cater towards **Task 3** – the visual-interactive analysis of simulation ensembles.

All these features are integrated in a flexible immersive analytics environment that allows their use in any order or combination. This fulfills **Task 4** of the contest, as any desired analysis workflow making use of the above features can thus be carried out. Switching between different levels of complexity is accommodated via the interactive drill-down/roll-up of the cluster structure (Fig. 1).

We furthermore extended the list of predefined tasks with an **additional task**: the presentation of the analysis results. That is because we noticed that the communication of insights gained in VR – even if shown live – often leaves uncertainties or questions among the audience about what can be seen: perspective distortion, occlusion, and the chosen aggregation/complexity level often leave the onlookers wondering whether there is more to the story? To alleviate these uncertainties, we have created a physical 3D model of the brain as a tangible and trackable input device to our framework. Not only is the 3D view synced with the orientation of the physical model, but the model can be tapped with a stylus-like pointer and the cluster corresponding to the tapped region will then unfold into its subclusters in the 3D view. By passing the physical model to the audience, the presenter can give them agency over the rotation and level of detail of the 3D view, so that they can convince themselves about the presented analysis results (Fig. 4).

3 USED METHODS

Data Transformation: The clustering of the neuronal network is computed using the *Louvain community detection* algorithm [1]. Specifically, we run the Louvain algorithm recursively to create a hierarchy of “flat” clusterings to impose constraints on the number and size of the produced clusters at each level. This way, we can ensure a similar branching factor at each level, so that the level of detail increases in a way that is useful (not too small), manageable (not too large), and foreseeable (approximately constant).

To manage the time series data provided for all neurons, we compute their *Perceptually Important Points* (PIPs). To speed-up their calculation, we use an adaptation of the *Splitting algorithm* [5] with priority queues instead of linked lists. The outcome is a reordering of time points from chronological to perceptual order. Querying for the first n data points will then return the top- n PIPs of a time series.

Visual Representation: The 3D rendering of the neuronal network uses the neuron positions provided by the contest dataset. Clusters are shown using translucent convex hulls that envelop the region of contained neurons. A click on a cluster expands it in the spirit of an *exploded view* [8] into its next level of subclusters by translating the clusters further outwards with each level of detail. The synapses are bundled using *Mean Shift Edge Bundling* [2], which we have extended to natively take edge weights into account.

*Corneliu Boboc, e-mail: 202102078@post.au.dk
Joachim Brendborg, e-mail: 201807077@post.au.dk
Marius Hografer, e-mail: mhografer@cs.au.dk
Andreas Asferg Jacobsen, e-mail: 201905139@post.au.dk
Jacob Møller Kjeldsen, email: 201808379@post.au.dk
Davide Mottin, e-mail: davide@cs.au.dk
Ken Pfeuffer, e-mail: ken@cs.au.dk
Asger Ullersted Rasmussen, e-mail: 201806934@post.au.dk
Troels Rasmussen, e-mail: troels.rasmussen@au.dk
Juan Manuel Rodriguez, e-mail: jnmrodr@cs.au.dk
Mathias Ryborg, e-mail: 201809050@post.au.dk
Zeinab Schäfer, e-mail: zeinab.schaefer@cs.au.dk
Hans-Jörg Schulz, e-mail: hjschulz@cs.au.dk
Vidur Singh, e-mail: 202101676@post.au.dk
Theodor Tollersrud, e-mail: theo@post.au.dk

To place multiple simulation trajectories in the line chart, we needed to order the trajectories of clusters/neurons to maintain their spatial proximity as observed in the 3D neuronal network view as best as possible. This was achieved by constructing a *space-filling curve* [12] across all neurons and clusters. Yet, a 3D space-filling curve does not work well in this case, as all neurons lie on the surface of the brain, leaving it hollow, which creates rather unreasonable curves/linear orders. Making use of the likeness between neurons on the brain's surface and points on a globe, we first applied a map projection to the 3D data and then used a 2D space-filling curve on the resulting "map of the brain". After testing various projections and space-filling curves, we chose a combination of the *v.d.Grinten projection* and a *Hilbert curve* that gave us the best overall fit.

User Interaction: The VR user interface supports both the fundamental interaction metaphors of virtual hand and virtual pointer via controllers for intuitive rotate, scale, and drag operations on the 3D models and for scene navigation [11]. The newly developed "format painter" tool allows for transferring interactive adjustments made on one visualization to another. At any time, the user can access a virtual handheld menu for further options (setting encoding parameters, adjusting filters, selecting time steps, play/pause of the simulation). Lastly, a real-world passive interface prop based on an actual 3D brain scan provided by one of our domain experts enables to specify spatial relationships in a natural user dialog [6].

4 IMPLEMENTATION

All source code except for the brain scan used for the 3D-printed tangible interface prop is made openly available under a permissive Apache 2.0 license at <https://vis-au.github.io/scivis23/>. The visualization framework runs on a Dell XPS 17 9720 laptop with an Intel i9-12900HK (2.5 GHz, 14 cores, 20 threads) with 64 GByte RAM, a 2 TByte SSD, and a GeForce RTX 3060 graphics card. The visualization backend uses a Mongo DB instance that is coupled with a Redis in-memory cache. Data partitioning in the database follows the hierarchical levels of the data clustering. The backend is exposed to the visualization frontend via a custom server written in Go, communicating using the gRPC protocol. The frontend is realized in Unity, with some visualization functionality using the Immersive Analytics Toolkit [3] and the interaction functionality building on Microsoft's Mixed Reality Toolkit 3. The final output is provided on an HTC Vive Pro 2 head-mounted display with controllers.

5 ANSWERS TO THE CONTEST QUESTIONS

Overall, our domain experts found that in many instances the simulation data did not align with their physiological background knowledge. Appendix B lists some of these instances. In the following, we give brief answers to the five questions posed on the contest website.

(1) Which parts of the brain are more active in firing and restructuring than others? The answer depends very much on the simulation and is illustrated in Appendix C. In simulation 1 (no-network), all regions/clusters were found to be equally active with the connectivity increasing up to the "fully connected brain" for all regions in a similar manner. For simulation 2 (disable), those regions closest to the disabled parts dropped in connectivity, while others further away remained mostly unaffected. For simulation 3 (stimulus), we observe that the clusters subsuming the stimulated regions are most active while the stimulus is applied. Afterwards, activity propagates through the brain, decreasing in intensity. For simulation 4 (calcium), connectivity slowly dropped with slight regional differences to levels as low as the most affected regions in simulation 2 (those closest to the disabled ones).

(2) Can it be determined when the neurons reached a steady state using the calcium concentration and the fluctuation in grown elements? Yes, we observe that calcium levels correlate with dendrite and axon counts, with a steady state being visible as a

plateau in calcium and connectivity curves. (see Appendix D) This effect may be delayed, though, as we can see in the initial growth phase of simulation 1 (no-network), where calcium increases linearly while connectivity follows in an oscillatory or step-wise manner.

(3) Is that the case for all neurons and/or simulations simultaneously? We observe the brain reaching a steady state in terms of calcium concentration and connectivity for all but one simulation, as shown in Appendix D. In simulation 4 (calcium), the brain does not appear to reach a steady state after the new target levels are defined. Instead, calcium levels and connectivity continuously decrease throughout the simulation.

(4) How are still-alive neurons affected by the loss of neighboring neurons based on lesions? Still-alive neurons lose connections to neurons in disabled regions, but they do not pick up in terms of connectivity or activity in response. (see Appendix E) This may be due to the lack of external stimuli after deactivating the brain regions, which would be necessary for the surrounding regions to "learn" the functions previously performed by the disabled parts. In real life, this stimulus is usually provided through rehabilitation methods [9].

(5) Does "learning" improve the connections of groups of neurons? We observe changes in the topology of the brain network before and after the stimuli were applied (see Appendix F). However, as the class labels computed by our clustering differ from the labels provided by the contest data, it is not immediately clear whether these changes directly relate to the stimulated regions.

ACKNOWLEDGMENTS

We are indebted to Mikkel Petersen from Aarhus University's Center of Functionally Integrative Neuroscience for his crucial input during the project. We furthermore thank the Centre for Advanced Visualisation and Interaction at Aarhus University for making their virtual studio available to us for filming our contest video.

REFERENCES

- [1] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.
- [2] J. Böttger, A. Schäfer, G. Lohmann, A. Villringer, and D. S. Margulies. Three-dimensional mean-shift edge bundling for the visualization of functional connectivity in the brain. *IEEE TVCG*, 20(3):471–480, 2014.
- [3] M. Cordeil, A. Cunningham, B. Bach, C. Hurter, B. H. Thomas, K. Marriott, and T. Dwyer. IATK: An immersive analytics toolkit. In *IEEE VR*, pages 200–209, 2019.
- [4] N. Elmqvist and J.-D. Fekete. Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines. *IEEE TVCG*, 16(3):439–454, 2010.
- [5] T.-C. Fu, Y.-K. Hung, and F.-L. Chung. Improvement algorithms of perceptually important point identification for time series data mining. In *IEEE ISCM*, pages 11–15, 2017.
- [6] K. Hinckley, R. Pausch, J. C. Goble, and N. F. Kassell. Passive real-world interface props for neurosurgical visualization. In *Proc. of CHI*, pages 452–458, 1994.
- [7] S. Jaeger, K. Klein, L. Joos, J. Zagermann, M. de Ridder, J. Kim, J. Yang, U. Pfeil, H. Reiterer, and F. Schreiber. Challenges for brain data analysis in VR environments. In *IEEE PacificVis*, pages 42–46, 2019.
- [8] D. Kalkofen, M. Tatzgern, and D. Schmalstieg. Explosion diagrams in augmented reality. In *IEEE VR*, pages 71–78, 2009.
- [9] J. A. Kleim and T. A. Jones. Principles of experience-dependent neural plasticity: Implications for rehabilitation after brain damage. *Journal of Speech, Language, and Hearing Research*, 51(1):S225–S239, 2008.
- [10] M. Pekna and M. Pekny. The neurobiology of brain injury. *Cerebrum*, 2012(4):9:1–10, 2012.
- [11] I. Poupyrev, T. Ichikawa, S. Weghorst, and M. Billinghurst. Egocentric object manipulation in virtual environments: empirical evaluation of interaction techniques. *Computer Graphics Forum*, 17(3):41–52, 1998.
- [12] L. Zhou, C. R. Johnson, and D. Weiskopf. Data-driven space-filling curves. *IEEE TVCG*, 27(2):1591–1600, 2021.

APPENDIX A: SELECTED FUNCTIONALITIES OF OUR VR FRAMEWORK

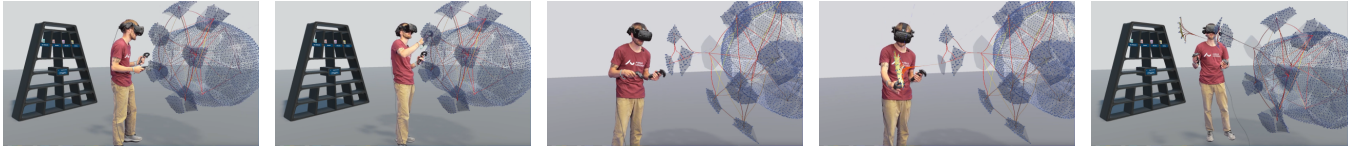


Figure 1: The neuronal network view allows exploring the data on different levels of detail by exploding the cluster hierarchy. In the shown example, the user drills down into the data from the highest aggregation level all the way to individual neurons (left to right).

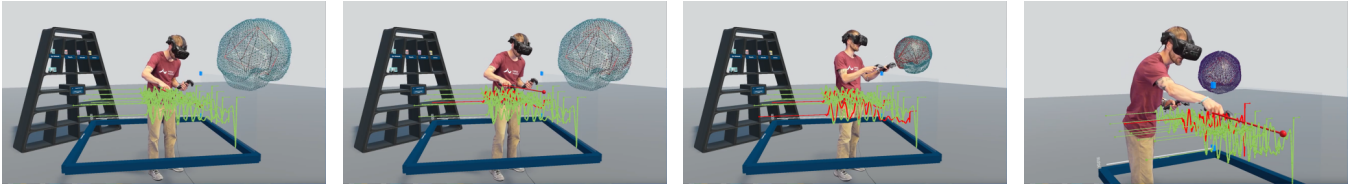


Figure 2: Temporal and network views support brushing and linking, allowing analysts to investigate the spatial and topological relation between temporal patterns of interest and vice-versa. Here, the user selects clusters in the temporal view for simulation 3 (stimulus) that have lower connectivity. The network view reveals that these clusters are in close proximity to each other in the brain.

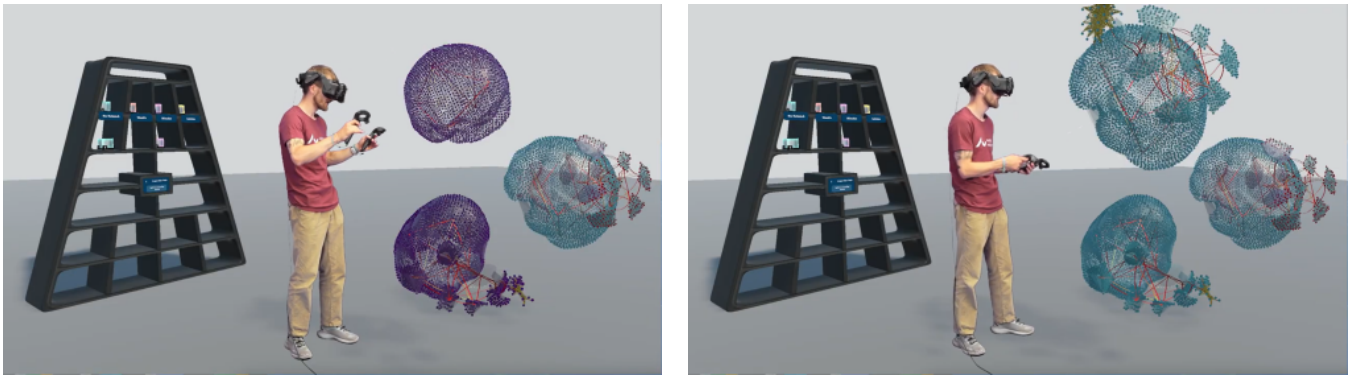


Figure 3: Ensemble visualization of simulation 1 (no-network), simulation 2 (disable), and simulation 3 (stimulus). Using the format painter interaction, the user can apply a preferred configuration of one brain (rotation, cluster explosion, simulation time step) to any other shown simulation, streamlining their comparison.

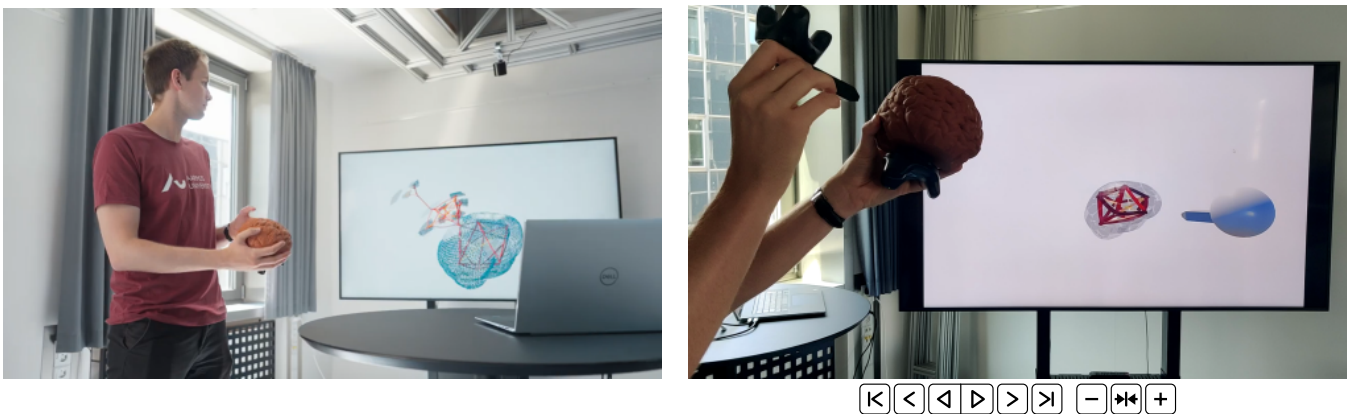


Figure 4: Using a 3D-printed interaction prop of the human brain allows for tangible interaction with the 3D network view to facilitate the communication of insights and their further discussion and exploration outside of VR. This is particularly useful for stakeholders not familiar with the VR controls or the simulation data, as they can relate observations to the physical brain in their hands. Note: To view the animation (right) a stand-alone PDF viewer like Adobe Reader is needed. Alternatively, a video is available on our project website <https://vis-au.github.io/scivis23/>.

APPENDIX B: INSTANCES OF THE SIMULATION NOT ALIGNING WITH BRAIN PHYSIOLOGY.

Note: 2D plots have been added for presentational clarity in this PDF report, which does not lend itself well to communicating data insights from a 3D virtual environment. The 2D plots have been created outside of our software using Python/Matplotlib.

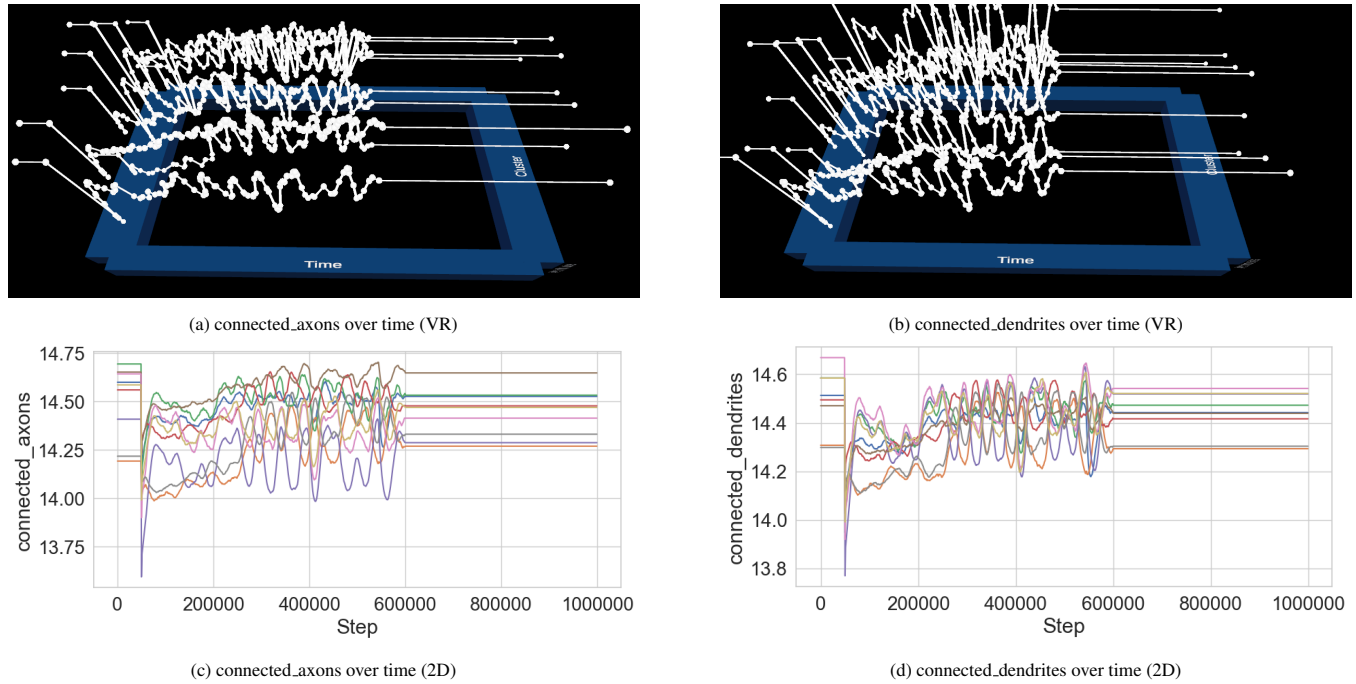


Figure 5: Simulation 3 (stimulus) exhibits an unnatural stability in the number of connected axons and dendrites for the last part of the simulation. This is highly unusual, as even in a resting state without external stimulation, the brain exhibits background activity which should still result in slight connection changes shown as small fluctuations in these curves.

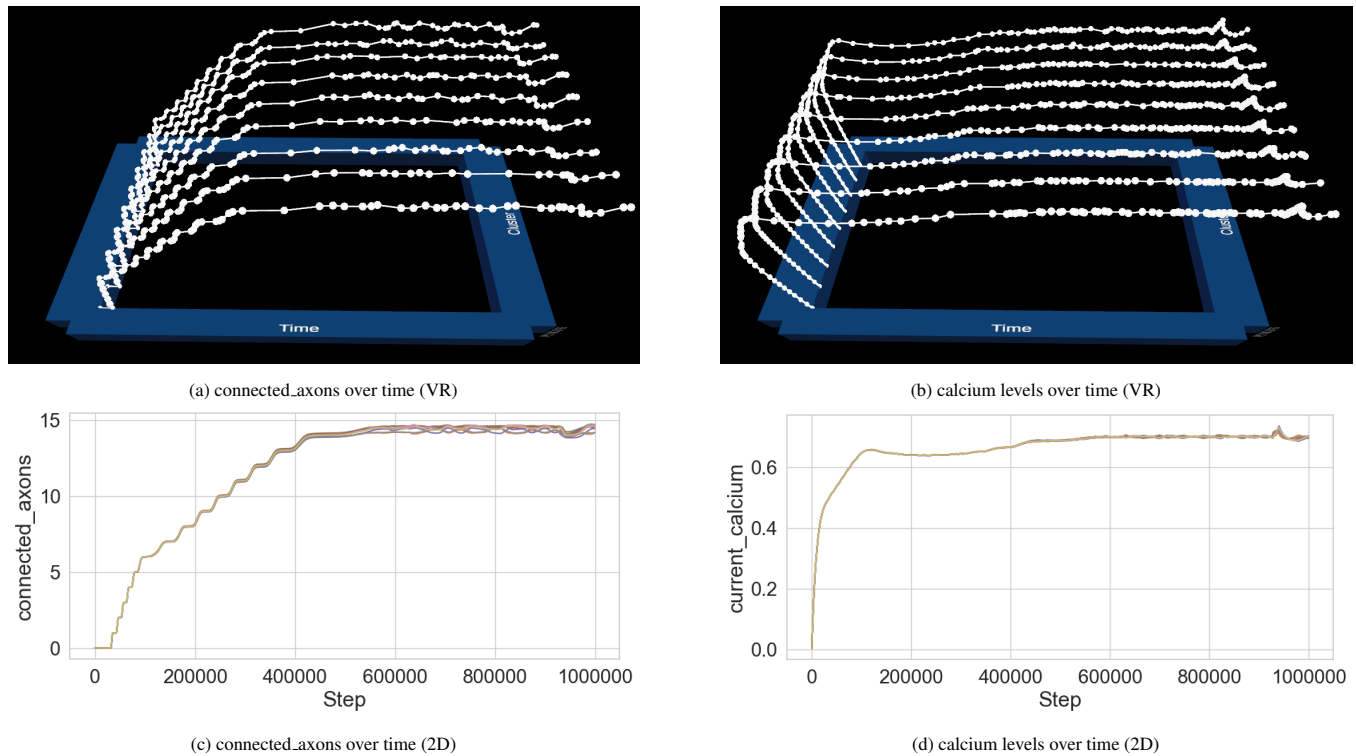


Figure 6: Simulation 1 (no-network) exhibits a “dip” in connectivity towards the end of the simulation run. This dip is preceded by a calcium spike. Both – dip and spike – can be observed for the entire brain, as evidenced by all line charts/clusters showing them. Without external stimuli and encompassing the whole brain at once, these effects are very unlikely to occur in general and in such unison in particular.

APPENDIX C: (1) WHICH PARTS OF THE BRAIN ARE MORE ACTIVE IN FIRING AND RESTRUCTURING THAN OTHERS?

Note: 2D plots have been added for presentational clarity in this PDF report, which does not lend itself well to communicating data insights from a 3D virtual environment. The 2D plots have been created outside of our software using Python/Matplotlib.

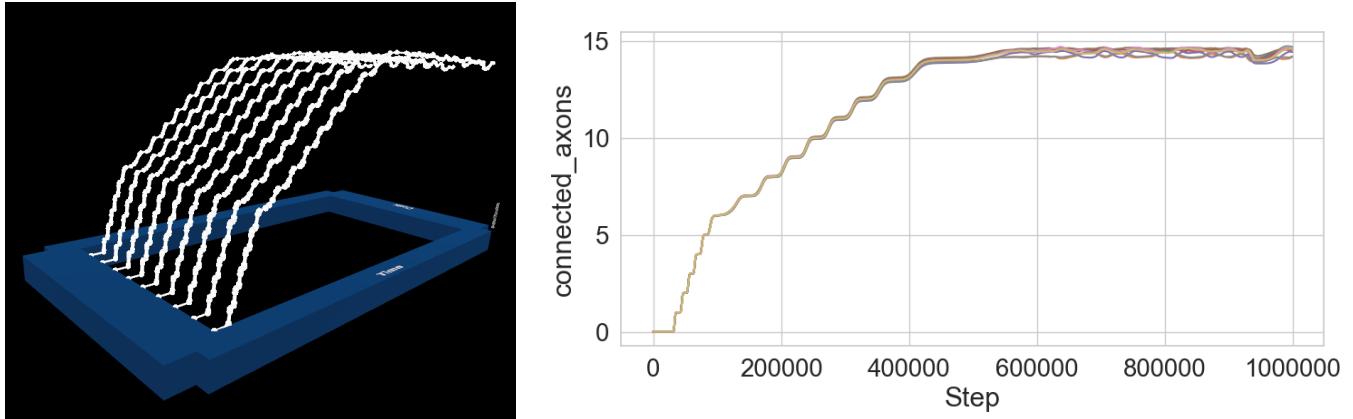


Figure 7: Illustrating restructuring of the neuronal network in simulation 1 (no-network) by plotting the number of connected axons over time. The stepped increase of connections in the beginning of the simulation is clearly visible.

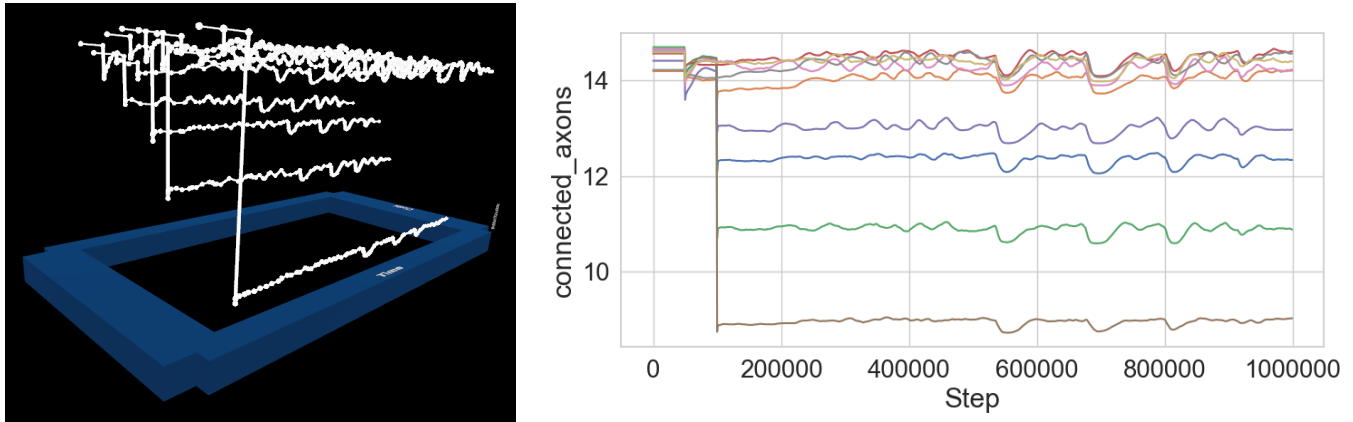
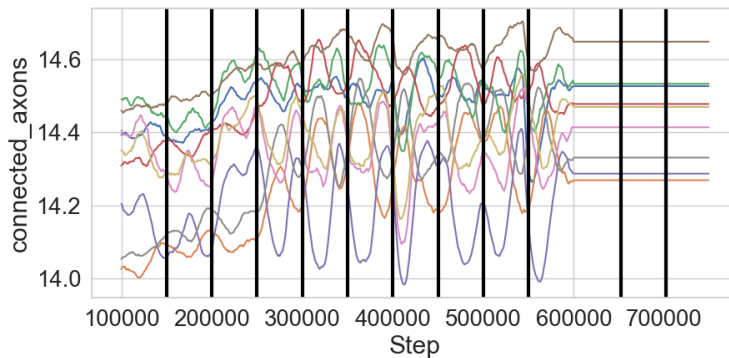
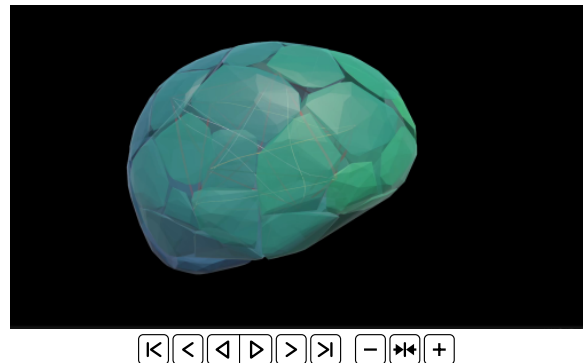


Figure 8: Showing the evolution of connected axons for simulation 2 (disable). The regional differences are clearly visible among the clusters with some being not or only slightly affected, while others have severely reduced connectivity.



(a) Heightened activity being visible in some, most likely the stimulated regions (e.g., red cluster) at the time points of stimulation (black vertical lines).



(b) Animation of the propagation of electric activity shown as cluster color after a stimulus was applied. Color scale from blue (min) to yellow (max).

Figure 9: Showing the brain activity in simulation 3 (stimulus). Color scale: Viridis from blue (min) to yellow (max). To view the animation (right) a stand-alone PDF viewer like Adobe Reader is needed.

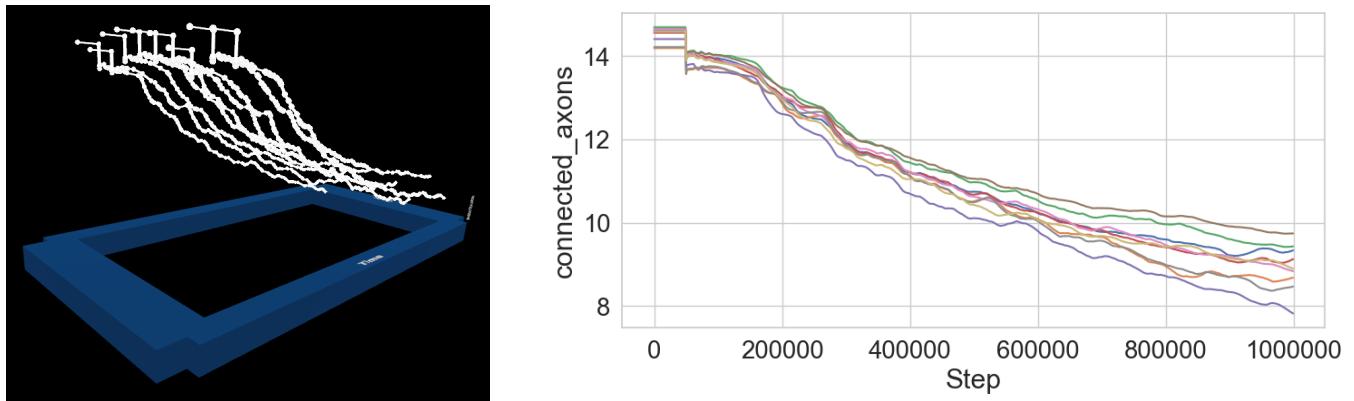


Figure 10: Showing the slow decrease of connected axons for simulation 4 (calcium) over the course of the simulation run.

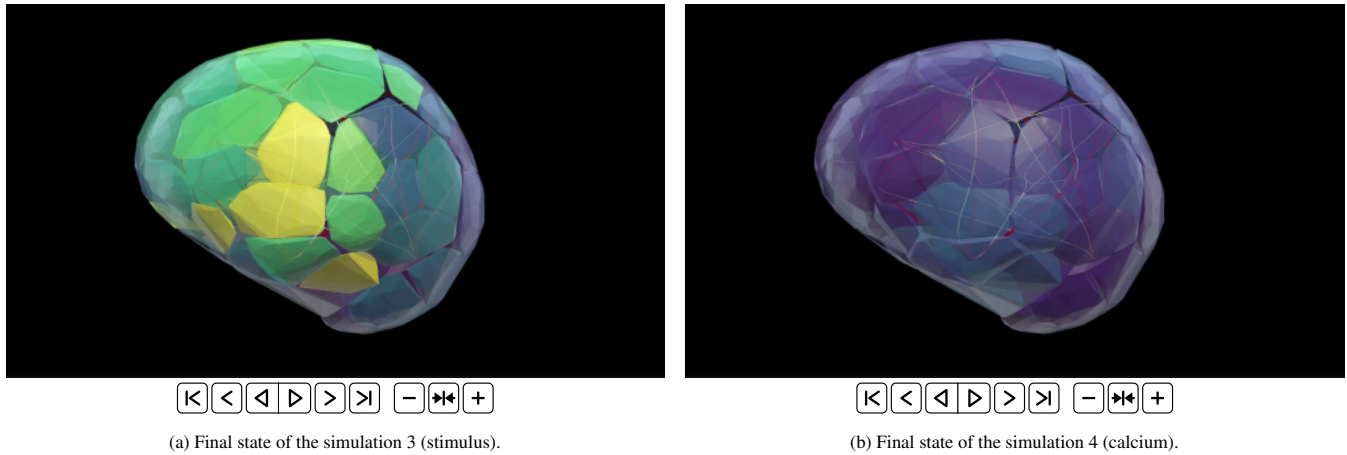


Figure 11: Showing the regional discrepancies of connected axons at the end of simulation 3 (disable) and of simulation 4 (calcium).

APPENDIX D: (2) CAN IT BE DETERMINED WHEN THE NEURONS REACHED A STEADY STATE USING THE CALCIUM LEVELS AND THE FLUCTUATION IN GROWN ELEMENTS? (3) IS THAT THE CASE FOR ALL NEURONS AND/OR SIMULATIONS SIMULTANEOUSLY?

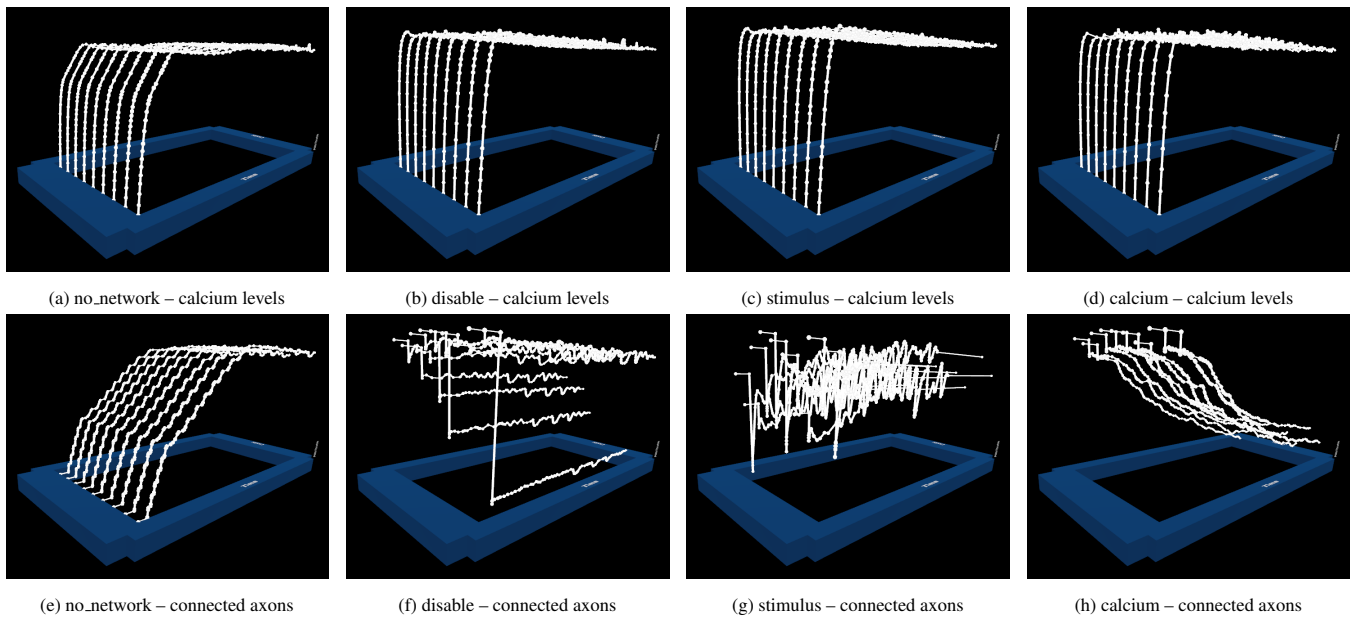


Figure 12: Showing calcium levels (top row) and connected axons (bottom row) to determine the steady state across simulations for the top-most clusters. All simulations gradually converge and flatten into a steady state, except for the connected axons in simulation 4 (calcium).

APPENDIX E: (4) HOW ARE STILL-ALIVE NEURONS AFFECTED BY THE LOSS OF NEIGHBORING NEURONS BASED ON LESIONS?

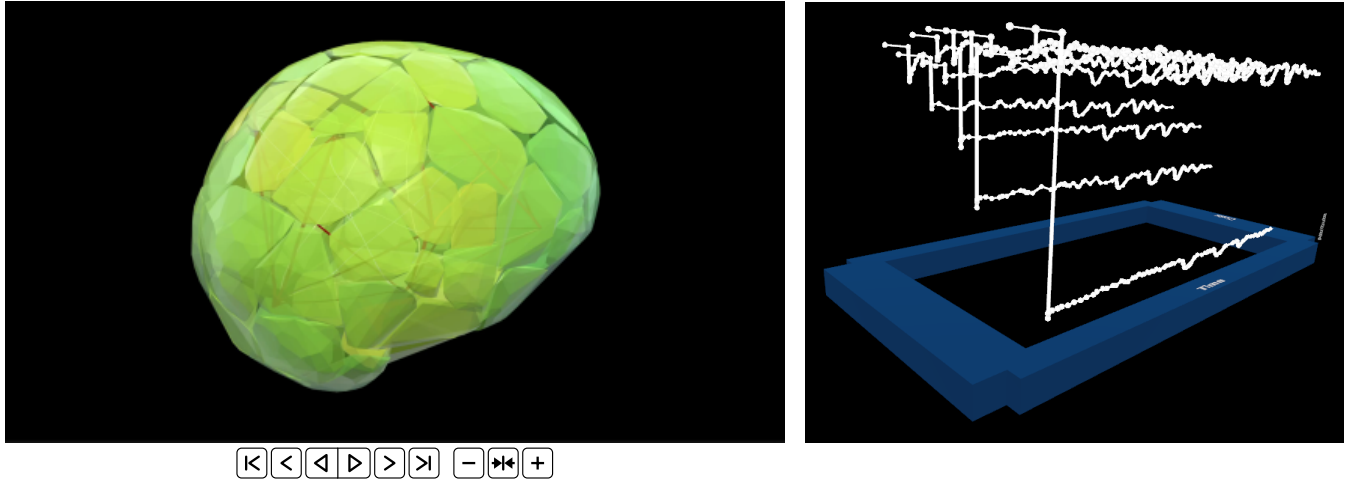


Figure 13: Neuronal and temporal overview of the number of connected axons over the course of simulation 2 (disable), showing the effect of a lesion on neighboring neuronal regions. Note: To view the animation (left) a stand-alone PDF viewer like Adobe Reader is needed. Alternatively, selected frames of the video are shown below in Fig. 14.

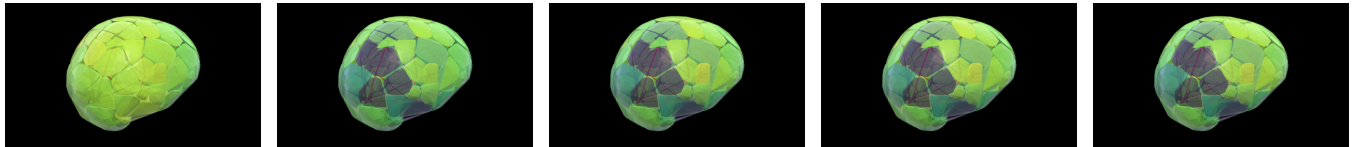


Figure 14: Evolution of the number of connected axons per cluster in simulation 2 (disable). After the initial drop, connectivity remains largely constant.

APPENDIX F: (5) DOES “LEARNING” IMPROVE THE CONNECTIONS OF GROUPS OF NEURONS?

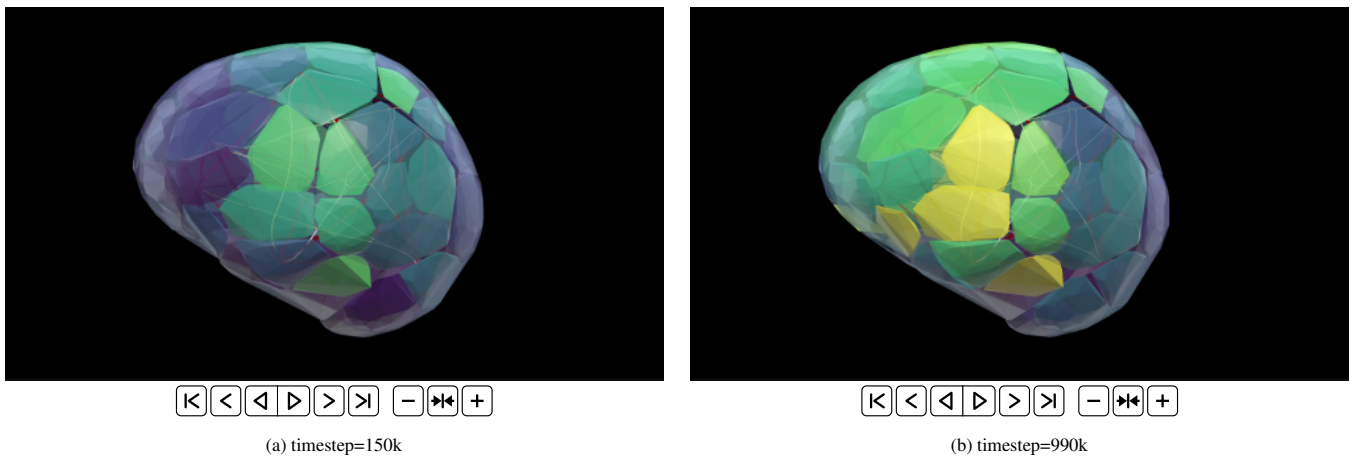


Figure 15: Animations showing the number of connected axons within each cluster in simulation 3 (stimulus) – color coded with Viridis, yellow (high) and blue (low) – at the beginning and at the end of the simulation run. It is evident that connectivity within individual brain regions increases. Note: To view the animations, a stand-alone PDF viewer like Adobe Reader is needed. Alternatively, selected frames of the “after state” (timestep = 990k) are also shown below in Fig. 16.

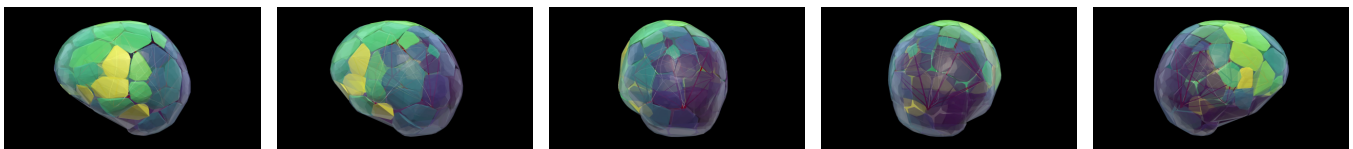


Figure 16: Selected perspectives on the end state of simulation 3 (calcium) showing the internal connectivity (connected axons) for each cluster.