

Integration neuronaler Modalitäten für Konnektomanalysen und populationsbasierte Simulationen

Aus dem Institut für Anatomie der
Universität Rostock
Dissertation zur Erlangung des akademischen Grades eines
Dr. rer. hum.

vorgelegt von
Herr Dipl.-Math. Peter Eipert
aus
Eisenhüttenstadt

Rostock, 2015

Dekan: Prof. Dr. med. Emil Christian Reisinger

1. Gutachter: Prof. Dr. Oliver Schmitt
Institut für Anatomie, Universität Rostock
2. Gutachter: Prof. Dr. Konrad Engel
Institut für Mathematik, Universität Rostock
3. Gutachter: Prof. Dr. Marcus Kaiser
School of Computing Science, Newcastle University

Datum der Einreichung: 05.05.2015

Datum der Verteidigung: 27.10.2015

Danksagung

An erster Stelle möchte ich mich ganz herzlich bei Herrn Prof. Dr. Oliver Schmitt bedanken, ohne dessen Visionen und Arbeit diese Dissertation nicht möglich gewesen wäre. Besonders wertvoll war das aufwändige Zusammenstellen der Daten und der Austausch über neue Ideen oder Probleme bei der praktischen Anwendung von *neuroVIISAS*.

Erik Viertel möchte ich danken, für die Mitarbeit an *neuroVIISAS*, insbesondere der Integration von InfoNode und der Motivsuche.

Ebenfalls möchte ich Prof. Dr. Andreas Wree, Prof. Dr. Klaus-Peter Schmitz, Prof. Dr. Klaus Neymeyr, Prof. Dr. Manfred Tasche und Frau Prof. Dr. Hei-drun Schumann für ihre Unterstützung danken.

Abschließend möchte ich mich bei meinen Eltern bedanken, die mich immer auf meinem Weg unterstützt haben.

Zusammenfassung

Die hohe Komplexität der Nervensysteme höherer Spezies, wie den Säugetieren, stellt die Wissenschaft bei deren Erforschung vor besondere Herausforderungen. Die Fokussierungen auf einzelne Teilaspekte liefern eine Vielzahl von Daten unterschiedlicher Modalitäten und Skalierungen. Dies sind z.B. topografische und funktionelle Definitionen von Neuronenpopulationen, deren neurophysiologische und molekularbiologische Charakterisierung, neuronale Verbindungen auf unterschiedlichen Skalenebenen und Atlas- bzw. Bilddaten unterschiedlicher Auflösung. Mit dieser Arbeit wird ein neues Werkzeug vorgestellt, mit dem es möglich ist, neuronale Verbindungsdaten, regionspezifische Daten, Bilddaten und räumliche Informationen unterschiedlicher Quellen und Spezies in einem System zu integrieren und auszuwerten. Das mit Java erstellte Programm *neuroVIISAS* ist unter Windows, Linux und Mac OS verwendbar und bietet viele Methoden zur Visualisierung, Filterung und Analyse von Daten. Neuronenpopulationen können hierarchisch gegliedert und deren Bild- und Konturdaten in 2D und 3D visualisiert werden. Verbindungsdaten können auf unterschiedlichen Ebenen ausgewählt, gefiltert, visualisiert und graphentheoretisch ausgewertet werden. Darüber hinaus ist es möglich, die gesammelten Informationen für eine populationsbasierte neuronale Simulation mit der Simulationssoftware NEST zu verwenden.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Existierende Projekte	2
1.2	Zielsetzung	4
2	Material und Methode	5
2.1	Auswahl der vorliegenden Daten	6
3	Ergebnisse	9
3.1	Multilingualität	9
3.2	Referenzierung der Daten	10
3.3	Datenintegration	10
3.4	Analysemethoden	16
3.4.1	Netzwerke	18
3.4.2	Filtern von Verbindungen	22
3.4.3	Matrizen	23
3.4.4	Globale Netzwerkparameter	28
3.4.5	Lokale Netzwerkparameter	30
3.4.6	Hierarchische Clusteranalyse	32
3.4.7	Motivsuche	34
3.4.8	Pfadsuche	36
3.5	Visualisierungen	38
3.5.1	Kreishierarchie	38
3.5.2	Graphenvisualisierung	39
3.5.3	3D-Visualisierung	41
3.5.4	Parametervisualisierung und Dimensionsreduktion	43
3.5.5	Vernetzung der Visualisierungen	46

3.5.6	Export	47
3.6	Simulation	48
3.7	Virtuelle Schnitte	52
3.8	Datenhaltung und Export	53
3.9	Praktischer Einsatz	55
4	Diskussion	59
4.1	Zusammenfassung	59
4.2	Anwendungen	60
4.3	Ausblick	60
	Literaturverzeichnis	67
	A Eigene Publikationen	69
	B Erklärung	73

Kapitel 1

Einleitung

Unterschiedlichste strukturelle und funktionelle Eigenschaften von Nervensystemen (NS) werden weltweit in der biomedizinischen Forschung und den Naturwissenschaften untersucht. Alivisatos et al. schreiben dazu „*Understanding how the brain works is arguably one of the greatest scientific challenges of our time.*“ [1]. Ein Schwerpunkt dabei ist das Verstehen der neuronalen Verarbeitungsprozesse (Informationsverarbeitung, Biosignalverarbeitung). Neben der medizinischen Forschung, wo das Verständnis der normalen Funktion des NS zur Entwicklung von Therapien bei dessen Erkrankungen beitragen kann, besteht auch ein Interesse daran, die Dynamik der Informationsverarbeitung zu verstehen, um sie für technische Zwecke nachbilden zu können [27]. Die Komplexität des NS stellt die neurowissenschaftliche Forschung dabei schon allein vor quantitative Herausforderungen [20]. Das Gehirn einer erwachsenen Laborratte enthält z.B. ungefähr 200 Millionen Neurone [15]. Beim Menschen sind es etwa 100 Milliarden [14]. Die Anzahl der Synapsen, die die Neurone miteinander verbinden, beträgt jeweils ein Vielfaches davon. Berücksichtigt man die räumlich-zeitliche Plastizität der Synapsen und die Dynamik des neuronalen Zellumsatzes neuroregenerativer Zonen, so betrachtet man ein hochkomplexes nicht-statisches System. Um sich dem Problem der Komplexität zu nähern und Teilprobleme zu lösen, gibt es unterschiedliche Herangehensweisen. Ein Ansatz ist herauszufinden, wie die Signalverarbeitung der einzelnen Neurone funktioniert, um sie mit Hilfe mathematischer Neuronenmodelle zu beschreiben [18]. Mit Hilfe dieser Modelle können dann einzelne Neurone oder Netzwerke von Neuronen simuliert werden [11]. Ein Problem dabei ist aber, dass die exakte

Verschaltung aller Neurone nur für kleine Systeme, wie dem NS des Fadenwurms (*C. elegans*) mit 302 Nervenzellen, bekannt ist [45]. Ein Verfahren, das zur Detektion von einzelnen synaptischen Verbindungen angewendet wird, ist die *serial block face scanning electron microscopy* (SBFSEM) [34], [26].

Ein anderer Ansatz fasst Neurone mit ähnlichen Eigenschaften zu Gruppen zusammen, die als Kerngebiete oder Hirnregionen bezeichnet werden. Diese Gruppierungen können auf unterschiedlichen Eigenschaften (z.B. räumlich, funktionell, strukturell) basieren und auf unterschiedlichen Skalenebenen liegen (Mikro-, Meso-, Makroskala). Die gebündelten neuronalen Verbindungen zwischen den Regionen können dann z.B. mit Tract-Tracing-Untersuchungen nachgewiesen werden. Man abstrahiert in diesem Ansatz von einzelnen Neuronen und Synapsen und generiert ein Netzwerk von Regionen und Regionverbindungen. Auf diesen Netzwerken können höhere Funktionen untersucht werden, wie zum Beispiel das Sehen oder, bei Einbeziehung des peripheren NS, der Informationsfluss von Gelenk- oder Hautrezeptoren in das Zentralnervensystem (ZNS).

1.1 Existierende Projekte

Es gibt verschiedene Großprojekte, wie das *Human Brain Project* (<https://www.humanbrainproject.eu>) oder die *Brain Research through Advancing Innovative Neurotechnologies (BRAIN) Initiative* (<http://braininitiative.nih.gov>), die sich zum Ziel gesetzt haben die Funktionsweise von Nervensystemen zu verstehen. Um die dafür relevanten Daten sammeln, auswerten und visualisieren zu können, werden spezielle benötigt. Das in dieser Arbeit vorzustellende Projekt ist ein solches Werkzeug, das spezies- und plattformunabhängig in einem einzigen System eine Vielzahl von Funktionen vereint, wie sie teilweise in den im Folgenden aufgeführten Systemen verfügbar sind.

Das *Brain Architecture Management System BAMS* (<http://brancusi.usc.edu>) von Michail Bota [5] stellt eine neuroanatomische Ontologie über ein Webinterface zur Verfügung. Die Ontologie besteht aus einer Hierarchie von Strukturen wie Hirnregionen oder Zellen und Beziehungen zwischen diesen, wie zum Beispiel neuronalen Verbindungen zwischen den Hirnregionen. Auf der Webseite können Regionen, Verbindungen oder Zellen gesucht, und

Informationen darüber angezeigt werden. Es besteht auch die Möglichkeit sich für eine Auswahl von Regionen eine Verbindungsmatrix anzeigen zu lassen.

Das *BrainInfo* Projekt (<http://braininfo.rprc.washington.edu>) enthält eine Datenbank mit Hirnregionen, die Informationen über Synonyme, Zellen und Genexpressionen zur Verfügung stellt. Darüber hinaus gibt es auch Hirnatlanten vom Makaken, die mit Hilfe eines Viewers online in 2D oder 3D angezeigt werden können.

Der *Allen Brain Atlas* (<http://www.brain-map.org>) des *Allen Institute* stellt Hirnatlanten der Maus zur Verfügung, mit dem Schwerpunkt auf Genexpressionen im ZNS. Darüber hinaus gibt es für die Maus Verbindungsinformationen, bei denen zu einzelnen Tracing-Experimenten die Schnittbildserien mit den Ergebnissen verfügbar sind [29].

Der *Scalable Brain Atlas* (<http://scalablebrainatlas.incf.org>) enthält Atlanten von Makaken, Mäusen, Ratten, Menschen, Seidenaffe und Opossum, die mit einem Viewer online eingesehen werden können [3].

Die *Brain Connectivity Toolbox* (<https://sites.google.com/site/bctnet/Home>) ist eine Sammlung von Matlab-Funktionen, die bei der Analyse von neuronalen Netzwerken auf Populationsebene hilfreich sein können [33].

Das Programm *Cytoscape* (<http://www.cytoscape.org>) ist eine Softwareplattform zur Visualisierung komplexer Netzwerke [38]. Diese können neben den Knoten und Kanten auch zusätzliche Daten wie Genexpressionen enthalten. Das ursprünglich für biologische Netzwerke entwickelte Programm ermöglicht neben der Visualisierung der Netzwerke auch deren Analyse. So gibt es zum Beispiel eine Clustersuche, die besonders stark verbundene Teilnetze, sogenannte Cluster, identifiziert.

Die Graphenvisualisierungsplattform *Gephi* (<http://gephi.github.io>) ist ein Werkzeug zur interaktiven Visualisierung und Analyse von Netzwerken [4]. Es kann Netzwerkparameter berechnen, Cluster detektieren und Zufallsnetzwerke generieren.

In [39] wird das Programm *neuroMap* beschrieben, das für die Visualisierung des Nervensystems der Fruchtfliege entwickelt wird. Der Schwerpunkt von *neuroMap* liegt dabei auf einer interaktiven Darstellung von Netzwerken von Neuronen.

Zu Beginn dieser Arbeit war es mit den verfügbaren Softwaresystemen nicht

möglich, eine Hierarchie von Regionen mit Verbindungs- und Atlasdaten auf unterschiedlichen Skalenebenen so zu integrieren, dass die Daten innerhalb einer einzigen Analyseumgebung ausgewertet, visualisiert und für Simulationen und Modellbildungen genutzt werden können.

1.2 Zielsetzung

Olaf Sporns Satz *„We cannot fully understand brain function unless we approach the brain on multiple scales, by identifying the networks that bind cells into coherent populations, organize cell groups into functional brain regions, integrate regions into systems, and link brain and body in a complete organism.“*, aus dem Buch *Networks of the Brain* [41] beschreibt sehr gut die Motivation dieser Arbeit. Es soll ein plattformunabhängiges Werkzeug entwickelt werden, mit dem es möglich ist, die unterschiedlichen Daten, die zum Verständnis der Funktion eines Gehirns beitragen können, zusammenzuführen und zu analysiert. Den Ausgangspunkt bilden dabei die Verbindungen im Nervensystem auf verschiedenen Skalenebenen. Verfügbare Informationen zu diesen Verbindungen sollen zusammen mit anderen Daten, die mit diesen in Zusammenhang stehen, in dem System integriert werden. Dazu gehören unter anderem auch Informationen über die verbundenen Neurone, beziehungsweise Neuronenpopulationen. Die Verschiedenartigkeit der zu integrierenden Daten stellt dabei eine besondere Herausforderung dar. Existierende Analyse- und Visualisierungsmethoden sollen in das System implementiert und wenn nötig angepasst oder erweitert werden. Die gesammelten Daten und Analyseergebnisse sollen exportierbar sein, um diese auch externen, noch nicht implementierten Analyse- oder Visualisierungswerkzeugen zugänglich zu machen. Darüber hinaus sollen sie auch für externe Systeme zur Modellbildung und Simulation bereitgestellt werden, so dass Untersuchungen von dynamischen Eigenschaften des NS möglich sind. Ein neuer Aspekt besteht darin, dass das zu modellierende neuronale System in eine natürliche anatomische Modellumgebung integriert werden soll. Dieser intuitive Ansatz erlaubt die Berücksichtigung von externen Einflüssen auf das zu betrachtende System.

Kapitel 2

Material und Methode

Um die geforderte Plattformunabhängigkeit des zu entwickelnden Programms zu gewährleisten, wurde die Programmiersprache Java von Sun Microsystems (2010 von Oracle übernommen) gewählt. In Java geschriebene Programme werden in einer virtuellen Maschine, der JVM, ausgeführt. Diese bildet zusammen mit den Standardbibliotheken die Java Laufzeitumgebung (JRE), die es für die Systeme Linux, Mac OS, Solaris und Windows gibt. Java-Programme, die ohne externe plattformspezifische Bibliotheken auskommen, lassen sich unter allen Systemen ausführen, für die es die Java Laufzeitumgebung (JRE) gibt. Als integrierte Entwicklungsumgebung (IDE) kam Eclipse zur Anwendung. Es wurden folgende externe Java-Bibliotheken eingebunden:

- *InfoNode Docking Windows*,
- *Yusuke Kamiyamanes Fugue Icons*,
- *Michael Thomas Flanagans Java Scientific Library*,
- *Cern Colt*,
- *yFiles*,
- *JavaHelp*,
- *Thomas Abeels EpsGraphics* und
- *Apache PDFBox*.

Neben den Java-Bibliotheken, die plattformunabhängig sind, wurden zwei externe plattformspezifische Bibliotheken verwendet, die für das jeweilige Betriebssystem kompiliert und dem Programm beigelegt wurden. Für die 3D-Rekonstruktion und 3D-Visualisierung wurde das *Visualization Toolkit* (VTK) verwendet und zum Öffnen von Virtuellen Schnitten, die Bibliotheken von *OpenSlide*. Die *OpenSlide*-Bibliotheken sind bisher noch nicht für die Mac OS-Version kompiliert. Zum Erstellen eines Installers für Windows wurde das Programm *Inno Setup* (<http://www.jrssoftware.org/isinfo.php>) verwendet.

2.1 Auswahl der vorliegenden Daten

Der Großteil der in dieser Arbeit verwendeten Daten wurde im Institut für Anatomie Rostock generiert und stammt von der Labormaus. Dieses Säugetier wurde gewählt, da von diesem die meisten Verbindungsdaten vorliegen. Die Gesamtheit der bekannten neuronalen Verbindungen in einem Nervensystem wird als Konnektom bezeichnet, ein Begriff, der 2005 von Hagmann [13] und Sporns et. al. [42] eingeführt wurde. Die Wahl der Labormaus bedeutet keine Einschränkung auf diese Spezies. Der generische Ansatz erlaubt auch die Verwendung von Daten anderer Spezies, wie zum Beispiel Konnektome der humanen *UCLA Multimodal Connectivity Database*, von *CoCoMac* (Makaken), dem Allen Institute (Maus) oder dem *Open Connectome Project* (diverse Spezies).

Den Ausgangspunkt der Daten bilden Regionen oder Gebiete, wie sie z.B. in den Atlanten von Swanson [43] und Paxinos [31] beschrieben werden. Dies sind räumlich begrenzte Neuronenpopulationen mit charakteristischen strukturellen oder funktionellen Eigenschaften. Je nach Auswahl können diese Regionen eine räumliche Partitionierung des Nervensystems in disjunkte Teilgebiete oder aber auch eine Hierarchie von ineinander verschachtelten Regionen bilden, wobei die Gebiete verschiedener Zweige der Hierarchie disjunkt sind. Die Gesamtheit aller in den Publikationen beschriebenen Regionen lässt sich aber nicht in einer einzigen solchen Hierarchie widerspruchsfrei abbilden. Die Ursache ist, dass es verschiedene Kriterien gibt, nach denen die Gebiete definiert werden und die zu verschiedenen Partitionierungen und Hierarchien führen können. Viele Regionen haben mehrere unterschiedliche Bezeichnungen und auch die

verwendeten Abkürzungen sind nicht eindeutig definiert. Die *Amygdala* wird zum Beispiel auch als *Amygdalar complex*, *Amygdaloid complex*, *Amygdaloid corpus* oder *Corpus amygdaloideum* bezeichnet und verwendete Abkürzungen sind unter anderem *AC*, *AMY* oder *Am*. Eine Eigenschaft des Gehirns der meisten höheren Organismen ist die symmetrische Aufteilung in eine linke und eine rechte Hemisphäre. Die Regionen der linken Hemisphäre finden sich spiegelbildlich ebenso in der Rechten. In Publikationen (z.B. Atlas von Swanson [43]) wird ein Gebiet mitunter nur in einer Hemisphäre beschrieben und impliziert, dass es in der anderen spiegelsymmetrisch mit analogen Eigenschaften existiert. Da die linken und rechten Ausprägungen aber zum Teil räumlich getrennte Regionen sind, werden sie im Folgenden auch als verschiedene Strukturen betrachtet. Dies hat den Vorteil, dass auch Nervensysteme mit strukturellen oder funktionellen Lateralitäten betrachtet werden können. Auch Regionen, die in der sagittalen Medianebene und damit in beiden Hemisphären liegen und für die ipsi- und kontralaterale Verbindungen beschrieben wurden, wie das periaquäduktale Grau oder der mediane Raphe Kern, können in einen linken und einen rechten Teil gegliedert werden. Zu den Gebieten gibt es teilweise räumliche Informationen, wie z.B. die Atlanten von Swanson [43] und Paxinos [31], wo Regionen innerhalb einer, mit einem stereotaktischen Koordinatensystem versehenen, histologischen Schnittbildserie kartiert sind. Eine vollständige Ratte in Form von segmentierten Schnittbildern wird in *A high-resolution anatomical rat atlas* [2] vorgestellt. Die dazu verfügbaren Daten sollen in das zu entwickelnde System integriert werden können, um sie für Visualisierungen unterschiedlicher Dimensionalität verwenden zu können und sie mit den Konnektomdaten zu fusionieren.

Darüber hinaus gibt es auch weitere Daten, wie zum Beispiel Informationen über Neuronentypen und deren neurophysiologische Parameter oder topografische und funktionelle Definitionen von Regionen, die in Textform in Datenbanken wie *BrainInfo*, *NeuroElectro* und *NeuroLex* oder Büchern wie die *Desk reference for neuroscience* [23] verfügbar sind.

Tract-Tracing Studien liefern Daten über die neuronalen Verbindungen zwischen den Hirnregionen. In Metastudien können diese Verbindungsdaten aus zahlreichen Publikationen zusammengetragen und in Tabellen überführt werden, die dann maschinell verarbeitbar sind. Dabei müssen Ausgangs- und Ziel-

region einer Verbindung angegeben werden. Oft sind zusätzliche Informationen verfügbar, wie zum Beispiel die Stärke der Verbindung, die Tracersubstanz oder die Tracertransportrichtung (anterograd oder retrograd). Darüber hinaus gibt es auch Verbindungen, die als axonale Kolateralen oder polysynaptische Pfade beschrieben werden. In den Studien werden entweder bereits in anderen Publikationen verwendete Regionenbezeichnungen für die Ursprungs- und Zielgebiete übernommen oder aber auch neue Gebiete definiert, wie zum Beispiel *Rostrocaudal column in the dorso intermediate region of the pons* [6]. Die eindeutige Zuordnung zu einem bestehenden Datensatz ist dabei nicht immer einfach und bedarf neuroanatomischen Fachwissens. Das Auswerten der Tract-Tracing Studien und Erstellen der Hierarchie der Hirnregionen wurde und wird aktuell auch weiterhin von Prof. Dr. Oliver Schmitt, Mitarbeitern und einem Team von Doktoranden und Studenten der Universität Rostock übernommen.

Bilddaten zu Gehirnen gibt es in Form von Schnittbildserien, die aus unterschiedlichen Quellen stammen, wie zum Beispiel MRT, CT oder histologische Schnitte. Je nach Verfahren sind unterschiedliche maximale Auflösungen und daraus resultierende Analysen möglich. So erlauben hoch aufgelöste histologische Schnittbildserien beispielsweise die Erkennung und Klassifikation einzelner Zellen. Die Bilddaten können in unterschiedlichen Formaten vorliegen, wie zum Beispiel Serien von Einzelbildern (JPG, TIF, PNG,..), virtuelle Schnitte (Mirax MRXS, Leica SCN,..) oder mehrdimensionale Bilddaten (NIfTI-1, DICOM, MINC 2.0,..). Für einen Import dieser Formate können in vielen Fällen existierende freie Bibliotheken verwendet werden.

Kapitel 3

Ergebnisse

Im Rahmen dieser Arbeit wurde das Programm *neuroVIISAS* (neuro Visualization, Imagemapping, Information System for Analysis and Simulation) entwickelt, das die Integration und Analyse von Daten von Nervensystemen bereitstellt. In der aktuellen Version 0.336 (Stand 20.04.2015) umfasst der Quellcode über 150000 Codezeilen in 585 Dateien. Das Programm wurde unter Windows XP, Windows 7 (32bit und 64bit), Suse Linux in verschiedenen Versionen und Mac OS X getestet. Es befindet sich aktuell im praktischen Einsatz im Institut für Anatomie der Universität Rostock. Im Folgenden werden zentrale funktionelle Eigenschaften von *neuroVIISAS* beschrieben.

3.1 Multilingualität

Um eine globale Verwendung von *neuroVIISAS* zu erleichtern, wurde das Programm multilingual konzipiert. Bisher stehen eine deutsche und eine angloamerikanische Version zur Verfügung. Das Hinzufügen neuer Sprachen ist ohne Programmcodeänderungen möglich. Jeder Nutzer kann durch das Übersetzen der XML-Datei mit den englischen Texten eine neue XML-Sprachendatei erstellen und durch deren Platzieren im Language-Unterverzeichnis von *neuroVIISAS* diese verfügbar machen. Die umfangreiche Hilfefunktion liegt bisher nur als angloamerikanische Version vor.

3.2 Referenzierung der Daten

Um die Quellen der Daten, die mit *neuroVIISAS* zusammengetragen werden, zu dokumentieren, ist es an vielen Stellen möglich eine Quellangabe zu machen. Dazu wird eine BibTeX-Datei verwendet, auf deren Einträge man über den BibTeX-Schlüssel in *neuroVIISAS* verweisen kann. Bei der Erstellung der BibTeX-Datei hat sich das Programm JabRef als hilfreich erwiesen. Mit jedem BibTeX-Schlüssel kann in *neuroVIISAS* ein Link auf eine Webseite verknüpft werden und ein relativer Pfad zu einem Dokument innerhalb eines festgelegten Dokumentenverzeichnis. Dadurch ist ein direkter Zugriff auf die Quellinformationen möglich. Die BibTeX-Datei und das Dokumentenverzeichnis können für jedes Projekt einzeln gewählt werden.

3.3 Datenintegration

Das Programm *neuroVIISAS* bietet die Möglichkeit Projekte anzulegen, in denen die verschiedenen Daten zusammengeführt und abgespeichert werden. Innerhalb eines Projekts können Regionen angelegt, die jeweils durch mehrere Namen und Abkürzungen beschreibbar sind. Jeder Name einer anatomischen Struktur und jede Abkürzung kann mit einer Quellangabe verknüpft werden. Ein definierter primärer Name dient dabei der eindeutigen Identifizierung und sollte nicht doppelt verwendet werden. Regionen können einer Hemisphäre (linke oder rechte) zugeordnet werden. Zusätzlich ist es möglich den Regionen weitere Informationen in Form von Tabellen zuzuordnen. In den Tabellen können Textinformationen, Literaturverweise, Links, Dokumente oder Bilder (als verlinkte Dateien) eingetragen werden. Die Regionen können hierarchisch unter einem Wurzelknoten angeordnet werden und es ist möglich, neben der Basishierarchie, die alle angelegten Regionen enthält, auch Hierarchievarianten zu erstellen, in denen die Regionen anders angeordnet oder auch entfernt werden können. Alle Analysen können dann auf einer gewählten Hierarchievariante durchgeführt werden. Wird nur eine Liste von Regionen ohne Hierarchiestruktur untersucht, so ist es möglich diese direkt unter dem obligatorisch existierenden Wurzelknoten einzufügen.

Um die Regionen mit räumlichen Daten zu verbinden, lässt sich ein Projekt

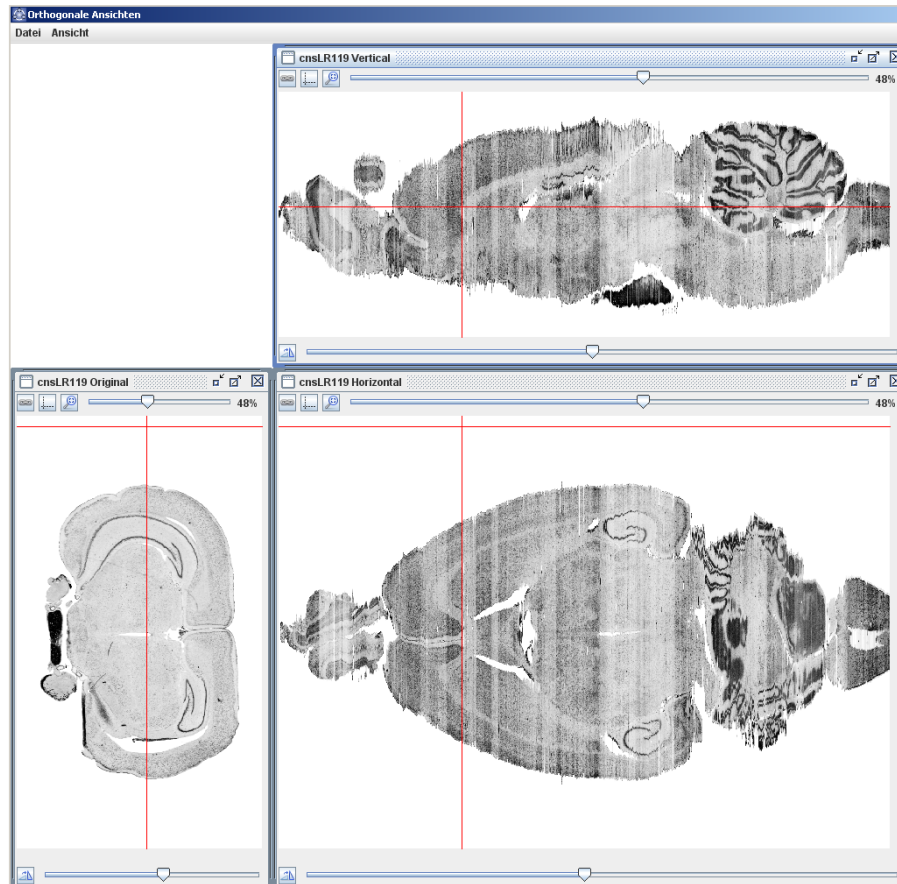


Abbildung 3.1: Orthogonale Ansichten einer Schnittbildserie in *neuroVIISAS*

mit einer Schnittbildserie verknüpfen, die als Menge von Einzelbildern vorliegen muss. Zulässige Bildformate sind dabei TIF, PNG, BMP, JPG und GIF. Es ist möglich in *neuroVIISAS* zu einer importierten Schnittbildserie die orthogonalen Schnittbildserien in horizontaler und vertikaler Richtung berechnen zu lassen. Diese drei Ansichten können dann zusammen in einer gemeinsamen Ansicht (Abbildung 3.1) angezeigt werden, wobei die drei Ansichten gekoppelt sind und ein Rechtsklick in einem Bild zu den entsprechenden Schnittbildern in den anderen beiden Ansichten springt.

Auf einer Schnittbildserie können Koordinatenachsen definiert werden, wobei pro Richtung auch die Definition mehrerer verschiedener Achsen möglich ist. Dazu müssen für eine Koordinatenachse entweder mindestens zwei Punkte mit zugehörigen Koordinaten angegeben werden oder ein Punkt und die Pixeldimension in dieser Richtung. Im ersten Fall ist es so möglich, stückweise unterschiedliche Pixeldimensionen zu definieren. Das ist insbesondere für die

Richtung senkrecht zu den Schnittbildern wichtig, da dadurch auch Schnittbildserien mit nicht konstanten Schnittdicken oder Schnittabständen korrekt abgebildet werden können. Innerhalb der 2D- und 3D-Visualisierungen ist es möglich, die darzustellenden Koordinatenachsen auszuwählen und die Koordinaten des Mauszeigers bezüglich einer Auswahl von Achsen anzeigen zu lassen. Man kann darüber hinaus innerhalb der Schnittbildserie zu einer gesuchten Koordinate springen und diese mit einem Kreis markieren lassen.

Innerhalb einzelner Schnittbilder ist es möglich einer Region eine Fläche zuzuordnen, die das Innere der Region darstellt. Diese Fläche muss nicht zusammenhängend sein und darf auch Löcher enthalten. Realisiert wird das durch geschlossene Polygonzüge (eine Folge zusammenhängender Geraden), die die Ränder der Region definieren. Polygonzüge innerhalb eines äußeren Polygonzuges werden als Löcher betrachtet. Es ist auch möglich innerhalb von Löchern Polygonzüge zu definieren, die dann wieder das Äußere der Region darstellen. Die Konturen der Regionen können entweder manuell in *neuroVIISAS* in die Schnittbilder eingezeichnet oder importiert werden. Ein Editor ermöglicht die Bearbeitung der Konturen, wobei Endpunkte von Kanten entfernt, hinzugefügt und verschoben werden können. Zusätzlich ist es auch möglich einen ganzen Polygonzug zu rotieren, zu verschieben oder zu skalieren. Zu importierende Konturen müssen entweder als segmentierte Bilder vorliegen oder in einer einfachen XML-Datei, in der die Polygonzüge in der Form

```
<Object1>
  <X>1</X>
  <Y>1</Y>
</Object1>
```

beschrieben sind. Der Name *Object1* ist dabei die Identifizierung (ID) eines geschlossenen Polygonzuges. Jeder Punkt des Polygonzuges, in der Reihenfolge wie sie durchlaufen werden, wird mit einem solchen XML-Block dargestellt. Nach dem Import müssen die Polygonzüge dann einer Region zugeordnet werden, wofür ein Dialog zur Verfügung steht. Sind in einem Schnittbild zu einer Region Konturdaten vorhanden, so ist es möglich, durch einen Klick in das Schnittbild, zu deren Position in der Baumdarstellung der Hierarchie zu springen. Dort ist jeder Region eine Auswahlbox zugeordnet, über die festgelegt werden kann, ob die Konturen dieser Region innerhalb der Schnittbilder

sichtbar sein sollen. Dadurch kann eine beliebige Auswahl von Regionen für die Visualisierung zusammengestellt werden. Optional können die Konturen gefüllt und teilweise transparent dargestellt werden, um so zum Beispiel die Gebiete innerhalb von histologischen Schnitten einzufärben.

Die neuronalen Verbindungen zwischen Regionen sind in *neuroVIISAS* immer gerichtete Verbindungen von einer Ausgangs- zu einer Zielregion. Sie können manuell angelegt oder in Form von Tabellen im CSV-Format importiert werden. Eine Verbindung enthält immer mindestens die Ausgangs- und Zielregion. Darüber hinaus können noch beliebige weitere Informationen in Textform angegeben sein. Einige häufig vorkommende Kategorien sind fest implementiert, wie zum Beispiel die Referenz auf die Publikation in der die Verbindung beschrieben wurde. Diese ist wichtig, um nachvollziehen zu können, woher die Informationen stammen. Zusätzlich ermöglicht die Angabe der Referenz auch zu analysieren, wie viele verschiedene Publikationen eine bestimmte Verbindung beschrieben haben. Ist die Publikation unbekannt, dann kann hier der Eintrag *unknown* als Referenz verwendet werden, welche sich später bei einer Analyse gezielt herausfiltern und zum Beispiel ausschließen lässt. Darüber hinaus kann die Stärke der Verbindung in Form von semiquantitativen Gewichten angegeben werden, für die die folgende Werte zur Verfügung stehen:

- unknown
- fibers of passage
- not clear
- exists
- not present
- very light
- light/sparse
- light/moderate
- moderate/dense
- moderate/strong
- strong
- very strong

Diese Definition der Gewichte basiert auf den verwendeten Gewichtsbeschreibungen in den bisher ausgewerteten Publikationen. Der Wert *not present* beschreibt dabei, dass eine Verbindung explizit nicht existiert. Diese Information

dient der Unterscheidung von nicht untersuchten und nicht existenten Verbindungen. Wird eine Verbindung in mehreren Experimenten untersucht, so können deren Ergebnisse separat in *neuroVIISAS* eingetragen werden. Dadurch ist es möglich, dass zu einer Verbindung unterschiedliche und auch widersprüchliche Informationen existieren. So gibt es unter anderem Fälle, wo Verbindungen in einer Publikation als existent und in einer anderen als nicht existent beschrieben werden. Diese Unterschiede und Widersprüche innerhalb der Verbindungsinformationen können in *neuroVIISAS* festgestellt, visualisiert und derart gefiltert werden, dass nur besonders zuverlässige Verbindungen für weiteren Analysen ausgewählt werden.

Für dem Import von Verbindungen über eine CSV-Datei, muss diese so aufgebaut sein, dass jede Zeile eine Verbindungsbeschreibung enthält. Der Nutzer kann ein Trennzeichen für die Spalten wählen, wobei das Tabulatorzeichen, das Leerzeichen, der Punkt, das Komma, der Doppelpunkt oder das Semikolon zur Verfügung stehen. Es kann weiterhin angegeben werden, ob die Tabelle eine Tabellenkopfzeile enthält, die beim Import übersprungen werden soll. Im nächsten Schritt müssen die Spalten einzelnen Rubriken zugeordnet werden, wobei mindestens die Spalten der Quellregion, der Zielregion und der Lateralität angegeben werden müssen. Die Regionen werden dabei über ihre primäre Abkürzung identifiziert, die keine Seiteninformationen enthalten. Die Lateralität gibt dann an, welcher Hemisphäre die Quell- und Zielregionen der Verbindung angehören, um sie eindeutig einer Region in *neuroVIISAS* zuordnen zu können. Die folgenden Einträge sind dabei möglich:

- **IPSI** - Die Verbindung läuft von Links nach Links und von Rechts nach Rechts.
- **CONTRA** - Die Verbindung läuft von Links nach Rechts und von Rechts nach Links.
- **IPSICONTRA** - Es werden alle Verbindungen der Fälle IPSI und CONTRA angelegt.
- Sonst wird eine Kombination aus zwei der Buchstaben **L**, **R** und **U** eingetragen, wobei L für Links, R für Rechts und U für eine unpaare Struktur steht und der erste Buchstabe für die Quellregion und der zweite für die Zielregion der Verbindung steht.

In den ersten beiden Fällen werden also jeweils zwei Verbindung angelegt und

The screenshot shows a software window titled "Projektstatistik cnsLR119" with a subtitle "Verbindungen zwischen Hierarchieebenen". The main part of the window is a 21x21 matrix of numbers representing connections between hierarchical levels 3 to 21. Below the matrix are four summary tables:

Outdegree	Ipsilateral	Outdegree	Kontralateral	Indegree	Ipsilateral	Indegree	Kontralateral
Lateral_hypothalamic_area_L	1373	Locus_coeruleus_R	730	Lateral_hypothalamic_area_L	801	Superior_colliculus_L	490
Lateral_hypothalamic_area_R	1372	Locus_coeruleus_L	729	Lateral_hypothalamic_area_R	801	Superior_colliculus_R	490
Locus_coeruleus_L	1103	Gigantocellular_reticular_nucleus_L	469	Nucleus_of_the_solitary_tract_L	770	Periaqueductal_gray_L	447
Locus_coeruleus_R	1102	Gigantocellular_reticular_nucleus_R	469	Nucleus_of_the_solitary_tract_R	770	Periaqueductal_gray_R	447
Gigantocellular_reticular_nucleus_L	792	Pedunculopontine_tegmental_nucleus_L	352	Paraventricular_hypothalamic_nucleus_L	720	Nucleus_of_the_solitary_tract_L	428
Gigantocellular_reticular_nucleus_R	792	Pedunculopontine_tegmental_nucleus_R	352	Paraventricular_hypothalamic_nucleus_R	720	Nucleus_of_the_solitary_tract_R	427
Raphe_magnus_nucleus_L	704	Primary_somatosensory_cortex_L	340	Parabrachial_nucleus_L	679	Ventrolateral_periaqueductal_gray_L	420
Raphe_magnus_nucleus_R	704	Primary_somatosensory_cortex_R	340	Parabrachial_nucleus_R	679	Ventrolateral_periaqueductal_gray_R	420
Koelliker_Fuse_nucleus_L	642	Lateral_agranular_prefrontal_cortex_L	321	Ventrolateral_periaqueductal_gray_L	583	Cuneiforme_nucleus_L	415
Koelliker_Fuse_nucleus_R	642	Lateral_agranular_prefrontal_cortex_R	321	Ventrolateral_periaqueductal_gray_R	583	Cuneiforme_nucleus_R	415

Abbildung 3.2: Ausschnitt aus den Statistiken eines aktuellen Konnektomprojektes zur Labormaus

im dritten Fall vier. Die letzte Option ist dann wichtig, wenn Verbindungen nicht in beiden Hemisphären identisch existieren oder Verbindungen zu unpaaren Strukturen, wie zum Beispiel Bauchorganen, angelegt werden sollen. Weitere Spalten mit Rubriken sind optional, wobei für jede Spalte entweder eine der fest implementierten Rubriken gewählt oder eine neue Rubrik erstellt werden kann. Bei einer neuen Rubrik muss ein Rubrikname angegeben werden, der, wenn vorhanden, aus dem zugehörigen Tabellenkopfeintrag übernommen oder manuell eingetragen werden kann. So ist es im Rahmen von Metastudien mit mehreren beteiligten Personen möglicherweise sinnvoll, zu speichern, wer die importierten Daten zusammengetragen hat. Es ist auch möglich Spalten vom Import auszuschließen, wenn deren Daten nicht benötigt werden.

Um einen Überblick über die in einem Projekt gesammelten Regionen- und Verbindungsdaten zu bekommen, gibt es ein Fenster mit grundlegenden Projektinformationen (Abbildung 3.2). Dort kann man sich unter anderem anzeigen lassen, wie viele Regionen angelegt sind und zu wie vielen davon Kontrurdaten existieren. Es wird für jedes der möglichen Gewichte angezeigt, wie viele Verbindungen mit diesem Gewicht beschrieben wurden und wie viele Verbindungen zwischen den unterschiedlichen Hierarchieebenen verlaufen. Es gibt die Top Ten der Regionen mit den meisten Verbindungen, wobei diese nach ankommenden, abgehenden, ipsilateralen und kontralateralen Verbindungen unterschieden werden.

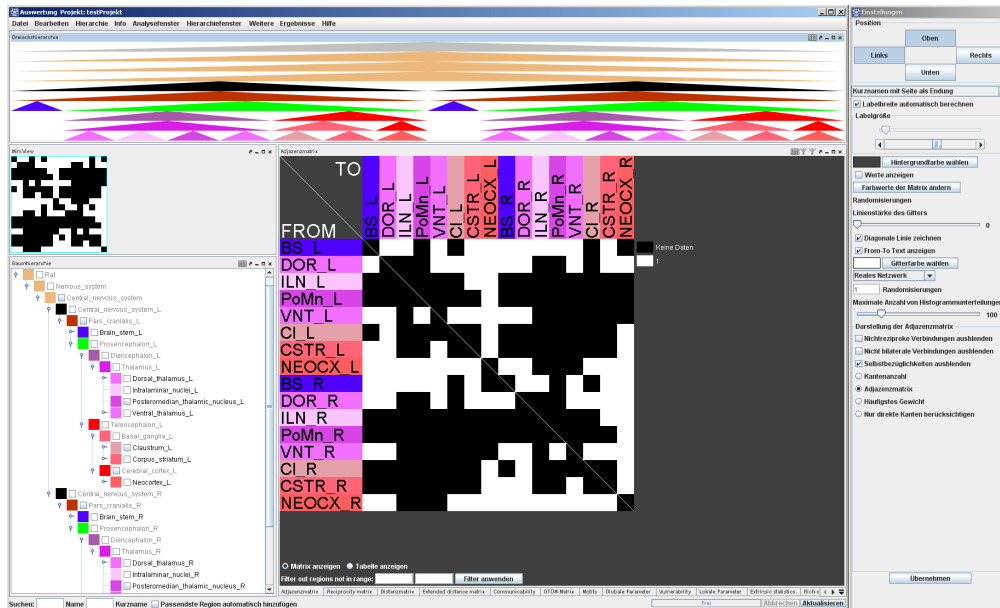


Abbildung 3.3: Konnektivitäten-Analyse-Fenster

3.4 Analysemethoden

Der Großteil der implementierten Analysemethoden untersucht Eigenschaften der neuronalen Verbindungen zwischen Regionen. Dafür gibt es ein eigenes Programmfenster, die *Erweiterte Konnektivitäten-Analyse* (Abbildung 3.3). Bei der Entwicklung der graphischen Oberfläche mit den *Docking Windows* Bibliotheken der Firma *InfoNode* und der Motivsuche war Erik Viertel im Rahmen seiner Diplomarbeit [44] beteiligt.

Um eine Analyse durchzuführen, muss eine Auswahl von Regionen festgelegt werden, zwischen denen die Verbindungen untersucht werden sollen. Diese Auswahl der Regionen ist für die Analysen von elementarer Bedeutung und, auf Grund der hierarchischen Struktur der Gebiete und dem Verlauf von Verbindungen zwischen unterschiedlichen Hierarchieebenen, nicht immer trivial. Aus diesem Grund wurden für die Auswahl von Regionen verschiedene unterstützende Funktionen implementiert. Man kann in einer Baumhierarchie die Regionen durch Auf- und Zuklappen auswählen oder einzelne Regionen über den Namen suchen und hinzufügen. Für jede Region kann mit Hilfe eines Tooltips angezeigt werden, wie viele explizite Verbindungen und wie viele Teilbaumverbindungen von der Region ausgehen oder dort ankommen, was bei der Entscheidung helfen kann, ob diese Region oder deren Unterregionen für die

Auswahl verwendet werden sollen. Es ist auch möglich Regionen automatisch hinzuzufügen, die mit bereits ausgewählten Regionen verbunden sind, wobei die Richtung der Verbindung, die Hemisphäre der hinzuzufügenden Regionen und eine maximale Hierarchietiefe vorgegeben werden können.

Da für eine Auswahl von Regionen nicht immer alle zugehörigen Verbindungen in Studien untersucht wurden, können in *neuroVIISAS* Verbindungen auf zwei verschiedene Arten definiert und analysiert werden.

1. **Explizite Verbindungen:** Eine explizite Verbindung von einer Region A zu einer Region B ist eine Verbindung, die in Studien als existent beschrieben wird, wobei dort die Regionen A und B explizit als Quell- und Zielregion genannt werden.
2. **Teilbaumverbindungen:** Eine Teilbaumverbindung von einer Region A zu einer Region B existiert immer dann, wenn mindestens eine explizite Verbindung von der Quellregion A oder deren Teilgebieten zur Zielregion B oder Teilgebieten von B beschrieben ist. Da die Teilgebiete einer Region hierarchisch als Baum unter dieser Region angeordnet sind, kann man alle diese Kanten als Verbindungen zwischen den Teilbäumen der Quell- und Zielregion bezeichnen. Aus dieser Definition folgt unter anderem, dass jede explizite Verbindung auch eine Teilbaumverbindung ist.

Das Rhombencephalon wird zum Beispiel in Metencephalon und Myelencephalon unterteilt. Eine Verbindung, die in einer Publikation mit Ziel Myelencephalon beschrieben wurde, wird bei der Verwendung von Teilbaumverbindungen auch gezählt, wenn das Ziel Rhombencephalon betrachtet wird. Eine Teilbaumverbindung kann also aus vielen expliziten Verbindungen bestehen, die zwischen den zwei betrachteten Regionen und zwischen deren Teilregionen beschrieben wurden. Dabei ist entscheidend, dass der Teilbaum unterhalb einer Region auch aus räumlich innerhalb dieser Region liegenden Gebieten besteht. Gegebenfalls kann man, mit Hilfe der Hierarchievarianten, für eine Analyse eine geeignete Hierarchie erstellen. Die Verwendung der Teilbaumverbindungen stellt ein Alleinstellungsmerkmal von *neuroVIISAS* dar und ermöglicht es, für eine Analyse alle bekannten Verbindungen einer Region zu berücksichtigen, auch wenn diese Verbindungen nicht explizit dieser Region, sondern einer

Teilregion zugeordnet sind. Die Originalinformationen aus den, die Verbindung beschreibenden, Publikationen bleibt dabei erhalten.

3.4.1 Netzwerke

Die Regionen mit ihren Verbindungen werden im Folgenden auch abstrakt als gerichteter Graph oder Netzwerk bezeichnet. Ein gerichteter Graph $G = (V, E)$ besteht aus einer Menge von Knoten $V = \{v_1, v_2, \dots\}$, in diesem Fall die Regionen, und einer Menge von gerichteten Verbindungen oder Kanten $E \subseteq V \times V = \{(v, w) | v, w \in V\}$ zwischen den Knoten von V . Liegen für die Verbindungen Gewichtsinformationen vor, die in den Analysen berücksichtigt werden sollen, können diese als reelle Zahlen den Kanten zugeordnet werden. Dadurch wird der Graph zu einem gewichteten Graphen. Die meisten der implementierten Analysemethoden stammen aus der Graphentheorie und sind nicht auf Netzwerke von Regionen des Nervensystems beschränkt. Mit *neuroVIISAS* ist es möglich auch beliebige andere Netzwerke mit gerichteten Verbindungen zu analysieren. Dazu können Knoten und Kanten als einfache Textdateien importiert werden, die entweder eine Liste von Verbindungen oder eine Verbindungsmatrix enthalten. Eine Angabe von Kantengewichten ist dabei auch möglich. Dafür stehen aktuell zehn diskrete Werte zur Verfügung (die als existent definierten semiquantitativen Verbindungsstärken), denen für die Analyse ein reeller Wert zwischen Null und Eins zugewiesen werden kann. Diese Zuweisung kann manuell geschehen, oder per Knopfdruck linear oder logarithmisch erfolgen. Dies ist für bestimmte biologische Dichteannahmen von Nervenfaserbündeln wichtig [36]. Für einige Analysen werden die Gewichte nicht als Stärke der Verbindung, sondern als Distanz zwischen den Regionen interpretiert. Aus diesem Grund können zusätzlich Gewichte für eine Distanzberechnung definiert werden. In der Anwendung als Verbindungsstärke bedeutet ein hohes Gewicht einer Kante eher eine größere Nähe der verbundenen Regionen, weshalb in der Voreinstellung die Gewichte für die Distanzberechnung umgekehrt proportional zu denen der anderen Analysen gewählt sind.

Zufallsnetzwerke

Um die Ergebnisse von Analysen der Netzwerke von Regionen besser interpretieren zu können, kann es sinnvoll sein, diese mit den entsprechenden Analyseergebnissen von Zufallsnetzwerken zu vergleichen. Um bestimmte Eigenschaften auch in den Zufallsnetzwerken nachbilden zu können, wurden verschiedene Verfahren zum Erzeugen von Zufallsnetzwerken implementiert. Alle Verfahren erzeugen dabei Netzwerke mit derselben Knoten- und Kantenanzahl wie das Netzwerk, mit dem sie verglichen werden sollen. Diese Bedingung führt im Folgenden dazu, dass die Implementierung der verschiedenen Verfahren der Zufallsmodellberechnung vom theoretischen Modell abweichen kann. Eigenkanten werden bei allen Implementationen separat behandelt und unabhängig von den anderen Kanten wieder als Eigenkanten zufällig verteilt. Es haben weder die Eigenkanten einen Einfluss auf die Verteilung der übrigen Kanten noch umgekehrt.

Das einfachste Modell ist das **Erdős-Rényi-Model**, bei dem die Wahrscheinlichkeit für die Existenz jeder Kante gleich groß ist. Um die feste Kantenanzahl zu garantieren, werden bei der Berechnung für jede Kante die Knoten zufällig bestimmt, die sie verbindet. Gibt es mehr als die Hälfte aller möglichen Verbindungen, werden aus Effizienzgründen die nicht verbundenen Knoten zufällig bestimmt. Für die folgenden Modelle, mit Ausnahme der beiden Rewiring-Modelle, wird das Erdős-Rényi-Model immer auch dann verwendet, wenn die Mindestanzahl von fünf Knoten oder zehn Kanten nicht erreicht wird oder wenn die Anzahl der Kanten den Maximalwert, von der maximal möglichen Kantenanzahl minus der Knotenanzahl, überschreitet. Das sind die Fälle, wo die Anwendung der anderen Verfahren nicht möglich oder nicht sinnvoll ist.

Das **Watts-Strogatz-Model** ist für ungerichtete Graphen so definiert, dass die Knoten in einem Ring angeordnet sind und jeder Knoten mit seinen K Nachbarknoten verbunden wird. Für jeden Knoten wird dann jede seiner Kanten mit einer gegebenen Wahrscheinlichkeit β ($0 \leq \beta \leq 1$) zu einem anderen Knoten umgelenkt, so dass keine doppelten und keine Eigenkanten entstehen. Je kleiner β gewählt wird, desto stärker ist die Clusterung und je größer β ist, desto mehr nähert sich ein generierter Graph einem Erdős-Rényi-Zufallsgraphen an. Bei der Implementierung für einen gerichteten Graphen mit fester Kantenanzahl wurde das Verfahren abgewandelt. Jeder Knoten

wird, wenn die Kantenanzahl dafür ausreicht, mit mindestens vier Nachbarn verbunden. Gibt es weniger Kanten als dafür nötig sind, werden nur so viele Knoten mit ihren vier Nachbarn verbunden wie dies mit der geforderten Kantenanzahl möglich ist. Die übrigen Knoten bleiben dann isoliert. Gibt es mehr Kanten zu verteilen, werden die Knoten mit den weiter entfernten Nachbarn verbunden, bis die geforderte Kantenzahl erreicht ist. Im Anschluss werden dann die abgehenden Kanten eines jeden Knotens mit einer vom Nutzer gewählten Wahrscheinlichkeit $0 < \beta < 1$ zu einem neuen zufällig gewählten Zielknoten umgelenkt. Die Spezialfälle $\beta = 0$ und $\beta = 1$ werden ausgeschlossen, weil der erzeugte Graph im ersteren Fall nicht zufällig ist und im zweiten Fall ein Erdős-Rényi-Graph erzeugt wird.

Das **Barabási-Albert-Model** dient der Erzeugung zufälliger skalenfreier Netzwerke. Ein Netzwerk wird als skalenfrei bezeichnet, wenn der Anteil von Knoten, mit Grad k für große k exponentiell abnimmt. Der Grad eines Knotens ist dabei die Anzahl von Kanten, an denen der Knoten beteiligt ist. Der Algorithmus ist so definiert, dass zu einer Ausgangsmenge von m_0 verbundenen Knoten sukzessive Knoten hinzugefügt und mit $m \leq m_0$ zufällig gewählten Knoten mit einer gewissen Wahrscheinlichkeit verbunden werden. Die Wahrscheinlichkeit ist dabei der Anteil des Grades des Knotens, mit dem der neu hinzukommende Knoten verbunden werden soll, an der Summe der Grade aller Knoten. Das bedeutet, dass Knoten, die schon einen hohen Grad haben, tendenziell auch mehr Kanten hinzubekommen, so dass wenige Knoten mit hohem Grad entstehen und viele mit kleinem Grad. In der implementierten Variante für gerichtete Graphen wird mit der kleinsten Startmenge von komplett verbundenen Knoten begonnen, so dass die übrigen sukzessive mit einer konstanten Anzahl von Kanten wie oben beschrieben hinzugefügt werden können. Die letzten Knoten können dabei weniger Kanten bekommen oder auch isoliert bleiben.

Die Idee des Barabási-Albert-Modells, dass Knoten mit hohem Grad bei der Vergabe neuer Kanten bevorzugt werden, wurde bei der Entwicklung einer neuen Version übernommen. Bei dem **Modifizierten Barabási-Albert-Model** hat jeder Knoten zu Beginn ein frei wählbares Gewicht $\beta > 0$. Dann werden schrittweise die Kanten hinzugefügt, wobei die Wahrscheinlichkeit an einer Kante beteiligt zu sein gleich dem Anteil des Knotengewichts am Gesamtge-

wicht aller Knoten ist. Nach dem Hinzufügen der Kante steigen die Gewichte der beiden beteiligten Knoten um jeweils eins, d.h. ihre Wahrscheinlichkeit im Folgenden wieder eine Kante zu erhalten, steigt. Mit Hilfe des Parameters β kann man steuern, wie stark dieser Effekt ist. Mit zunehmendem β reduziert sich der Einfluss der hinzukommenden Kanten und das Netzwerk nähert sich einem Erdős-Rényi-Zufallsgraphen an.

Das **Modifizierte OHO-Model** basiert auf dem Model von *Ozit, Hunt und Ott* [30], das für ungerichtete Graphen definiert ist. Ausgehend von einer Menge von $m + 1$ vollständig verbundenen Knoten, die man sich gleichmäßig auf einem Kreis angeordnet vorstellen kann, werden sukzessive neue Knoten hinzugefügt. Ein neuer Knoten wird dabei an einer zufällig gewählten Stelle zwischen zwei Knoten auf dem Kreis eingefügt und mit den m nächsten Nachbarn verbunden. Für das gerichtete Netzwerk mit n Knoten und e Kanten wurde der Algorithmus wie folgt angepasst. Die Zahl $m = \lceil \frac{e}{n} \rceil$ ergibt sich aus der Tatsache, dass pro Knoten genau m Kanten hinzugefügt werden sollen. Die $m + 1$ Startknoten werden im gerichteten Fall auch durch $(m + 1) \cdot m$ Kanten vollständig verbunden. Falls die Kantenanzahl $e < 2n$ ist, wird dieses Modell nicht verwendet und stattdessen ein Erdős-Rényi-Graph berechnet. Bei jedem Hinzufügen der neuen Knoten werden die m Kanten zufällig unter den $\lceil \frac{m}{2} \rceil$ rechten und linken Nachbarn verteilt, wobei durch die Gerichtetheit doppelt so viele Kanten möglich wären. Falls die Kantenanzahl kein Vielfaches der Knotenanzahl ist, wird der Wert m in jedem Schritt so angepasst, dass die noch zu vergebenden Kanten gleichmäßig auf die noch hinzuzufügenden Knoten verteilt werden.

Da viele der Netzwerkparameter stark von den Graden der Knoten abhängen, ist es interessant zu prüfen, wie der Graph sich von Zufallsnetzen unterscheidet, die dieselbe Gradverteilung haben. Dies ist durch das **Rewiring-Model** möglich, bei dem ausgehend vom Originalnetzwerk Kanten so umgelegt werden, dass der Grad der Knoten konstant bleibt. Dazu werden in jedem Iterationsschritt jeweils zwei Kanten zufällig gewählt und deren Zielknoten vertauscht, falls sie keinen gemeinsamen Knoten besitzen. Die Anzahl der Iterationsschritte wurde hierfür auf das Vierfache der Kantenzahl festgelegt.

Es hat sich in den Analysen der Netzwerke aus Regionen und deren Verbindungen gezeigt, dass die Anzahl reziproker Kanten signifikant höher ist,

als in allen oben beschriebenen Zufallsnetzwerken. Um die Frage zu beantworten, wo bei gleicher Gradverteilung und gleicher Anzahl reziproker Kanten vielleicht noch Unterschiede zu Zufallsnetzen bestehen, wurde das **Rewiring-Reziprok-Model** entwickelt, wo zusätzlich zur Gradverteilung auch die Anzahl reziproker Kanten identisch zum Originalnetzwerk ist. Analog zum einfachen Rewiring-Model werden jeweils zwei Kanten mit vier unterschiedlichen Knoten zufällig gewählt und die Zielknoten vertauscht, wenn dadurch die Anzahl reziproker Kanten konstant bleibt. Gilt dies nicht, aber die vier Knoten bilden zwei Paare, die jeweils mit einem Paar reziproker Kanten verbunden sind, dann können für die reziproken Kantenpaare je ein Knoten des einen Paares durch einen des anderen getauscht werden, falls diese Verbindungen nicht schon bestehen.

Die hier vorgestellten Modelle für die Generierung von Zufallsnetzwerken können in *neuroVIISAS* bei den Netzwerkanalysen gewählt werden, um deren Ergebnisse mit denen des realen Netzwerkes zu vergleichen.

3.4.2 Filtern von Verbindungen

Innerhalb einer Auswahl von Regionen ist es möglich Verbindungen zu filtern, um für die Analyse nur Verbindungen mit bestimmten Eigenschaften zu verwenden. Für die Teilbaumverbindungen stehen dazu drei Filter zur Verfügung. Man kann ein Intervall für die Anzahl der Teilbaumkanten angeben oder ein Intervall für das häufigste Gewicht innerhalb der Teilbaumkanten. Der dritte Filter, filtert die Verbindungen danach, ob die Quell und/oder Zielregionen mit einem Marker versehen sind. Man kann so zum Beispiel die Regionen in zwei Klassen einteilen und dann nur Verbindungen innerhalb der Klassen oder nur Verbindungen zwischen den Klassen untersuchen. Den letzten Filter gibt es auch für die expliziten Verbindungen. Darüber hinaus können dort noch eine Vielzahl weiterer Kanteneigenschaften gefiltert werden. So zum Beispiel die Anzahl der Publikationen, in denen die Verbindung beschrieben wurde, das mittlere Gewicht aus allen Publikationen, die Tracertypen oder auch, ob eine Verbindung Teil einer kollateralen Verbindung ist.

Alle erzeugten Filter lassen sich dann mit den logischen Operatoren *NICHT*, *UND* und *ODER* verknüpfen, so dass man jede mögliche Kombination der Basisfilter konstruieren kann.

3.4.3 Matrizen

Die Existenz einer Verbindung zwischen zwei Regionen lässt sich mit einer **Adjazenzmatrix** darstellen, wie sie in der Mitte von Abbildung 3.3 zu sehen ist. Unterhalb von *FROM* stehen die Abkürzungen der Ausgangsregionen einer Verbindung und rechts von *TO* stehen die Abkürzungen der Zielregionen. Ein weißes Kästchen in der Matrix bedeutet dabei, dass eine Verbindung von der entsprechenden Region in dieser Zeile zu der Region in der Spalte existiert und ein schwarzes, dass entweder keine solche Verbindung existiert, keine Informationen dazu vorliegen oder diese Verbindung mit Hilfe eines Filters ausgeblendet wurde. Neben einer Adjazenzmatrix, die nur die Existenz einer Verbindung beschreibt, kann man auch eine gewichtete Adjazenzmatrix erzeugen, die ein Maß für die Stärke der Verbindungen enthält, das zwischen Null und Eins liegt. Darüber hinaus gibt es noch die Möglichkeit weitere Eigenschaften der expliziten Verbindungen in Matrixform anzuzeigen. Das sind zum Beispiel die Anzahl Publikationen in denen eine Verbindung beschrieben wurde, die verwendeten Tracerrichtungen oder ein Maß für die Unterschiede in den beschriebenen Stärken der Verbindung.

Ein Maß für die Verlässlichkeit der Verbindungsinformationen in der Adjazenzmatrix liefert die sogenannte Reliabilitätsdarstellung. Zu einer expliziten Kante wird dabei ein Wert bestimmt, der umso höher ist, je mehr gleichartige Beobachtungen bezüglich der Existenz der Verbindung existieren. Die Reliabilität kann dabei auch die Verlässlichkeit der Information über die Nichtexistenz einer Kante beschreiben. Zusätzlich erhöht sich der Wert in dem Fall, dass Tracerrichtungen gemacht sind und sowohl retrograde als auch anterograde Tracer verwendet wurden. Gibt es für eine Kante sowohl Beobachtungen, dass eine Verbindung existiert als auch solche, die deren Existenz ablehnen, so sinkt der Wert für die Verlässlichkeit der Informationen. Die Formel für die Reliabilität R ist wie folgt definiert:

$$R = | (\sum w_+) + t_+ + (\sum w_-) - t_- |$$

Die Variablen w_+ und w_- sind dabei positive beziehungsweise negative Werte, die einer Verbindungsbeobachtung mit einem gewissen Gewicht zugeordnet werden können. Die Standardzuordnung wurde dabei so gewählt, dass eine Beschreibung der Nichtexistenz mit -1 gewichtet wird. Bei Existenz einer Verbin-

dung sind die Werte positiv, wobei diese für angegebene Verbindungsstärken 1 sind und für nicht Angegebene zwischen 0,7 und 0,9 liegen. Der Spezialfall *fibers of passage* wird standardmäßig mit 0 gewichtet und geht somit nicht in die Berechnung ein. Die Summanden t_+ und t_- sind frei definierbare Werte, die die verwendeten Tracerrichtungen berücksichtigen. Der Wert t_+ berücksichtigt dabei die verwendeten Tracer in den Experimenten, die die Verbindung als existent beschreiben und t_- die Tracer der Experimente, aus denen die Nichtexistenz der Verbindung gefolgert wurde. Es können separate Werte angegeben werden, für die Verwendung von ausschließlich anterograden, retrograden oder bidirektionalen Tracern und auch für jede mögliche Kombination von verschiedenen verwendeten Tracertypen. In der Standardeinstellung ist der Gedanke umgesetzt, dass ein Nachweis mit sowohl retrograden als auch anterograden Tracern zu der höchsten Reliabilität führt. Wird die Existenz einer Verbindung nur mit anterograden oder nur mit retrograden Tracern beschrieben, so ist t_+ in der vorgegebenen Einstellung gleich 0,5. Im Falle von ausschließlich bidirektionalen Tracern gilt $t_+ = 0,25$. Wurden anterograde oder retrograde zusammen mit bidirektionalen Tracern in Experimenten verwendet so ist $t_+ = 0,7$ und der Maximalwert von $t_+ = 1$ wird dann angenommen, wenn sowohl anterograde als auch retrograde Tracer zum Nachweis der Verbindung verwendet wurden. Analog gilt dies für t_- , mit Experimenten, die die Nichtexistenz einer Verbindung beschreiben.

Werden Teilbaumverbindungen gewählt, gibt es neben der Adjazenzmatrix auch eine Matrix, die als Einträge die Anzahl der Kanten zwischen den beiden Teilbäumen der Regionen enthält. Bei der Generierung der Adjazenzmatrix werden automatisch alle Daten gespeichert, die für das schnelle Umschalten zu einer anderen Ansicht nötig sind. Als aktuell ein Netzwerk mit circa 20000 Knoten analysiert werden sollte, hat sich dieses Vorgehen auf Grund des hohen Speicherbedarfs als Nachteil herausgestellt. Um zumindest einfache Analysen mit Netzwerken dieser Größe zu erlauben, wurde eine Option ergänzt, die bewirkt, dass diese zusätzlichen Daten nicht gespeichert werden und nur die Adjazenzmatrix verfügbar ist. Gleichzeitig wird auch die Visualisierung der Matrix deaktiviert, da diese in der aktuellen Implementierung aus Performancegründen relativ viel Speicher benötigt.

Ausgehend von der Adjazenzmatrix lassen sich verschiedene Analysen

durchführen. Detaillierte Informationen über die verschiedenen implementierten Maße finden sich in den Anhängen von den Publikationen [37] und [36]. Die Anwendung von graphentheoretischen Maßen bei der Analyse von neuronalen Netzen wird zum Beispiel von Olaf Sporns in [41] und [40] sowie von Marcus Kaiser in [19] beschrieben. Hier soll nur exemplarisch eine Auswahl der implementierten Analysemöglichkeiten vorgestellt werden.

Die **Distanzmatrix** gibt für ein Paar von Regionen an, wie lang der kürzeste Pfad ist, der von der Ausgangs- zur Zielregion führt. Ein **Pfad** ist dabei eine Folge von Verbindungen, wobei in dieser Folge die Zielregion einer Verbindung immer die Ausgangsregion der folgenden Verbindung ist. Zur Berechnung der Distanzmatrix wurde der Algorithmus von Floyd [10] verwendet. Existiert eine direkte Verbindung von Ausgangs- zur Zielregion, so ist die Distanz Eins. Gelangt man nur über eine oder mehrere Zwischenregionen von der einen zur anderen Region, wird die Distanz größer. Ein Wert Null wird in *neuroVIISAS* verwendet, um zu beschreiben, dass es keinen Pfad von der Ausgangs- zur Zielregion gibt. Die Diagonale der Distanzmatrix gibt in *neuroVIISAS* die Länge des kürzesten Pfades von einer Region zu sich selbst an. Ist die Distanz zwischen zwei Regionen größer als Eins, so kann es mehrere verschiedene Pfade dieser Länge geben, von denen einer mit Hilfe eines Tooltips angezeigt wird. Die Distanzmatrix ist Grundlage für eine Reihe von globalen und lokalen Netzwerkparametern.

Die **Communicabilitymatrix** [7] ist eine Erweiterung der Distanzmatrix, die nicht nur kürzeste Pfade zwischen zwei Regionen sucht, sondern alle Pfade von einer Ausgangs- zu einer Zielregion gewichtet in Abhängigkeit von der Länge des Pfades aufsummiert. Die Communicabilitymatrix G ist dabei wie folgt definiert:

$$G = \sum_{k=0}^{\infty} \frac{1}{k!} A^k.$$

Dabei ist A die Adjazenzmatrix und A^0 die Einheitsmatrix. Die Matrix A^k enthält an einer Stelle (i, j) die Anzahl Pfade der Länge k vom Knoten i zum Knoten j . Diese Anzahl Pfade geht in die Summe gewichtet mit dem Faktor $\frac{1}{k!}$ ein, der für steigende k schnell gegen Null konvergiert und die Konvergenz der Reihe garantiert. Die obere Schranke für die Anzahl Pfade der Länge k wird bei einer voll besetzten $(n \times n)$ -Adjazenzmatrix erreicht. Dort gilt, dass

A^k für $k > 0$ an jeder Stelle den Wert n^{k-1} hat. Für die Komponenten einer beliebigen Matrix G gilt dann:

$$G_{ij} \leq 1 + \sum_{k=1}^{\infty} \frac{n^{k-1}}{k!} = 1 + \frac{e^n - 1}{n}.$$

Die Communicabilitymatrix könnte ein Maß sein für die Menge der Informationen, die von einer Region zu einer anderen übertragen werden können. Ein Nachteil der obigen Definition ist, dass die Werte der Communicabilitymatrix exponentiell mit der Größe der Matrix steigen. Aus diesem Grund wurde eine Variante \hat{G} entwickelt, deren obere Schranke für die Werte unabhängig von der Größe der Matrix ist:

$$\hat{G} = \sum_{k=1}^{\infty} \frac{1}{(2n)^{k-1}} A^k.$$

Analog zu Abschätzung für G gilt:

$$\hat{G}_{ij} \leq \sum_{k=1}^{\infty} \frac{n^{k-1}}{(2n)^{k-1}} = \sum_{k=1}^{\infty} \frac{1}{2^{k-1}} = \sum_{k=0}^{\infty} \frac{1}{2^k} = 2.$$

Inhaltlich bedeutet dies, dass eine direkte Verbindung von Knoten i zu Knoten j mit dem Wert Eins in \hat{G}_{ij} eingeht. Die Anzahl aller Pfade der Länge Zwei geht dann nur noch mit dem Anteil $\frac{1}{(2n)}$ in die Summe ein. Im Maximalfall ist dies $\frac{1}{2}$, falls alle möglichen Pfade dieser Länge existieren. Im Vergleich zur Matrix G werden in \hat{G} längere Pfade geringer gewichtet.

Eine andere Erweiterung der Distanzmatrix bezieht neben den graphentheoretischen Distanzen auch die räumlichen Distanzen zwischen den Regionen sowie die Kantengewichte in die Distanzberechnung mit ein. Ausgangspunkt für die Definition dieser **Erweiterten Distanzmatrix** war die Vorstellung eines Informationstransportes über die Kanten eines Netzwerkes. Hohe räumliche Distanzen verzögern den Transport ebenso wie Zwischenknoten. Hohe Kantengewichte bedeuten hier einen hohen Informationsdurchsatz. Ausgehend von einer Matrix mit den neu zu definierenden Abständen der direkt verbundenen Regionen, wird wie die Distanzmatrix auch die Erweiterte Distanzmatrix mit dem Algorithmus von Floyd berechnet. Zur Definition des Abstands \hat{d} zweier direkt verbundener Regionen stehen zwei Formeln mit frei wählbaren Parametern zur Verfügung.

$$\hat{d}_1 = F \frac{\vec{d}}{g} + S$$

$$\hat{d}_2 = F_1 \cdot \vec{d} + \frac{F_2}{g} + S$$

Dabei ist \vec{d} die räumliche Distanz der beiden Regionen und g das Gewicht der Verbindung. Der Parameter S liefert den Einfluss des graphentheoretischen Abstands zwischen den Regionen. Eine Pfadlänge von n geht in das neue Maß dann additiv mit $n \cdot S$ ein. Die Parameter S , F , F_1 und F_2 sind als positive reelle Zahlen frei wählbar und beeinflussen die Gewichtung der unterschiedlichen Einflussgrößen.

Die **Connectivity-Matching Matrizen** sind ein Maß für die Ähnlichkeit der Verbindungsstruktur zweier Regionen. Die Implementierung wurde übernommen aus der *Brain Connectivity Toolbox* [33]. Dabei werden drei Matrizen unterschieden. Die Connectivity-Matching-Out Matrix vergleicht nur abgehende, die Connectivity-Matching-In Matrix nur ankommende und die Connectivity-Matching-All Matrix sowohl abgehende als auch ankommende Verbindungen. Eigenkanten werden dabei jeweils nicht berücksichtigt. Je größer der Anteil gemeinsamer Nachbarn zweier Knoten ist, desto größer ist deren Connectivity-Matching-Index.

Neben den Matrizen, die Beziehungen zwischen Regionen angeben, gibt es auch die **Joint Degree Distribution Matrix**, die an der Position (i, j) die Anzahl der Knoten angibt, die i abgehende Kanten und j ankommende Kanten besitzen. Diese Matrix gibt Aufschluss darüber, wie die ankommenden und abgehenden Kanten verteilt sind. Hat die Matrix hauptsächlich Einträge nahe der Diagonalen, so haben die Knoten in etwa so viele abgehende wie ankommende Kanten. Je weiter die Einträge von der Diagonalen entfernt sind, um so größer sind die Unterschiede zwischen dem Input und dem Output eines Knotens.

Um die Bedeutung von einzelnen Kanten für das Netzwerk zu bestimmen, wurde die **Edge Significance** definiert. Das Kriterium für die Wichtigkeit ist bei diesem Maß der Einfluss einer Kante auf die Pfadlängen im Netzwerk. Dazu wird die Definition der **Closeness**

$$C = \frac{1}{n^2} \sum_{d_{ij} > 0} \frac{1}{d_{ij}},$$

mit den positiven Einträge d_{ij} der Distanzmatrix, verwendet. Dabei werden hier auch die Distanzen d_{ii} berücksichtigt, also Pfade von einem Knoten zu

sich selbst. Die Edge Significance einer Kante ist dann der relative Abfall der Closeness nach einem Entfernen dieser Kante, in Prozent. Ein Edge Significance Wert von 10 für eine Kante bedeutet also, dass bei einem Entfernen dieser Kante die Closeness des Netzwerk um 10 Prozent sinkt. Die Werte der Edge Significance für jede Kante werden in einer Matrix, der **Edge Significance Matrix** dargestellt. Als **Edge Vulnerability** eines Netzwerks wurde der Mittelwert der Edge Significance Werte aller Kanten definiert.

3.4.4 Globale Netzwerkparameter

Neben den Beziehungen zwischen zwei Regionen oder den Eigenschaften von Kanten, gibt es auch 35 globale Parameter, die Eigenschaften des gesamten Netzwerks aller betrachteten Regionen beschreiben. Die einfachsten globalen Parameter sind zum Beispiel die Anzahl der Verbindungen oder die **Verbindungsichte**, die die Anzahl der Verbindungen ins Verhältnis setzt zur maximal möglichen Anzahl. Der **Durchmesser** eines Netzwerks gibt die maximale kürzeste Pfadlänge zwischen zwei Regionen an. Dieser Wert gibt Auskunft darüber, wie viele Verbindungen eine Information maximal passieren muss. Die mittlere Pfadlänge oder die **Global Efficiency** [22] eines Netzwerkes geben Auskunft darüber, wie groß die Distanzen zwischen den Regionen im Mittel sind, d.h. wie direkt Informationen in einem Netzwerk im Durchschnitt von einem Knoten zu einem anderen gelangen. Die Global Efficiency

$$E = \frac{1}{n(n-1)} \sum_{d_{ij} > 0, i \neq j} \frac{1}{d_{ij}}$$

entspricht dabei fast der im letzten Abschnitt eingeführten Closeness, mit dem Unterschied, dass hier keine Distanzen von einem Knoten zu sich selbst berücksichtigt werden. Die Global Efficiency des Netzwerks ist dabei umso größer, je kleiner die Distanzen zwischen den Regionen sind. Der maximale Wert von Eins bedeutet, dass jeder Knoten direkt von jedem anderen erreichbar ist.

Ein komplexerer Parameter ist die **Modularität**, beschrieben in [28], die dann hohe Werte liefert, wenn die Regionen in Gruppen, sogenannte Module oder Cluster, unterteilt werden können, bei denen die Regionen innerhalb eines Moduls stärker miteinander verbunden sind als Regionen verschiedener Module. Welche Aufteilung zu dem berechneten Wert für die Modularität geführt hat, wird in *neuroVIISAS* nach der Berechnung angezeigt (Abbildung

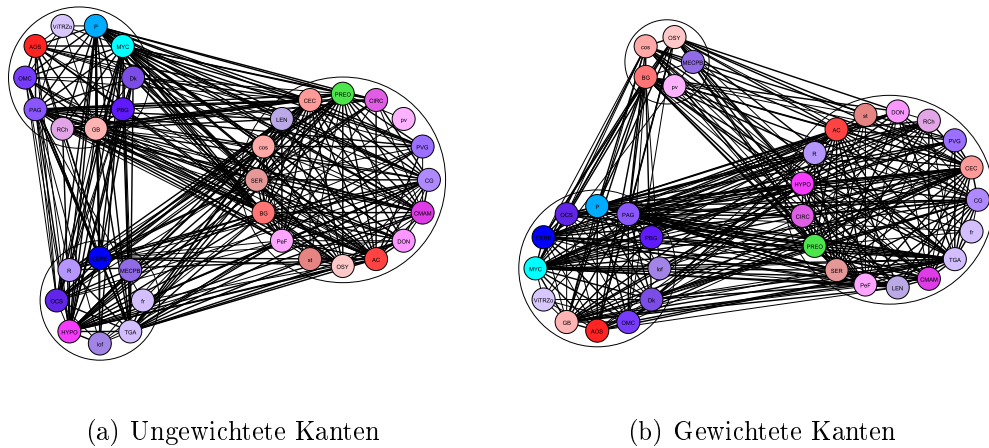


Abbildung 3.4: Regionen in den Modulen der Molularitätsberechnung

3.4a). Neben der ungewichteten Variante wurde eine gewichtete Variante implementiert, die die Stärke der Verbindungen berücksichtigt. Abbildung 3.4b zeigt, dass die Module bei der gewichteten Modularität anders zusammengesetzt sind, als die der ungewichteten Variante. Bei Wahl der gewichteten Modularität können die Gewichte optional durch unterschiedliche Liniendicken visualisiert werden. Neben den Modulen kann auch eine kleine Adjazenzmatrix eingeblendet werden, die die Anzahl der Kanten innerhalb und zwischen den Modulen enthält.

Einige globale Parameter sind Mittelwerte von lokalen Parametern, wie zum Beispiel die **mittlere Subgraph Centrality** [8] die das arithmetische Mittel der Diagonalelemente der Communicabilitymatrix ist.

Um die globalen Parameter besser interpretieren zu können, ist es möglich diese mit den entsprechenden Parametern von Zufallsnetzwerken zu vergleichen. Dafür kann für jedes implementierte Verfahren angegeben werden, wie viele Zufallsnetzwerke dieses Typs generiert werden sollen. Für die Zufallsnetzwerke eines Typs werden dann jeweils die globalen Parameter berechnet und gemittelt ausgegeben. Für das Erdős-Rényi-Zufallsmodell wird immer mindestens ein Zufallsnetzwerk generiert, da dessen Parameter für die Berechnung der Small-Worldness [17] benötigt werden.

Da die Berechnung einiger Parameter, gerade bei großen Netzwerken, sehr aufwändig ist, gibt es die Möglichkeit, für einen Parameter anzugeben, ob er berechnet werden soll oder nicht. Für viele der implementierten globalen Netzwerkparameter steht neben der ungewichteten auch eine gewichtete Variante

zur Verfügung.

3.4.5 Lokale Netzwerkparameter

Die umfangreichste Auswahl an implementierten Analysemethoden (42) steht für die Berechnung der lokalen Netzwerkparameter zur Verfügung, die Eigenschaften einzelner Knoten im Netzwerk beschreiben. Die Definitionen einiger Parameter wurden dabei für gerichtete Graphen angepasst, beziehungsweise erweitert. Wie bei den globalen Parametern ist auch bei vielen lokalen Parametern eine gewichtete Variante implementiert und es ist wählbar, welche Parameter berechnet werden sollen.

Ein einfacher Parameter ist der Grad eines Knotens, der angibt, wie viele Kanten mit diesem Knoten verbunden sind. Dabei kann der Grad separat für ankommende, abgehende oder alle Kanten bestimmt werden. Eine Klasse von Parametern, die stark mit dem Grad eines Knotens korrelieren, sind die sogenannten Zentralitätsmaße. Dazu gehört zum Beispiel die **Betweenness Centrality** eines Knotens v , bei der für alle Paare von anderen Knoten $a \neq b$ der Anteil der kürzesten Pfade von a nach b berechnet wird, der über v geht. Die Betweenness Centrality des Knotens ist dann der Mittelwert aller dieser Anteile. Der theoretische Maximalwert von Eins für einen Knoten würde bedeuten, dass dieser Knoten auf allen kürzesten Pfaden liegt. Dann wäre er Zentrum eines Sterns. Je höher die Betweenness Centrality eines Knotens, umso wichtiger ist er für die Verbundenheit im Netzwerk. Ein Entfernen eines solchen Knotens würde die Wege zwischen den übrigen Knoten eher verlängern als das Entfernen eines Knotens mit geringerer Betweenness Centrality. Der **Clusterkoeffizient** gibt an, wie stark die Nachbarn eines Knotens untereinander verbunden sind. Ist jeder der Nachbarn mit jedem anderen verbunden, bilden sie zusammen mit dem betrachteten Knoten eine sogenannte Clique, d.h. eine Menge von Knoten, in der jeder Knoten mit jedem anderen verbunden ist. Der Clusterkoeffizient für den betrachteten Knoten hätte dann den Maximalwert Eins. Sind die Nachbarn untereinander gar nicht verbunden, ist der Clusterkoeffizient Null. Für die gerichteten Graphen in *neuroVIISAS* wurde der Clusterkoeffizient in mehrere unterschiedliche Clusterkoeffizienten aufgespalten. Der Clusterkoeffizient $\text{Clu}C_{\text{out}}$ betrachtet nur OUT-Nachbarn eines Knotens, d.h. Nachbarn, wo die Verbindung von dem betrachteten Knoten zum Nachbarn führt. Analog ist es

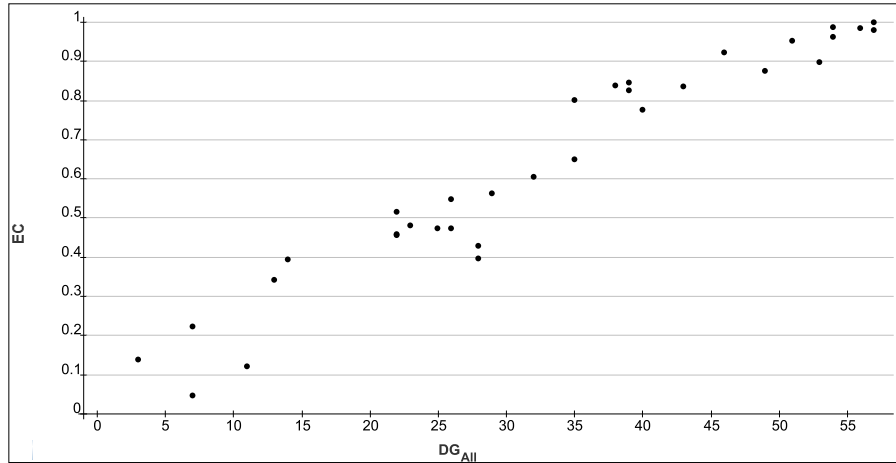


Abbildung 3.5: Grad und Eigenvektorzentralität der Knoten

beim $\text{Clu}C_{\text{in}}$ bei denen die IN-Nachbarn eines Knotens betrachtet werden, d.h. solche, von denen eine Verbindung zu dem betrachteten Knoten führt. $\text{Clu}C_{\text{all}}$ betrachtet dann sowohl OUT- als auch IN-Nachbarn. Beim Grad der Verbundenheit zwischen den Nachbarn wird dann der Anteil der existierenden Kanten an den maximal möglichen berechnet, wobei zwischen jedem Paar von Knoten zwei gerichtete Kanten möglich sind. Eine vierte implementierte Variante $\text{Clu}C_{\text{Triag}}$ eines Clusterkoeffizienten ist in [9] beschrieben.

Die lokalen Parameter für die Auswahl von Regionen werden nach der Berechnung in einer Tabelle dargestellt, die bezüglich jedes Parameters sortierbar ist. Es ist möglich, eine Matrix mit den Korrelationskoeffizienten aller Paare von Parametern ausrechnen zu lassen. Die Korrelationskoeffizienten geben den Grad der linearen Abhängigkeit der Parameter an. Werte die betragsmäßig nahe Eins sind bedeuten eine starke Abhängigkeit, wobei positive Werte bedeuten, dass wenn ein Parameter steigt, tendenziell auch der andere wächst. Die Zentralitätsmaße korrelieren in den berechneten Beispielen häufig mit Korrelationskoeffizienten $> 0,9$. In Abbildung 3.5 ist der Grad der Knoten zusammen mit der Eigenvektorzentralität dargestellt. Diese beiden Parameter haben in dem Beispiel einen Korrelationskoeffizienten von rund $-0,97$. Im negativen Fall fällt der eine Parameter tendenziell, wenn der andere wächst. In Abbildung 3.6 ist der Grad der Knoten zusammen mit dem mittleren Grad der Nachbarn eines Knotens dargestellt. Der Korrelationskoeffizient ist in diesem Fall rund $0,95$.

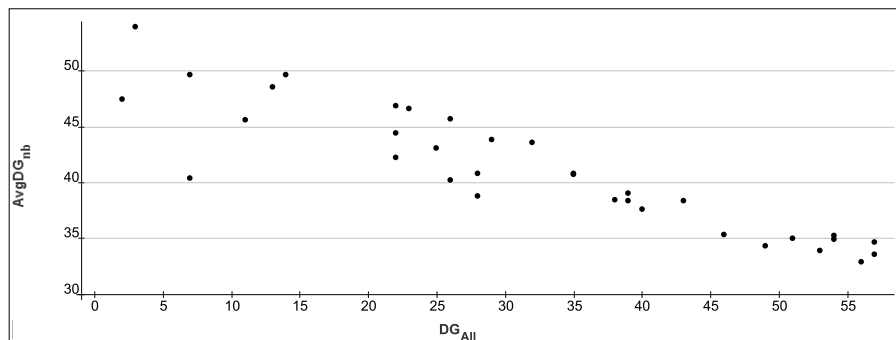


Abbildung 3.6: Grad des Knotens und mittlerer Grad der Nachbarknoten

3.4.6 Hierarchische Clusteranalyse

Das Ziel der Clusteranalyse ist es, ähnliche Regionen zu Gruppen, den sogenannten Clustern, zusammenzufassen. Eine hierarchische Clusteranalyse erzeugt darüber hinaus eine Hierarchie von Clustern, deren Blätter die Regionen sind. Dafür muss zunächst definiert werden, was die Ähnlichkeit von Regionen ausmachen soll. In *neuroVIISAS* sind dafür verschiedene Kriterien auswählbar.

Bei der **Modularitäts-Clusterung** wird die Clusterung aus der Modularitätsberechnung [28] berechnet, bei der Regionen so zu Clustern zusammengefasst werden, dass sie stärker mit Regionen desselben Clusters verbunden sind, als mit Regionen anderer Cluster. Diese Zerlegung bildet die erste Ebene der Clusterhierarchie. Enthält ein Cluster mehr als zwei Regionen, wird dieser Cluster mit dem selben Verfahren rekursiv so lange weiter zerlegt, bis keine neuen Cluster mehr entstehen. In Abbildung 3.7b ist erkennbar, dass in der Adjazenzmatrix nach der Modularitäts-Clusterung die Kantendichte in der Nähe der Diagonalen im Vergleich zur Ausgangsmatrix zugenommen hat. Das bedeutet, dass die Regionen jetzt durch die Clusterung so sortiert sind, dass sie mit benachbarten Regionen stärker verbunden sind.

Ein weiteres implementiertes Kriterium für die Ähnlichkeit der Regionen ist vergleichbar mit den Connectivity-Matching-Matrizen, die ein Maß für die Ähnlichkeiten der Verbindungsstruktur zweier Regionen sind. Bei der **Connectivity-Matching-Out-Clusterung** bilden die Zeilen der Adjazenzmatrix, das heißt die abgehenden Verbindungen eines Knotens, einen n -dimensionalen Vektor, wobei n die Anzahl der betrachteten Regionen ist. Bei der **Connectivity-Matching-In-Clusterung** sind es die Spalten

der Adjazenzmatrix, das heißt, die ankommenden Verbindungen und bei der **Connectivity-Matching-All-Clusterung** werden beide zu einem $2n$ -dimensionalen Vektor zusammengefasst. Eigenkanten werden dabei jeweils nicht berücksichtigt. Der Abstand zweier Regionen ist dann der euklidische Abstand der beiden Vektoren. Jede Region ist zu Beginn ein eigener Cluster. Dann werden schrittweise die zwei Cluster miteinander verschmolzen, die den geringsten Abstand haben. Gibt es mehr als zwei Cluster, die alle voneinander gleichweit entfernt sind, zum Beispiel weil sie identische Koordinaten haben, können auch in einem Schritt mehr als zwei Cluster zusammengefasst werden. Für das Verschmelzen von Clustern, die mehr als eine Region enthalten, stehen zwei wählbare Verfahren zur Verfügung. Die eine Methode sucht die kleinsten Abstände zweier Regionen der beiden Cluster und die andere vergleicht die Abstände bezüglich der Schwerpunkte der Cluster. In Abbildung 3.7c wird die Auswirkung der Clusterung dadurch deutlich, dass benachbarte Regionen relativ ähnliche Zeilen und Spalten in der Adjazenzmatrix besitzen.

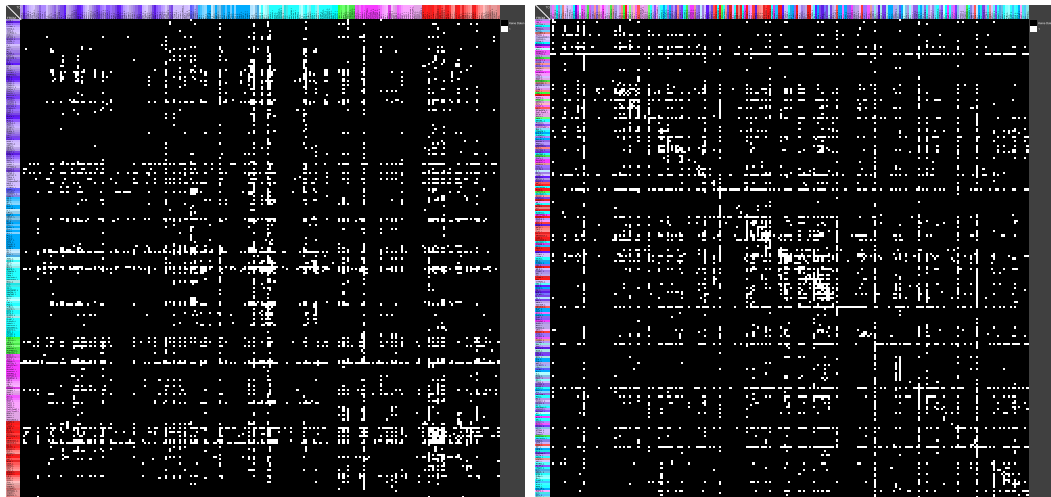
Bei der **Girvan-Newman-Clusterung** [12] werden aus der Gesamtheit aller Kanten sukzessive die Kanten mit der höchsten *Edge Betweenness* entfernt. Die *Edge Betweenness* ist ein Maß für den Anteil kürzester Pfade, die diese Kante enthalten. Dazu wird für jedes Paar von Regionen der Anteil von kürzesten Pfaden zwischen diesen bestimmt, die diese Kante passieren. Die aufsummierten Anteile entsprechen dann der *Edge Betweenness* dieser Kante. Der Graph zerfällt durch das Entfernen der Kanten nach und nach in einzelne Zusammenhangskomponenten, wobei bei jedem Zerfall zwei neue Cluster entstehen. Isolierte Knoten werden dabei dem nächsthöheren Cluster zugeordnet. Die Auswirkung der Clusterung ist beispielhaft in der Adjazenzmatrix in Abbildung 3.7d dargestellt.

Zusätzlich zu den obigen Verfahren wurde auch die **Spektralclusterung** implementiert. Das *Spektrum* einer Matrix bezeichnet die Menge aller Eigenwerte dieser Matrix, welche in dem Verfahren eine Rolle spielen. Voraussetzung für eine Spektralclusterung ist eine symmetrische Matrix mit den paarweisen Ähnlichkeiten der zu clusternden Regionen. Wählbare Ähnlichkeiten in *neuro-VIISAS* sind die Connectivity-Matching-Matrizen und zwei Ähnlichkeiten, die auf der direkten beziehungsweise der indirekten Verbundenheit der Regionen basieren. Mit der Adjazenzmatrix $(a_{ij})_{i,j=1}^n$ ist bei der direkten Verbunden-

heit die Ähnlichkeit von zwei Regionen i und j definiert als $\frac{1}{2}(a_{ij} + a_{ji})$. Bei der indirekten Verbundenheit mit der Distanzmatrix $(d_{ij})_{i,j=1}^n$ berechnet sich die definierte Ähnlichkeit aus $\frac{1}{2}(\frac{1}{d_{ij}} + \frac{1}{d_{ji}})$. Die Berechnung der Cluster erfolgt dann in mehreren Schritten. Zur Ähnlichkeitsmatrix wird die Laplacematrix L berechnet von der dann die Eigenwerte und Eigenvektoren bestimmt werden. Die Zeilen der Eigenvektoren zu den k kleinsten Eigenwerten bilden dann die Koordinaten der Regionen in einem k -dimensionalen Raum. Der Parameter k ist dabei die angestrebte Anzahl von Clustern, die vorher angegeben werden muss. Mit Hilfe eines k -Means-Algorithmus werden die Regionen dann k Clustern zugeordnet.

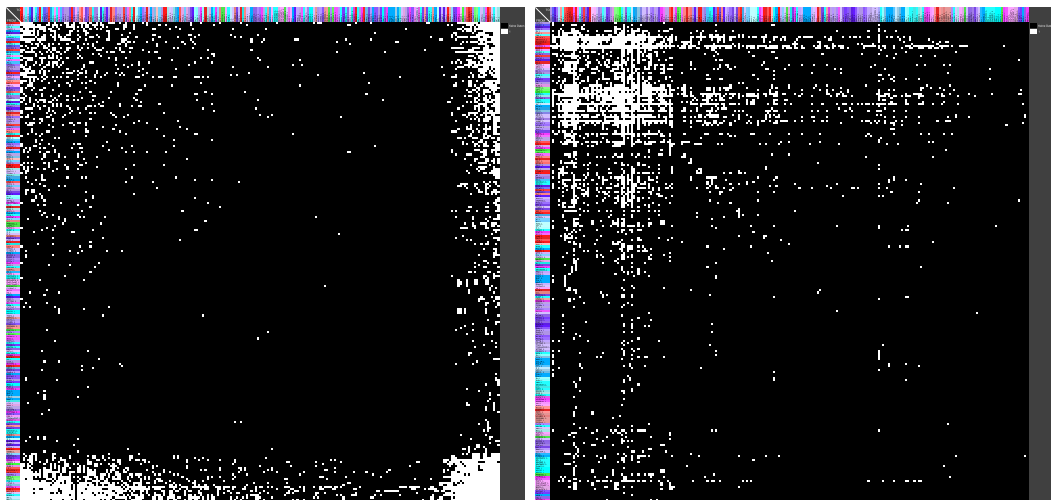
3.4.7 Motivsuche

Ein Motiv ist hier ein kleines zusammenhängendes Netzwerk mit einer bestimmten Verbindungsstruktur, dass man innerhalb eines großen Netzwerks sucht. Dazu kann in *neuroVIISAS* mit Hilfe einer graphischen Oberfläche ein Motiv gezeichnet und einer Liste von zu suchenden Motiven hinzugefügt werden. Darüber hinaus ist es auch möglich, automatisch alle möglichen Motive mit drei (13 Möglichkeiten), vier (199) oder fünf (9364) Knoten ohne Eigenkanten berechnen zu lassen. Zusätzlich können auch alle möglichen Motive mit Eigenkanten erzeugt werden. Die Liste der generierten Motive kann gespeichert und wieder geladen werden. Alle in der Liste ausgewählten Motive, können dann in dem Netzwerk der gewählten Gebiete gesucht werden. Dabei wird ermittelt, wie oft ein Motiv in dem Netzwerk vorkommt, aus welchen Regionen jedes dieser Vorkommen besteht und wie häufig jede Region in diesem Motiv vorkommt. Zusätzlich können die Motive auch in allen implementierten Zufallsnetzwerken gesucht werden, um deren Häufigkeiten innerhalb des Originalnetzwerks mit denen in zufällig erzeugten Netzwerken zu vergleichen. Dazu kann für ein Zufallsmodell die Anzahl der zu erzeugenden Zufallsnetzwerke ausgewählt werden, für die dann jeweils die Häufigkeiten der Motive bestimmt werden. Nach der Motivsuche im Original- und den Zufallsnetzwerken werden die Häufigkeiten dann in einem Diagramm (Abbildung 3.8) dargestellt. Der blaue Punkt über einem Motiv ist dabei die Häufigkeit des Motivs im Originalgraphen und die kleinen schwarzen Punkte sind jeweils die Häufigkeiten in einem der Zufallsnetze. Die rote Fläche ist der Bereich plus und minus der



(a) Adjazenzmatrix vor der Clusterung

(b) Modularitäts-Clusterung



(c) Connectivity-Matching-All-Clusterung

(d) Girvan-Newman-Clusterung

Abbildung 3.7: Vergleich der Adjazenzmatrizen vor und nach einer Clusteranalyse

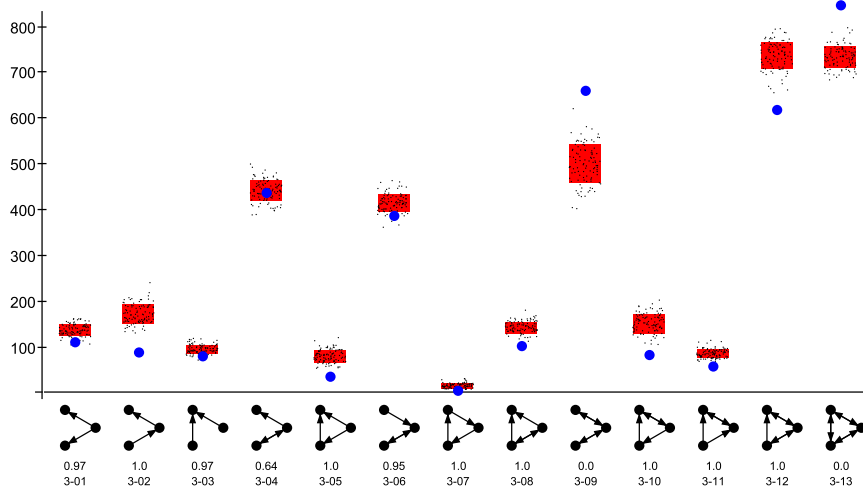


Abbildung 3.8: Häufigkeiten aller Dreiermotive im Originalnetzwerk (blaue Punkte) und in 100 Rewiring-Zufallsnetzwerken (schwarze Punkte)

Standardabweichung um die mittlere Häufigkeit bei den Zufallsnetzwerken. Neben dem Diagramm werden die berechneten Häufigkeiten auch in einer Tabelle dargestellt, die spaltenweise sortierbar ist und neben den Häufigkeiten der Motive noch zusätzliche Informationen enthält. So ist die mittlere Häufigkeit μ und die Standardabweichung σ in den Zufallsnetzwerken angegeben, sowie der z -Wert $z = \frac{F - \mu}{\sigma}$ mit der Häufigkeit F eines Motivs im Originalgraphen und der p -Wert, der den Anteil der Zufallsgraphen angibt, in denen das Motiv häufiger vorkommt als im Originalgraphen.

3.4.8 Pfadsuche

Neben der Pfadsuche in einem Netzwerk von ausgewählten Knoten, ist es auch möglich, Pfade von Verbindungen in der gesamten Hierarchie zu suchen. Dazu muss man die Start- und Zielregion des Pfades angeben und optional auch eine Folge von Zwischenregionen, die in der gegebenen Reihenfolge durchlaufen werden sollen. Die Möglichkeiten der Pfadsuche werden in [36] ausgeschöpft, um zu zeigen, wie häufig es neben den konventionellen Pfaden, wie etwa der direkten oder der indirekten Basalganglienschleife, alternative Pfade gibt. Existiert ein Pfad für die Auswahl von Regionen, werden alle kürzesten Pfade in einer Tabelle ausgegeben. Die Tabelle enthält neben den gefundenen Pfaden auch die Information, ob ein Pfad reziprok ist, das mittlere Gewicht der betei-

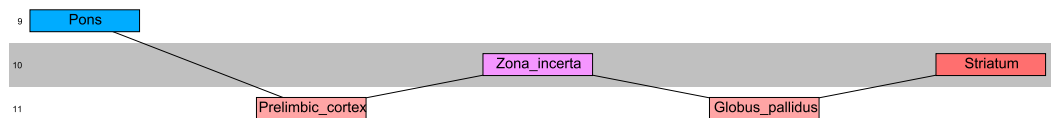


Abbildung 3.9: Ergebnis einer Pfadsuche von Pons über Zona incerta zum Striatum

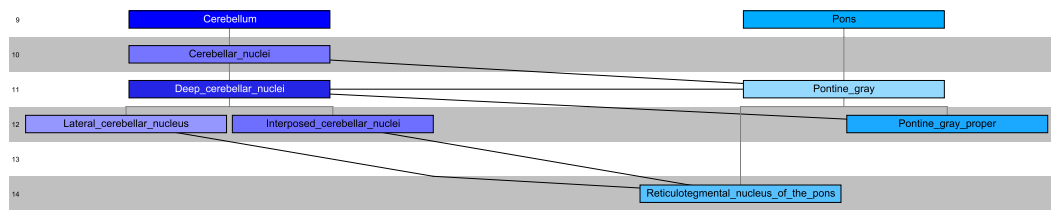


Abbildung 3.10: Ergebnis einer Pfadsuche vom Cerebellum nach Pons mit Teilbaumkanten

ligten Kanten, die mittlere Anzahl Publikationen pro Verbindung, die Anzahl der überwundenen Level in der Hierarchie, die durchschnittliche Validität der Kanten und, wenn verfügbar, eine räumliche Länge der Pfade. Die Tabelle ist exportierbar und lässt sich nach allen Spalten sortieren. Zusätzlich kann man sich die Pfade auch graphisch darstellen lassen (Abbildung 3.9), wobei reziproke Kanten hervorgehoben werden können und es möglich ist, sich an den Verbindungen die Anzahl von Publikationen anzeigen zu lassen, in denen die zugehörige Verbindung beschrieben wurde. Die grauen und weißen Streifen in der Visualisierung geben die Schichten in der Hierarchie der Regionen an. Sind Zwischenregionen ausgewählt, werden nur kürzeste Pfade mit expliziten Verbindungen gesucht. Sind keine Zwischenregionen gewählt, ist es darüber hinaus auch möglich alle Pfade bis zu einer gegebenen maximalen Länge bestimmen zu lassen. Neben den expliziten Verbindungen, können dann auch Pfade zwischen den Teilbäumen der Regionen gesucht werden (Abbildung 3.10). Nach einer Suche kann ausgegeben werden, wie oft eine Regionen beziehungsweise eine Verbindung in den gefundenen Pfaden vorkommt.

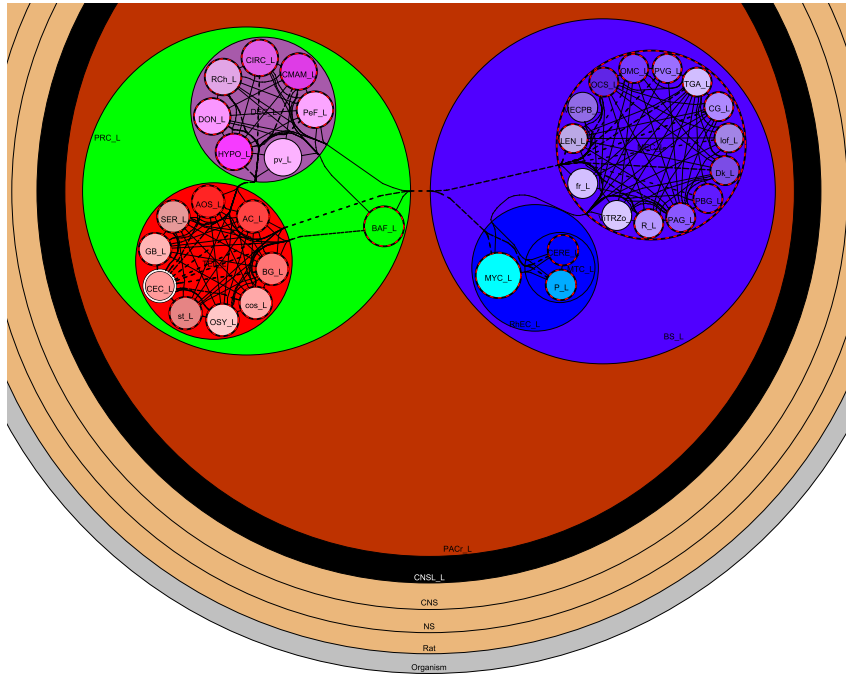


Abbildung 3.11: Kreishierarchie mit Verbindungen

3.5 Visualisierungen

Die Daten sowie die Analyseergebnisse können in *neuroVIISAS* auf verschiedene Arten visualisiert werden. Es gibt Baumdarstellungen für die Hierarchie der Regionen, eine Vielzahl von Tabellen für verschiedene strukturierte Daten oder Matrizen für die Darstellung von Verbindungsinformationen oder Zusammenhängen zwischen Regionen. Einige der aufwändigeren Visualisierungen sollen im Folgenden vorgestellt werden.

3.5.1 Kreishierarchie

Eine Visualisierung, die die Hierarchiestruktur der Regionen zusammen mit den Verbindungen abbildet, ist die Darstellung in verschachtelten Kreisen. Abbildung 3.11 zeigt die linke Hemisphäre der Ratte in der Kreishierarchiedarstellung auf einem relativ groben Level. Die Rechte Seite wurde in dieser Abbildung ausgeblendet. Jede Region ist als farbiger Kreis dargestellt und Unterregionen werden innerhalb des Kreises der Überregion platziert. Die Größenverhältnisse der Kreise werden dabei so gewählt, dass Kreise mit mehr sichtbaren Unterregionen größer gezeichnet werden als solche, mit weniger oder ohne

sichtbare Unterregionen. Diese Darstellung ist nicht nur eine statische Abbildung der gewählten Regionen, sondern kann auch zur Navigation innerhalb der Hierarchie verwendet werden. Ein Doppelklick auf einen Kreis zeigt dessen Unterregionen an oder blendet diese aus. Regionen können entfernt oder gezielt, über eine Suchfunktion, sichtbar gemacht werden.

Optional können zusätzlich die Teilbaumverbindungen zwischen den Regionen angezeigt werden. Der Verlauf der Verbindungslinien wurde, angelehnt an physikalische Modelle, so berechnet, dass eine Kante dem Weg eines geladenen Teilchens in einem Kraftfeld folgt, wobei der Zielpunkt der Verbindung eine Anziehungskraft und alle Kreise eine abstoßende Kraft ausüben. Verlassen Kanten einen Kreis, so werden sie an einem Port gebündelt und dann als eine Kante weitergeführt, um die Übersichtlichkeit zu erhöhen. In Abbildung 3.11 wurde der zerebrale Kortex (CEC_L) ausgewählt, dessen zugehöriger Kreis deshalb mit einem weißen Rand versehen ist. Die mit den gewählten Regionen verbundenen Regionen und die zugehörigen Kanten werden durch eine gestrichelte Darstellung hervorgehoben. Dabei können die Hervorhebungen individuell nur für den Input, den Output oder für Beide aktiviert werden. Zusätzlich ist auch eine Animation der Signalwege möglich, so dass die Richtung des Signalflusses sichtbar wird.

3.5.2 Graphenvisualisierung

Für die Darstellung des Netzwerks von Regionen mit ihren Verbindungen gibt es die Möglichkeit einer Visualisierung als Graph mit Knoten und Kanten. Dafür werden die Java-Bibliotheken von *yFiles* der Firma *yWorks* verwendet, die Layouts für Graphen berechnen. Dabei stehen diverse Algorithmen für das Knoten- und Kantenlayout zur Verfügung, die in *neuroVIISAS* gewählt werden können, um möglichst übersichtliche Ergebnisse zu erzielen. Es ist möglich Überregionen anzuzeigen (Abbildung 3.12) oder bei einer Darstellung von Regionen aus beiden Hemisphäre einen bezüglich der Lateralität symmetrischen Graphen zu erzeugen (Abbildung 3.13).

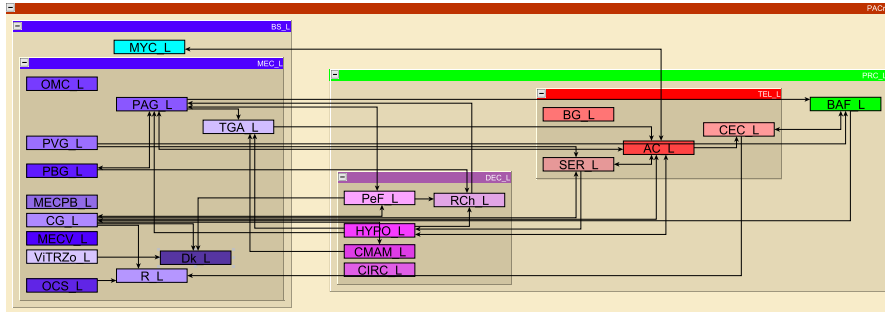


Abbildung 3.12: Hierarchic group layout von yFiles mit Darstellung von Überregionen

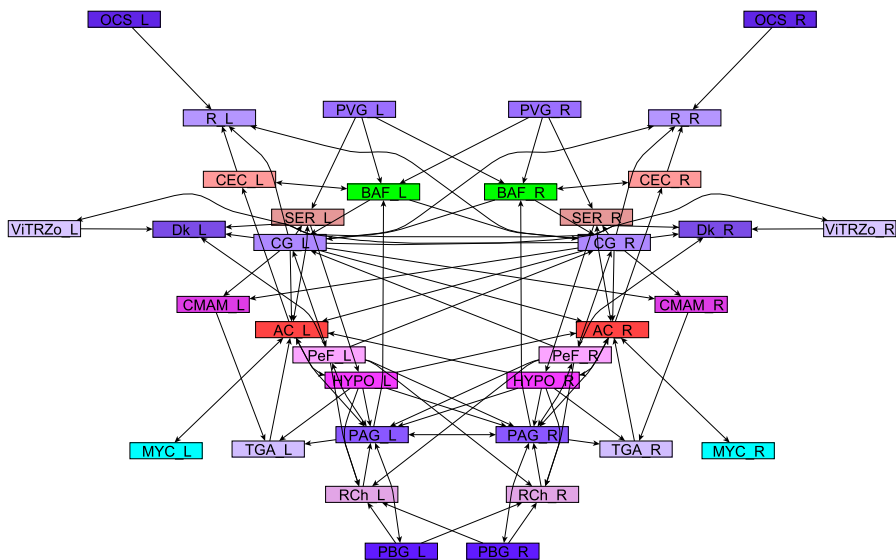


Abbildung 3.13: Symmetrisches Smart organic layout von yFiles

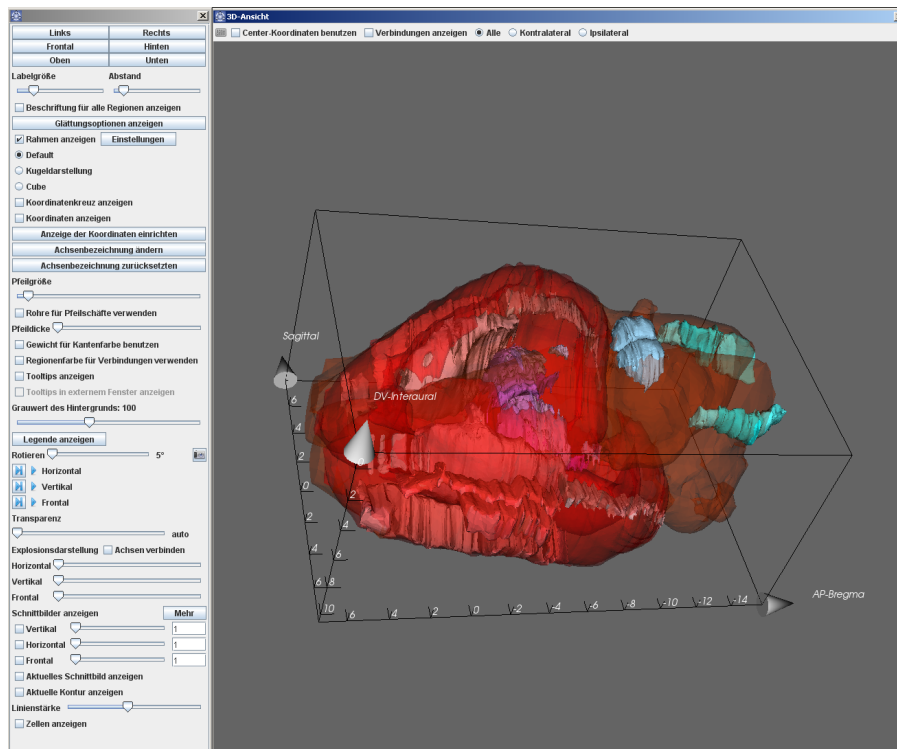


Abbildung 3.14: 3D-Ansicht von Hirnregionen in *neuroVIISAS*

3.5.3 3D-Visualisierung

Aus den Konturdaten der Regionen können in *neuroVIISAS* unter anderem mit Hilfe des Marching-Cube-Algorithmus [24] 3D-Rekonstruktionen der Regionen erzeugt werden (Abbildung 3.14). Dazu werden benachbarte Schnitte mit Konturen in Quadrate zerlegt, für deren Eckpunkte ermittelt wird, ob sie im Inneren oder außerhalb der Region liegen. Aus benachbarten Quadraten der zwei Schnittbilder werden dann Würfel gebildet und an den Marching-Cube-Algorithmus übergeben, der eine Triangulation der Oberfläche der Region zwischen den beiden Schnittbildern berechnet. Je kleiner die Quadrate gewählt werden, desto detaillierter gibt die 3D-Rekonstruktion die Konturen innerhalb der Bilder wieder. Größere Quadrate führen zu einer größeren Interpolation. In *neuroVIISAS* kann angegeben werden, wie oft ein Gebiet unterteilt werden soll und wie stark die Oberflächen nach der Berechnung geglättet werden. Für den Marching-Cube-Algorithmus sowie die 3D-Visualisierungen werden die Bibliotheken des *Visualization Toolkit (VTK)* verwendet, die dem Programm für die verschiedenen Plattformen beiliegen. Alternativ kann auch

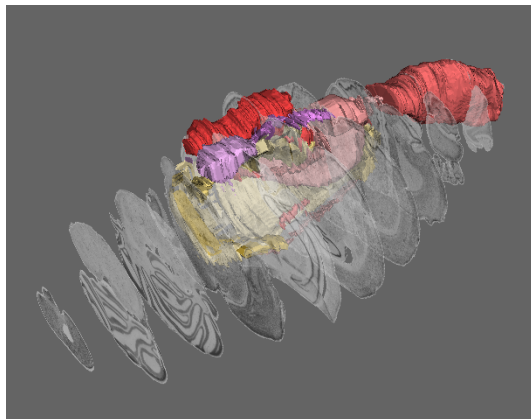


Abbildung 3.15: Eingblendete Schnittbilder in der 3D-Ansicht

eine in VTK verfügbare 3D Delaunay Triangulation gewählt werden, die die Oberfläche einer Region aus den Punkten der Konturen generiert. Zusätzlich zur 3D-Darstellung der Regionen, gibt es eine Vielzahl von Funktionen, die die Visualisierung ergänzen. Regionen können zum Beispiel transparent dargestellt werden, was insbesondere dann sinnvoll ist, wenn kleine Strukturen innerhalb von Größeren liegen. Die Explosionsdarstellung der Regionen sorgt für einen bessere Sicht auf innen liegende Strukturen und schafft Raum, wenn die Verbindungen zwischen den Regionen angezeigt werden sollen. Es kann auch von der konkreten Form der Regionen abstrahiert werden und nur eine einfache Kugel oder Quaderansicht der Regionen gewählt werden. Beim Clipping wird ein Teil der Regionen entlang einer Ebene abgeschnitten, um von einer Seite ins Innere zu schauen. Ist eine Schnittbildserie verfügbar, so können einzelne Schnittbilder oder mehrere in regelmäßigen Abständen eingeblendet werden (Abbildung 3.15). Es kann auch ein Koordinatensystem mit oder ohne Gitterlinien eingeblendet werden. Die Regionen werden in ihrer spezifischen Farbe dargestellt und können mit einem Label beschriftet werden. Ein optionaler Tooltip kann Informationen über das aktuell unter dem Mauszeiger befindliche Gebiet anzeigen. Es ist möglich, die dargestellten Regionen aus einer Auswahl von festen Blickwinkeln anzeigen zu lassen und sie in wählbaren Gradschritten um die drei Koordinatenachsen, aus der aktuellen Blickrichtung heraus, zu rotieren. Eine volle Rotation kann dabei unter Windows auch als GIF-Animation abgespeichert werden.

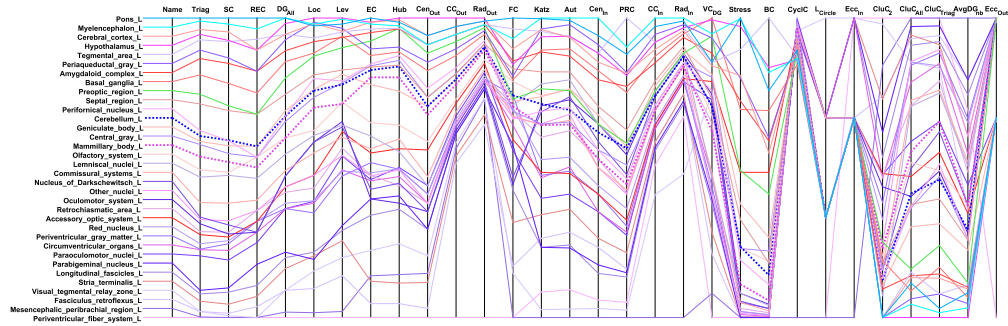


Abbildung 3.16: Lokale Parameter in parallelen Koordinaten

3.5.4 Parametervisualisierung und Dimensionsreduktion

Um Regionen bezüglich ihrer lokalen Parameter miteinander vergleichen zu können, werden sie zusammen in einer Tabelle dargestellt, bei der die Regionen nach den einzelnen Parameterwerten sortiert werden können. Darüber hinaus gibt es noch eine Darstellung in parallelen Koordinaten (Abbildung 3.16), wo es für jeden Parameter eine senkrechte Koordinatenachse gibt und für jede Region eine farbige Linie, die jede Koordinatenachse an der dem Parameterwert der Region entsprechenden Stelle berührt. Um Trends und Ähnlichkeiten von Parametern besser erkennen zu können, ist es möglich die Parameter entsprechend ihrer Ähnlichkeit zu sortieren. Dazu werden für alle Paare von Parametern die Korrelationskoeffizienten bezüglich der aktuellen Werte berechnet und die Parameter dann so sortiert, dass die beiden mit der höchsten Korrelation am Anfang stehen und im Folgenden immer derjenige aus den noch nicht platzierten Parametern auf den letzten folgt, mit dem dieser die höchste Korrelation aufweist. Das bewirkt, dass die Linien sich weniger kreuzen und man besser erkennen kann, welche Regionen ähnliche Parameterwerte haben (ähnlicher Verlauf der Linien) und welche Parameter welchen anderen sehr ähnlich sind (Linien zwischen den Parameterachsen verlaufen annähernd parallel).

Eine Visualisierungsmöglichkeit, die dazu dient Ähnlichkeiten und Unterschiede zwischen Regionen bezüglich ihrer lokalen Parameter darzustellen, ist die Platzierung der Regionen in einem Koordinatensystem. Dabei sollen bezüglich der Parameterwerte einander ähnliche Regionen nah beieinander liegen und unähnliche weiter voneinander entfernt. Jeder der n Parameter, den man dabei berücksichtigen möchte, bildet dabei eine Koordinatenachse in einem n -dimensionalen Koordinatensystem. Da wir in einem Bild über die Position

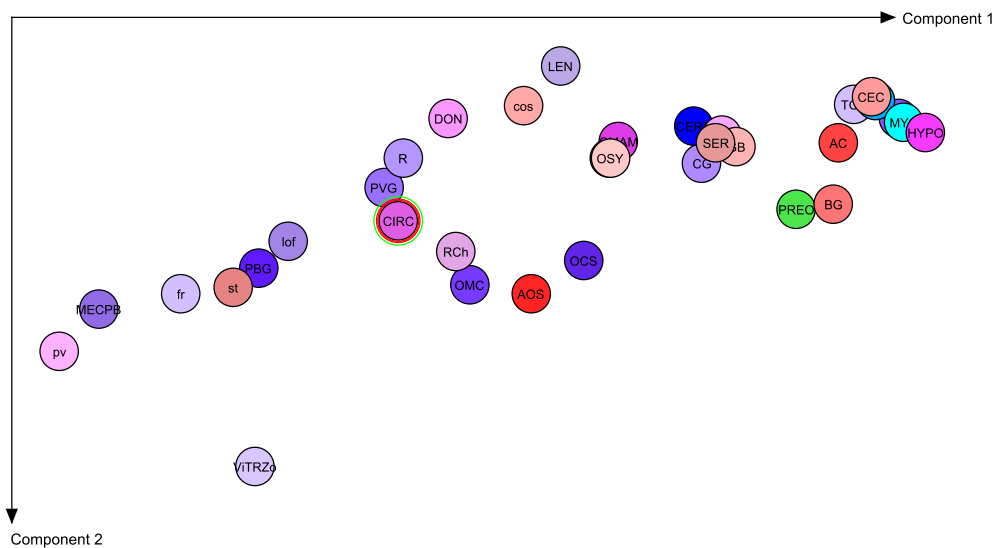


Abbildung 3.17: Regionen aus Abbildung 3.16 im Koordinatensystem der lokalen Parameter bezüglich der zwei Hauptkomponenten

nur zwei Dimensionen abbilden können, müssen die n -dimensionalen Koordinaten der Regionen möglichst sinnvoll in ein zweidimensionales Koordinatensystem übertragen werden. In *neuroVIISAS* gibt es dazu die Möglichkeit eine Hauptkomponentenanalyse durchzuführen, die eine Koordinatentransformation so durchführt, dass in der ersten Koordinatenachse die größte Varianz der Daten vorliegt. Absteigend nach der Varianz folgen dann die anderen Achsen. Die Regionen werden dann mit den Koordinaten bezüglich der ersten beiden Achsen in einem Koordinatensystem dargestellt (Abbildung 3.18). Übertragen auf eine Reduktion von drei auf zwei Dimensionen würde das bedeuten, dass man auf eine Punktwolke von der Richtung aus schaut, von der die Wolke die größte Ausdehnung hat. Es ist möglich die Parameter auszuwählen, die man in der Hauptkomponentenanalyse berücksichtigen möchte. Dabei werden die Parameterwerte gegebenenfalls möglichst sinnvoll auf das Intervall von $[0, 1]$ normiert, um verschiedene Größenordnungen bei den Parametern auszugleichen.

Eine andere Methode der Dimensionsreduktion zur Darstellung hochdimensionaler Daten ist die metrische multidimensionale Skalierung. Dabei wird versucht Objekte eines hochdimensionalen Raums in niedrigdimensionalen Räumen so anzuordnen, dass die Abstände zwischen den Objekten möglichst er-

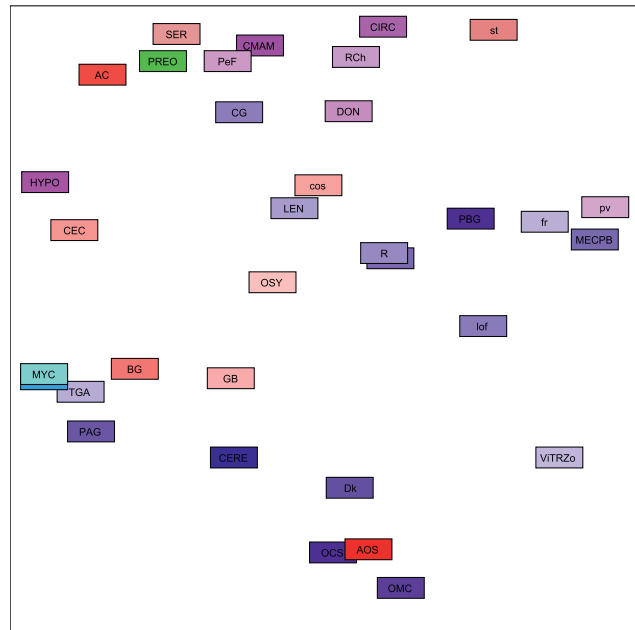


Abbildung 3.18: Darstellung der Verbindungsunterschiede der Regionen aus Abbildung 3.16 mit Hilfe der metrischen multidimensionalen Skalierung

halten bleiben. Diese Visualisierung wurde in *neuroVIISAS* implementiert, um die Regionen bezüglich ihrer Verbindungsstruktur zu vergleichen. Der Abstand zweier Regionen wurde dabei definiert, als die Anzahl der übrigen Regionen zu der genau eine der beiden Regionen eine abgehende Verbindung hat plus die Anzahl der Regionen von der in genau einer der Regionen eine Verbindung ankommt. Dazu werden die Unterschiede in den zu den Regionen gehörenden Zeilen und Spalten der Adjazenzmatrix aufsummiert, was auch als Hamming-Abstand bezeichnet wird. Die Verbindungen zwischen den beiden betrachteten Regionen werden dabei nicht berücksichtigt. Zwei Regionen, die mit genau denselben Regionen mit denselben Verbindungsrichtungen verbunden sind, haben den Abstand null. In Abbildung 3.18 sind die Regionen aus 3.16 bezüglich ihrer Verbindungsunterschiede nach einer metrischen multidimensionalen Skalierung dargestellt.

Zusätzlich zu den obigen Visualisierungen ist es auch möglich, berechnete lokale Netzwerkparameter innerhalb der Kreishierarchie, der Baumdarstellung der Hierarchie oder der 3D-Visualisierung anzeigen zu lassen. Dabei kann ein darzustellender Parameter gewählt werden, dessen Wert einer Region als prozentualer Anteil am Maximalwert visuell zugeordnet wird. In der Kreishier-

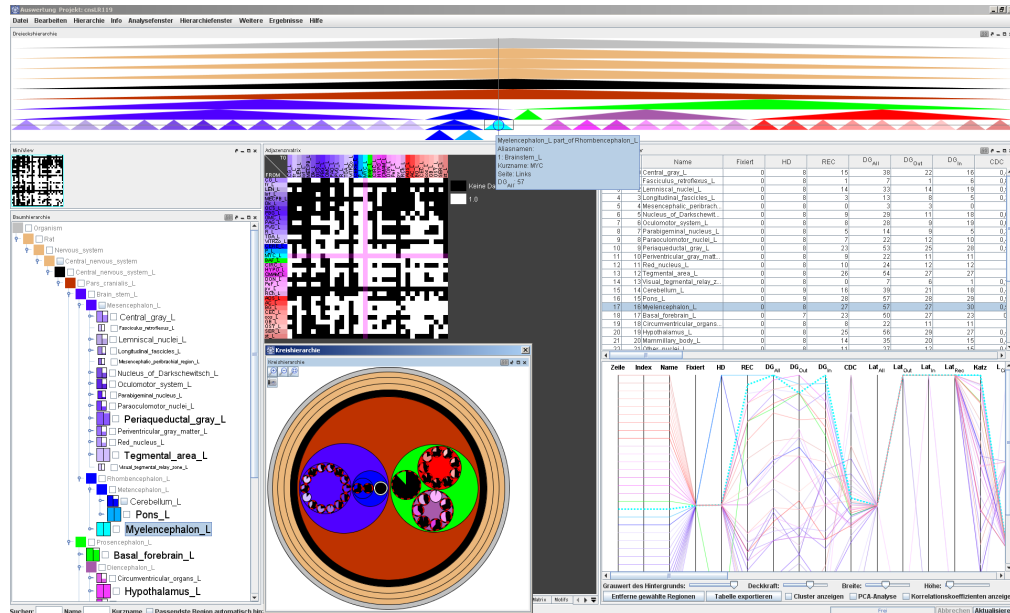


Abbildung 3.19: Vernetzte Visualisierungen und Darstellung eines lokalen Parameters

archie werden die Kreise beispielsweise tortenartig ausgefüllt, in der Baumhierarchie über die Größe der Schrift und ein anteilig ausgefülltes Kästchen und in der 3D-Visualisierung über eine Farbkodierung. In Abbildung 3.19 ist innerhalb der Kreis- und Baumhierarchie der Grad der Knoten visualisiert.

3.5.5 Vernetzung der Visualisierungen

Die Verschiedenen Visualisierungsfenster in *neuroVIISAS* sind miteinander gekoppelt, so dass es möglich ist, eine einzelne oder eine Menge von Regionen in einer Visualisierung auszuwählen, die dann in allen Visualisierungen hervorgehoben werden. In den Matrizen werden die zugehörigen Zeilen und Spalten mit transparenten Streifen überlagert. In den Darstellungen mit Knoten und Kanten werden die ausgewählten Regionen und mit diesen verbundene Kanten farblich hervorgehoben. Selbes gilt für die zu einer gewählten Region gehörenden Zeile in Tabellen. In Abbildung 3.19 ist das Myelencephalon ausgewählt und in den verschiedenen Visualisierungen hervorgehoben.

Für eine Auswahl von Regionen ist es außerdem möglich, sich den Graphen aus deren Nachbarschaftssicht anzeigen zu lassen. Abbildung 3.20 zeigt unten die Knoten der zwei ausgewählten Regionen. In der Schicht direkt über diesen

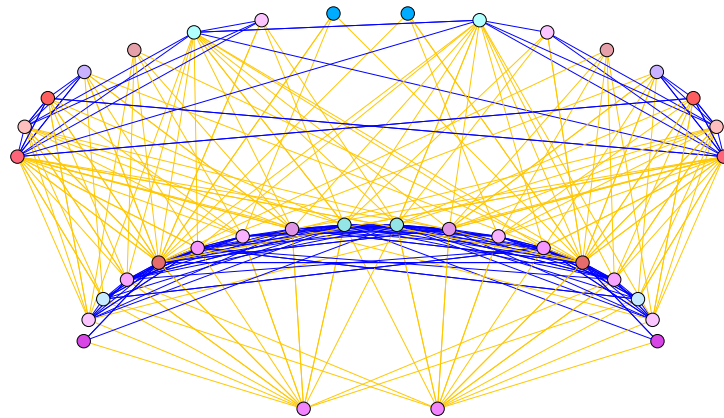


Abbildung 3.20: Nachbarschaftsdarstellung von Regionen in Bezug zu zwei Gewählten (unten)

werden die Knoten positioniert, die direkt mit den Gewählten verbunden sind. In der obersten Schicht sind die Knoten der Regionen dargestellt, die nur über die Schicht darunter mit den Gewählten verbunden sind. Je nach Graph kann es auch mehr Schichten geben, mit Regionen, die im Graphen noch weiter von den Gewählten entfernt sind. Um die Übersichtlichkeit zu erhöhen, ist es möglich Kanten innerhalb von Schichten oder zwischen einzelnen Schichten von Knoten auszublenden. Es können die Abkürzungen der Regionen angezeigt, die Knoten- und Schriftgröße variiert und statt der Bögen auch Kreise oder Ellipsen, um die gewählten Regionen herum, gewählt werden. Die Anordnung der Knoten kann optimiert werden, so dass es weniger Kantenüberschneidungen gibt.

3.5.6 Export

Um Visualisierungen aus *neuroVIISAS* in einer möglichst guten Qualität zu exportieren, wurde ein Bildexport implementiert, der neben den pixelbasierten Formaten GIF, PNG, BMP, JPG und TIF auch das Abspeichern als vektorbasierte Grafik im EPS-Format zulässt. Für die Umsetzung wurden dabei die *EpsGraphics*-Bibliotheken von Thomas Abeel verwendet. Damit ist es möglich, auch niedrig aufgelöste Bildschirmdarstellungen so zu exportieren, dass sie für den Druck von Publikationen geeignet sind. Die Tabellendaten können durch eine einfache Copy und Paste Funktionalität in externe Tabellenkalkulations-

programme übertragen oder in einigen Fällen auch mit Hilfe eines erweiterten Exports als CSV-Datei gespeichert werden.

3.6 Simulation

Die Modellierung und Simulation von Neuronen ist eine Aufgabe, der sich große Projekte wie GENESIS, NEURON oder NEST annehmen. Dabei werden mathematische Modelle erstellt, die zum Beispiel das Verhalten des Membranpotentials einer Nervenzelle mit Hilfe von Differentialgleichungssystemen beschreiben. Ein Beispiel dafür ist das *Hodgkin-Huxley-Modell*, das 1952 von Alan Lloyd Hodgkin und Andrew Fielding Huxley publiziert wurde [16]. Die Simulationssoftware NEST [11] ermöglicht es, große Populationen von Neuronen mit Hilfe verschiedener Neuronenmodelle zu simulieren. Die Verbindungsdaten aus *neuroVIISAS* können dabei als Grundlage für realistische Simulationen im Regionenmaßstab verwendet werden. Um das zu demonstrieren, wurde eine Schnittstelle implementiert, mit der es möglich ist, die Verbindungsdaten aus *neuroVIISAS* für eine Simulation mit NEST zu verwenden. NEST ist eine Skriptsprache, auf die über die Programmiersprache Python (PyNEST) NEST zugegriffen werden kann. Die Kommunikation zwischen *neuroVIISAS* und PyNEST wurde dabei mit Hilfe eines Pythonskripts realisiert, das von *neuroVIISAS* erzeugt wird und die Generierung der Neuronenpopulationen und Verbindungen und die Simulationseinstellungen beinhaltet. Ist PyNEST auf dem System installiert, kann die Simulation von *neuroVIISAS* aus gestartet und die Datei mit den Simulationsergebnissen importiert werden. Es ist aber auch möglich, die Simulation auf einem anderen System, zum Beispiel einem leistungsfähigeren Server, auszuführen. Dazu kann die Skriptdatei auf das andere System übertragen und dort von PyNEST ausgeführt werden. Die bei der Simulation erzeugte Ergebnisdatei kann dann in *neuroVIISAS* importiert und ausgewertet werden. Da das Projekt NEST regelmäßig aktualisiert wird (die aktuelle Version ist 2.6.0 vom 23. Dezember 2014), müssen bei größeren Änderungen gegebenenfalls auch die Skripte beziehungsweise der Programmcode, der die Skripte erzeugt, angepasst werden. Neu hinzugekommene Neuronenmodelle können aber ohne Codeänderung bei installiertem PyNEST abgefragt und dann in *neuroVIISAS* verwendet werden. In der implementierten Fassung

können zu einer Auswahl von Regionen für jede Region die Anzahl der Neurone, das Neuronenmodell mit den zugehörigen Parametern und die Anteile für inhibitorische und excitatorische Neurone festgelegt werden. Sind die Volumina und die Zelldichten für die Regionen bekannt, können die Neuronenzahlen auch automatisch berechnet werden. Anhand der Verbindungsdaten werden dann Verbindungen zwischen den Neuronen der Regionen erzeugt, deren Anzahl man in Abhängigkeit der Stärke der Verbindung bestimmen lassen kann. Für die Verbindungen zwischen den Neuronen innerhalb einer Region kann eine Verbindungsrate in Prozent angegeben werden, die für die inhibitorischen und excitatorischen Neurone separat wählbar ist. Für die Simulation bestimmt man eine oder mehrere Regionen, die ein erregendes Inputsignal erhalten, dessen Dauer und Frequenz variierbar sind. Weiterhin müssen Regionen ausgewählt werden, bei denen für alle Neurone die Zeitpunkte des Feuerns (Spike) aufgezeichnet und in die Ergebnisdatei geschrieben werden. Nach Wahl der Simulationsdauer, kann dann das Pythonskript erzeugt und, wenn NEST installiert ist, direkt ausgeführt werden. Bei dem Import einer Ergebnisdatei werden die Regionenauswahl und die Simulationseinstellungen aus der zugehörigen Simulation in *neuroVIISAS* übernommen, so dass die Wahl der Parameter nachvollzogen oder für eine neue Simulation angepasst werden kann.

Im Folgenden werden die Möglichkeiten der Datenauswertung dargestellt, wobei exemplarisch die Ergebnisse einer Simulation der bilateralen Regionen der Basalganglien sowie deren direkte intrinsische Verbindungen verwendet werden. Pro Gebiet wurden dafür in NEST 500 Neurone mit einem *Leaky-Integrate-And-Fire*-Neuronenmodell simuliert. Zur Visualisierung der importierten Daten wird eine Tabelle erzeugt, die unter anderem Histogramme der Spikeereignisse in frei wählbaren Intervallschritten enthält. Abbildung 3.21 zeigt eine solche Tabelle, wobei die Balkenbreite der Histogramme auf 0,5 ms gesetzt wurde. Mit Hilfe der Histogrammdarstellungen können die Aktivitäten der Regionen über die Zeit verglichen werden. In den folgenden Spalten werden die Zellzahlen, die Gesamtanzahl der Spikes aller Neurone, die mittlere Anzahl von Spikes pro Neuron oder der Zeitpunkt des ersten Spikes angegeben. Die Spalte *Pfadlänge* gibt an, wie weit eine Region von einem Input-Generator entfernt ist. In dem abgebildeten Beispiel ist ein Spike-Generator über zwei Zwischenregionen mit der *Substantia nigra reticular part* der rechten Hemi-

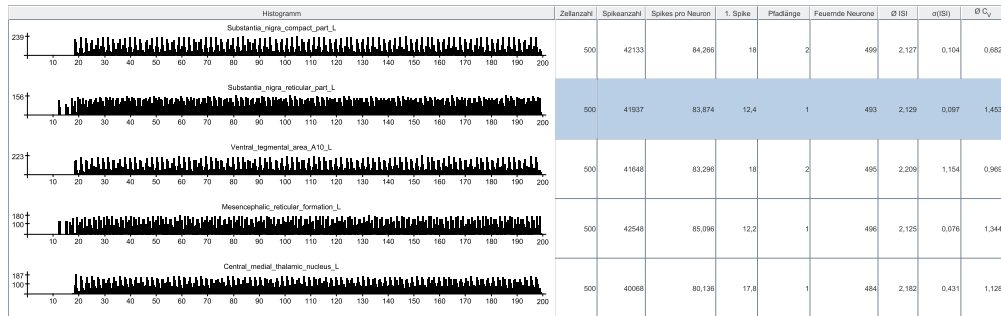
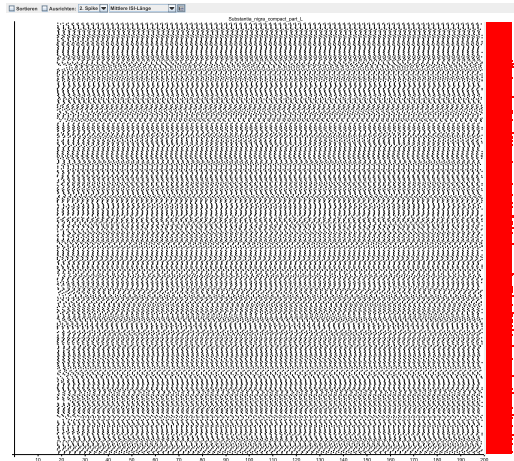


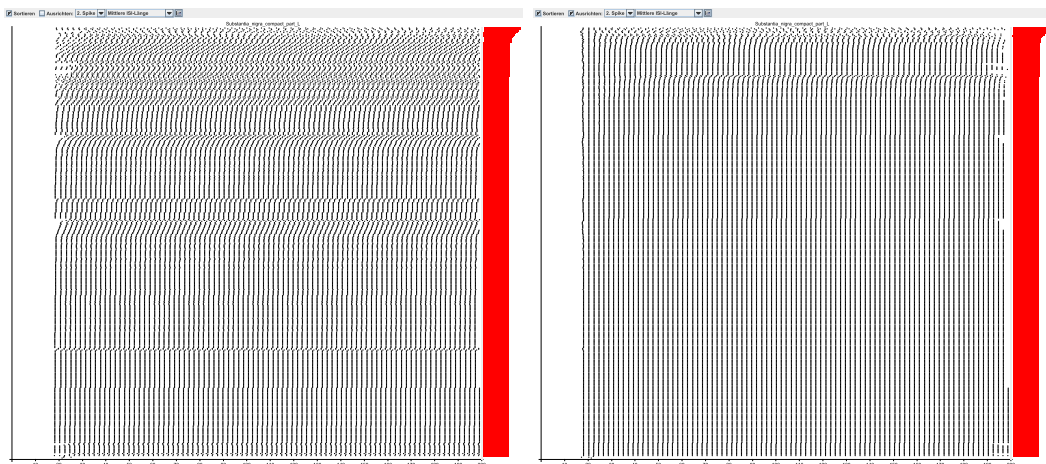
Abbildung 3.21: Ergebnisse einer Simulation mit NEST

sphäre verbunden. Das erste Spikeereignis tritt dort deshalb später auf als in den übrigen abgebildeten Regionen, die über eine Zwischenregion Input vom Spike-Generator bekommen. Für jedes Neuron wird die mittlere Zeit zwischen zwei Spikes das sogenannte Inter-Spike-Interval (ISI) berechnet. In der Spalte $\bar{\Delta} ISI$ steht dann der Mittelwert dieser Intervalle über alle Neurone einer Region. Die folgende Spalte gibt die dazu gehörende Standardabweichung an. Diese gibt Aufschluss darüber, wie stark sich die mittleren Inter-Spike-Intervalle unterschiedlicher Neurone innerhalb einer Region unterscheiden. In der letzten Spalte wird dann dargestellt, wie gleichmäßig die Inter-Spike-Intervalle bei einzelnen Neuronen sind. Dafür wird für jedes Neuron der Variationskoeffizient der ISI-Längen berechnet. Die Spalte $\bar{\Delta} C_V$ enthält dann den gemittelten Variationskoeffizienten über alle Neurone einer Region. Um die Ähnlichkeiten der Regionen bezüglich ihrer Spikeaktivitäten zu visualisieren, werden die paarweisen Ähnlichkeiten der Histogramme berechnet und in einer Matrix dargestellt. Dafür können die Histogramme am ersten Spike ausgerichtet und derart normiert werden, dass die Histogrammbalken prozentuale Spikeanzahlen, in Bezug auf die Gesamtanzahl der Spikes einer Region, enthalten.

Für jede Region kann man sich die Spike-Zeitpunkte aller simulierten Neurone dieser Region in einem Diagramm anzeigen lassen. Abbildung 3.22 zeigt die Zeitpunkte der Spike-Ereignisse von 500 simulierten Neuronen. In Teil b) wurden die Neurone nach der mittleren ISI-Länge sortiert, wodurch das relativ regelmäßige Feuern der Neurone sichtbar wird. Noch besser erkennt man das in Diagramm c), wo die Spike-Zeitpunkte zusätzlich noch am Zeitpunkt des zweiten Spikes ausgerichtet wurden. Je nach Simulation kann es sein, dass sich die Regelmäßigkeit beim Feuern erst nach mehreren Spikes einstellt. Deshalb



(a) Spikeereignisse von 500 simulierten unsortierten Neuronen



(b) Sortiert nach der mittleren ISI-Länge

(c) Sortiert und ausgerichtet

Abbildung 3.22: Spikeereignisse von 500 simulierten Neuronen

ist es möglich, die Zeitpunkte an einem der ersten fünf Spikes auszurichten. Zu beachten ist bei einer Ausrichtung, dass dabei die Informationen über die tatsächlichen Zeitpunkte der Spikes und deren Versatz zwischen verschiedenen Neuronen vernachlässigt werden. Rechts in den Diagrammen wird für jedes Neuron durch einen roten Balken die mittlere ISI-Länge dargestellt.

Die Aktivitäten der Regionen in der Simulation können in der 3D-Visualisierung als Animation dargestellt werden. Sind Simulationsergebnisse für die aktuelle Auswahl von Regionen geladen und Oberflächendaten verfügbar, so kann eine Animation gestartet werden, in der in jedem Zeitpunkt die Anzahl feuender Neurone für eine Region durch deren Helligkeit visualisiert wird. Bei

Aktivität leuchtet eine Region also auf. Alternativ können auf der Oberfläche der Regionen kleine Kugeln platziert werden, die einzelne Neurone aus der Simulation repräsentieren. In der Animation leuchten diese Kugeln dann zu den Zeitpunkten auf, in denen das zugehörige Neuron in der Simulation gefeuert hat.

Neben der Populationssimulation ist es auch möglich, mit Hilfe einer graphischen Nutzeroberfläche, manuell kleine Netzwerke, sogenannte Microcircuits zu erzeugen und zu simulieren. Dabei können einzelne Neurone als Knoten auf einer Oberfläche platziert und dann untereinander verbunden werden. Die Neuronen- und Synapsenmodelle können für jedes Objekt separat gewählt und mit unterschiedlichen Parametern versehen werden. Zusätzlich können Elemente hinzugefügt und mit den Neuronen verbunden werden, die erregende Signale senden, wie zum Beispiel eine bestimmte Spannung, Spike-Ereignisse oder Rauschen. Zum Aufzeichnen des Verhaltens der Neurone können diese dann noch mit einem Voltmeter oder Spikedetektor versehen werden, um die Membranpotenziale beziehungsweise die Zeitpunkte der Spikes aufzuzeichnen. Zu einem erzeugten Netzwerk kann dann ein Pythonskript erzeugt und mit PyNEST ausgeführt werden.

3.7 Virtuelle Schnitte

Um möglichst realistische Simulationen zu generieren, sind genaue Schätzungen von Nervenzellanzahlen und -typen für die zu simulierenden Regionen notwendig. Histologische Schnitte lassen sich mit speziellen Slide Scannern in so hoher Auflösung digitalisieren, dass einzelne Zellen detektierbar und klassifizierbar sind. Die dabei entstehenden virtuellen Schnitte können je nach Scanner und verwendeter Software in unterschiedlichen Formaten gespeichert werden. Die *OpenSlide*-Bibliotheken bieten die Möglichkeit, aus einem Java-Programm heraus auf unterschiedliche Formate zuzugreifen. In *neuroVIISAS* wurde ein Virtual-Slide Viewer implementiert, mit dem es möglich ist, hochaufgelöste histologische Schnitte anzuzeigen. Der Viewer bietet die Möglichkeit Bilder oder Bildausschnitte in herkömmlichen Bildformaten zu speichern. Große Bilder können dabei entweder herunterskaliert oder in mehreren Kacheln als Bildserie gespeichert werden.

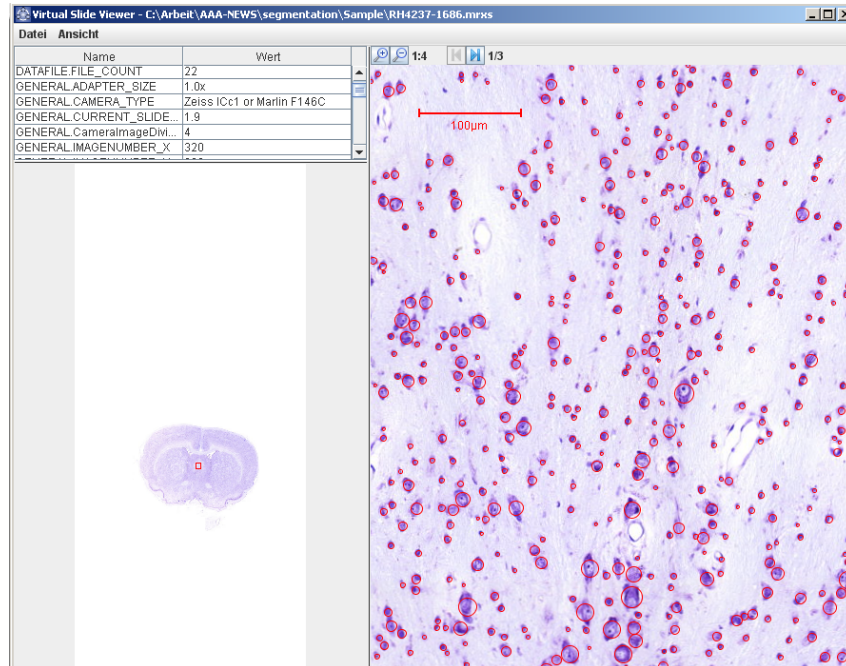


Abbildung 3.23: Virtueller Schnitt mit detektierten Zellen im Bereich des lateralen septalen Kerngebietes

In einem so exportierten virtuellen Schnitt wurden mit *Matlab* Zellen detektiert und die Zellpositionen und Durchmesser in eine Textdatei geschrieben. Die Datei mit den Zellinformationen kann in dem Viewer importiert werden, um die detektierten Zellen innerhalb des virtuellen Schnittes anzuzeigen (Abbildung 3.23). Um zu demonstrieren welche Möglichkeiten sich dadurch ergeben, wurde ein Projekt erstellt, in dem ein herunterskalierter virtueller Schnitt mit den generierten Zellinformationen und Hirnregionkonturen zusammengeführt wurden (Abbildung 3.24). Damit ist eine automatische regionenspezifische Quantifizierung von Zellen möglich.

3.8 Datenhaltung und Export

Innerhalb von *neuroVIISAS* werden alle Daten eines Projekts bis auf die Schnittbildserien und die 3D-Oberflächendaten im Speicher gehalten. Bei Bedarf werden benötigte Schnittbilder oder 3D-Oberflächen nachgeladen. Das Speichern eines Projekts wird durch Serialisierung der Java-Objektstrukturen der Daten in eine Datei realisiert. Diese wird zusammen mit den 3D-Ober-

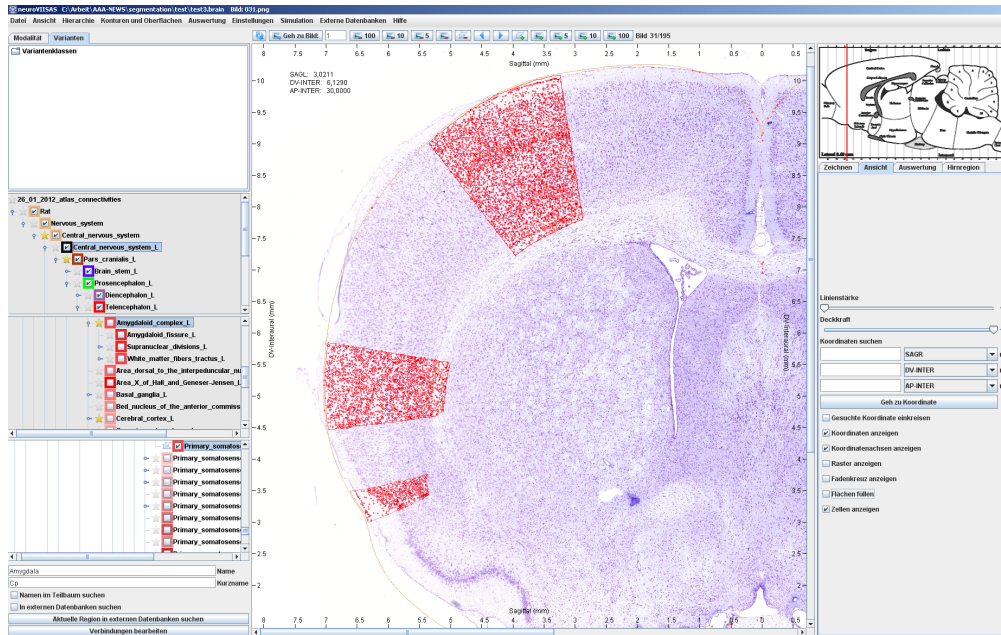


Abbildung 3.24: Detektierte Zellen in kortikalen Regionen eines Rattenhirns

flächendaten komprimiert in eine ZIP-Datei geschrieben. Diese kann einfach auf einen anderen Computer übertragen und dort wieder geöffnet werden. Eine zu dem Projekt gehörende Schnittbildserie muss in diesem Fall separat kopiert und deren Pfad beim Öffnen des Projekts angegeben werden. Da die Schnittbildserien häufig sehr groß sind und auch ausserhalb von *neuroVIISAS* verwendet werden, wurde darauf verzichtet, diese mit in die Projektdatei zu kopieren. Auch auf einen festgelegten relativen Pfad für alle Bildserien, der nach einem Kopieren des Projekts auf einen anderen Rechner die Angabe des Pfades unnötig machen würde, wurde verzichtet, um den Nutzer bei der Datenthaltung nicht unnötig einzuschränken.

Für den Export von Daten aus *neuroVIISAS* heraus gibt es verschiedene Möglichkeiten. Die Regionen- und Verbindungsdaten können mit Hilfe der Java SQL-Schnittstelle in eine SQL-Datenbank geschrieben werden, um sie zum Beispiel online zur Verfügung zu stellen. Alternativ können diese Daten auch als XML-Datei gespeichert werden, wofür der *java.beans.XMLCoder* verwendet wurde, der die Java-Objekthierarchie der Regionen und Verbindungen exportiert. Darüber hinaus gibt es noch spezielle Exportformate für *Cytoscape*, *Protégé* und *Mavisto*.

Die Verbindungsdaten können aus *neuroVIISAS* auch derart als CSV-

Tabelle exportiert werden, dass sie wieder in einer anderen Projektdatei importierbar sind. Es ist möglich alle Verbindungen zu exportieren oder mit Hilfe eines Auswahlfensters bestimmte Verbindungen für den Export zu selektieren. Dafür steht eine Tabelle zur Verfügung, in der alle Verbindungen mit ihren Gewichten und Publikationen angezeigt werden. Eine Möglichkeit der Selektion ist das manuelle Entfernen der Verbindungen, die nicht exportiert werden sollen. Darüber hinaus gibt es aber auch die Option, Filter anzulegen, mit deren Hilfe eine Auswahl von Verbindungen mit bestimmten Eigenschaften generiert werden kann. Filterkriterien sind dabei die Publikationen, die Gewichte sowie Ausgangs- oder Zielregionen der Verbindungen, wobei nicht nur eine Region sondern auch ein Teilbaum der Hierarchie gewählt werden kann. Angelegte Filter können dann mit den logischen Operatoren UND, ODER und NICHT miteinander kombiniert und auf die Menge der Verbindungen angewendet werden, um die gewünschten Verbindungen für einen Export zu selektieren.

3.9 Praktischer Einsatz

Das Programm *neuroVIISAS* ist parallel zu seiner Entwicklung im Institut für Anatomie der Universität Rostock im Einsatz. Die Hauptanwendung ist das Sammeln und Auswerten von Verbindungsinformationen aus Tract-Tracing Studien über die Ratte. Im Rahmen dieser Anwendung sind mehrere Publikationen und Doktorarbeiten entstanden. Die bisher vom Team von Prof. Dr. Oliver Schmitt in *neuroVIISAS* zusammengetragenen Konnektomdaten der Ratte wurden in eine MySQL-Datenbank exportiert, auf die über ein Webinterface auf der Seite <http://neuroviisas.med.uni-rostock.de/connectome/> zugegriffen werden kann. Diese Datenbank ist weltweit die umfangreichste verfügbare Zusammenstellung von Verbindungsinformationen des Rattenhirns. Sie enthält (Stand 20. April 2015) circa 47500 Regionen und 600000 Verbindungsbeschreibungen aus fast 850000 Experimenten. Auf die Daten kann über eine Suche nach einer Region oder der Suche nach einer Publikation beziehungsweise einem Author zugegriffen werden. Auch außerhalb des Instituts für Anatomie in Rostock findet das Programm Beachtung. Es wurde im Laufe des letzten Jahres über 100 mal vom *neuroVIISAS*-Server heruntergeladen.

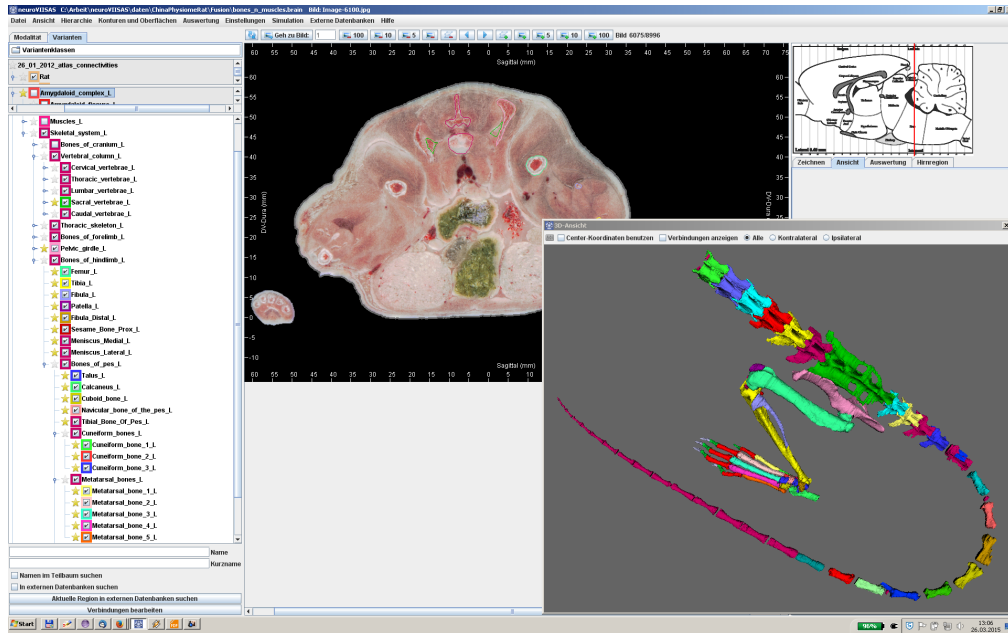


Abbildung 3.25: Kartierung des Skelettsystems der Ratte mit *neuroVIISAS*

Die Möglichkeit der Kartierung und Visualisierung von Strukturen in *neuroVIISAS* beschränkt sich nicht auf Regionen des Nervensystems. In zwei aktuellen Projekten werden das Skelett- und das Muskelsystem der Ratte so kartiert, dass jeder einzelne Knochen beziehungsweise Muskel als eine Struktur auswählbar und visualisierbar ist. Der Ausgangspunkt dieser Projekte ist jeweils eine segmentierte Schnittbildserie des kompletten Systems, für die die Konturen generiert und in *neuroVIISAS* importiert wurden. Die Konturen in den einzelnen Schnittbildern werden dann manuell einer Struktur zugeordnet. Abbildung 3.25 zeigt einen Teil des bisher mit *neuroVIISAS* kartierten Skelettsystems der Ratte. Links ist die zugehörige Hierarchie des Skelettsystems abgebildet, bei der jedes Blatt einen einzelnen Knochen darstellt. In Abbildung 3.26 wurde das kartierte Zentralnervensystem der Ratte in einem Projekt mit dem Skelettsystem und Organen fusioniert.

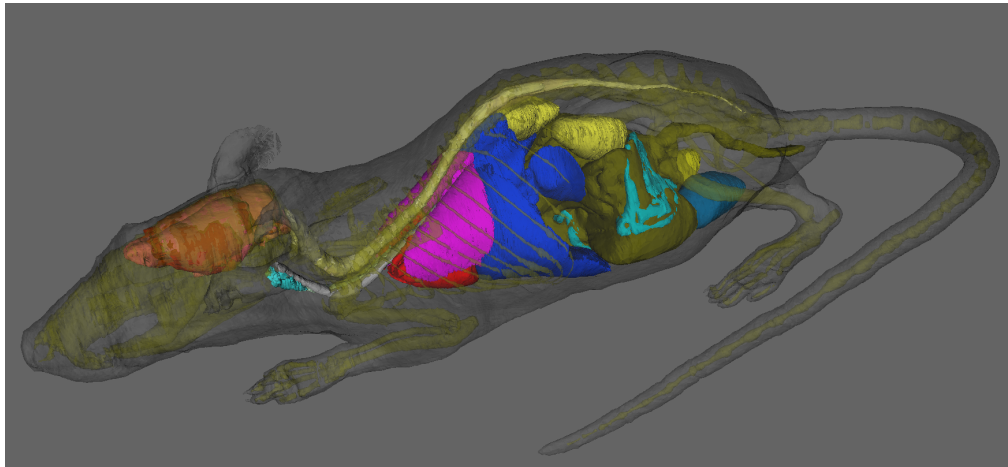


Abbildung 3.26: Integration von Skelett, Nervensystem und Organen in *neuroVIISAS*

Kapitel 4

Diskussion

4.1 Zusammenfassung

Gegenstand dieser Arbeit ist die Entwicklung der Software *neuroVIISAS*, mit deren Hilfe es möglich ist, neuronale Verbindungsdaten, regionspezifische Daten, Bilddaten und räumlichen Informationen in einem einzigen System zu integrieren, darzustellen und zu analysieren. Es wurden eine Vielzahl von Analysewerkzeugen und Visualisierungsmethoden implementiert. Import- und Exportschnittstellen ermöglichen es, Daten aus unterschiedlichen Quellen und in unterschiedlichen Formaten zu importieren und zu exportieren. Das Programm ist plattformunabhängig und kann unter Windows, Linux und Mac OS eingesetzt werden. Der Einsatzschwerpunkt von *neuroVIISAS* ist die Analyse des Nervensystems der Labormaus, aber seine Anwendung ist nicht auf Nervensysteme oder eine bestimmte Spezies beschränkt. Es wurden auch Projekte mit verfügbaren Konnektomen anderer Spezies in *neuroVIISAS* erstellt. Regionen, mit unterschiedlichen hierarchischen Gliederungen können angelegt und mit textbasierten und räumlichen Informationen, sowie zugehörigen Bilddaten verknüpft werden. Es ist möglich, Verbindungen zwischen diesen Regionen manuell anzulegen oder zu importieren, wobei Widersprüche bei der Existenz oder Stärke dieser Verbindungen abbild- und auswertbar sind. Darüber hinaus ist es möglich, die gesammelten Informationen für eine populationsbasierte neuronale Simulation mit der Simulationssoftware NEST zu verwenden. Das Programm kann in den verschiedenen Versionen unter <http://neuroviisas.med.uni-rostock.de/versions> zusammen mit Testdaten heruntergeladen werden. Zu-

sätzlich gibt es die Möglichkeit *neuroVIISAS* per Webstart direkt vom Browser aus zu starten, wobei in diesem Fall nicht alle Funktionen zur Verfügung stehen.

4.2 Anwendungen

Das Programm wird im Institut für Anatomie der Universität Rostock erfolgreich bei der Zusammenstellung und Analyse des Rattenkonnectoms eingesetzt. Die dabei mit *neuroVIISAS* gesammelten Konnectomdaten sind über eine Onlinedatenbank auf der Seite <http://neuroviisas.med.uni-rostock.de/connectome> öffentlich zugänglich.

Eine untersuchte Fragestellung ist, ob sich aus der konnektionalen Struktur vor und nach einem experimentellen Infarkt, wie der Okklusion der medialen cerebralen Arterie (MCA) bei der Ratte, Vorhersagen über die Aktivitäten der Regionen machen lassen [35]. Dazu wurden bei Verhaltensexperimenten vor und nach Okklusion der MCA die cFos-Expressionen der Regionen als Maß für deren neuronale Aktivität bestimmt. Die Änderungen der cFos-Expressionen können dann mit den Änderungen von lokalen Netzwerkparametern aus *neuroVIISAS* korreliert werden, um mögliche Zusammenhänge zwischen der Netzwerkstruktur und der Aktivität aufzudecken.

Zusätzlich zu der Generierung von Konnectomen, werden mit *neuroVIISAS* auch räumliche Kartierungen von anatomischen Strukturen erstellt. Neben den Strukturen des Nervensystems, werden dabei aktuell auch die einzelnen Muskeln und Knochen der Ratte kartiert.

4.3 Ausblick

Bei der Verwendung von *neuroVIISAS* hat sich gezeigt, dass mit zunehmenden Datenmengen das aktuelle Konzept der Haltung aller Daten im Speicher an seine Grenzen stößt. Ein aktuelles Projekt enthält über 47000 Regionen in der Hierarchie und über eine halbe Millionen Verbindungen. Langfristig sollte deshalb die Datenhaltung so umgestellt werden, dass weniger Daten schnell verfügbar im Speicher gehalten und nur bei Bedarf nachgeladen werden. Mögliche auszulagernde Daten sind die Kontur- und die Verbindungsinformationen.

Auch bei der Analyse der Daten werden Grenzen hinsichtlich des verfügbaren Speichers und der Rechenzeit erreicht. Sollen größere Netzwerke untersucht werden, müssen Strategien entwickelt werden, um den Speicherbedarf zu reduzieren und mittels Parallelisierung der Algorithmen die Rechenzeit zu reduzieren.

An zusätzlichen Funktionen sind die Implementierung einer Bildregistrierung geplant, um histologische Schnittbildserien ohne zusätzliche Werkzeuge in *neuroVIISAS* registrieren zu können. Hier wird vor allem die Koregistrierung von Serienschnittbildern mit Atlasdaten im Fokus stehen, um regionenspezifische Analysen durchführen zu können. Eine geplante Anwendung ist die Zellerkennung und -zählung in hochaufgelösten virtuellen Schnitten. Auch die Bestimmung von Rezeptordichten (Autoradiographie) oder Proteinexpressionen (Immunhistochemie) sind möglich. Das Herstellen von Beziehungen zwischen verschiedenen Bildmodalitäten, durch deren Koregistrierung, erfordert eine Erweiterung des bisherigen Konzepts der Bindung der Konturinformationen an eine vorliegende Schnittbildserie. Bisher kann innerhalb einer Projektdatei nur eine einzige Schnittbildserie angegeben werden, die die Dimension und Auflösung der Konturdaten bestimmt. Unterschiedliche Bildmodalitäten müssen bisher in verschiedenen Projektdateien verwaltet werden. Ein Lösungsansatz, um diese Beschränkung aufzuheben, ist ein unabhängiges Koordinatensystem, wie es bisher optional auch schon definierbar ist. Sind die verschiedenen Bildmodalitäten in einer Projektdatei integriert und durch das gemeinsame Koordinatensystem verbunden, so sind die Konturinformationen einfach auf die unterschiedlichen Bildmodalitäten übertragbar.

Die neuronalen Verbindungen in *neuroVIISAS* enthalten zur Zeit keine Informationen über ihren räumlichen Verlauf und werden in den 3D-Visualisierungen linear dargestellt. In realen Nervensystemen ist der Verlauf im Allgemeinen nichtlinear, so dass sich räumliche Distanzen für die Signalübertragung nicht einfach als euklidische Distanzen der beteiligten Regionen ableiten lassen. In einer Kooperation mit Santiago Canals Gamoneda [25] [32] sollen traktographische Analysen von diffusionsgewichteten Kleintierscans genutzt werden, um die Konnektivitäten mit ihren räumlichen Verläufen in Beziehung zu setzen.

Die Verwaltung der Hierarchievarianten hat sich in der Praxis als nicht intuitiv herausgestellt. Hier können neue Konzepte entwickelt werden, die das

Erstellen und Pflegen verschiedener Hierarchievarianten erleichtern.

Die Flexibilität des Fenstermanagements, das mit Hilfe von *InfoNode Docking Windows* in der *Erweiterten Konnektivitäten-Analyse* Verwendung findet, könnte auch für das Hauptfenster von *neuroVIISAS* übernommen werden, so dass der Nutzer die Anordnung einzelner Fensterelemente individueller gestalten kann.

Die Clusterung von Regionen in Abhängigkeit ihrer Eigenschaften ist ein Werkzeug, das dabei helfen kann, Strukturen in großen Netzwerken zu finden. Zusätzlich zu den implementierten Verfahren können auch Clusterungen von Regionen bezüglich einer beliebigen Kombination von lokalen Parametern sinnvoll sein. In diesem Zusammenhang könnte auch das sogenannte *Consensus clustering* [21] Anwendung finden, bei der die Ergebnisse mehrerer verschiedener Clusterungen zu einer neuen Clusterung kombiniert werden.

Im Gegensatz zur Benutzeroberfläche wurde die Hilfefunktionalität in *neuroVIISAS* nicht mehrsprachig konzipiert. Hier könnte ein neuer flexiblerer Ansatz entwickelt und implementiert werden, der ein einfaches Hinzufügen neuer Sprachen für die Hilfe ohne Programmcodeänderungen ermöglicht.

Aktuell wird mit *neuroVIISAS* das muskuloskelettale System der Ratte digitalisiert, welches im Anschluss mit den neuronalen Atlasdaten und den Konnektomdaten fusioniert werden soll. Das so geschaffene Modell der Ratte erlaubt nicht nur die Simulation von Neuronenpoulationen, sondern könnte auch, zum Beispiel mit *OpenSim*, für eine Simulation von Bewegungsabläufen genutzt werden. Mit diesem Ansatz ließe sich ein Prozessfluss von der neuronalen Ebene zum muskuloskelettalen System abbilden.

Literaturverzeichnis

- [1] A Paul Alivisatos, Miyoung Chun, George M Church, Ralph J Greenspan, Michael L Roukes, and Rafael Yuste. The brain activity map project and the challenge of functional connectomics. *Neuron*, 74(6):970–974, 2012.
- [2] Xueling Bai, Li Yu, Qian Liu, Jie Zhang, Anan Li, Dao Han, Qingming Luo, and Hui Gong. A high-resolution anatomical rat atlas. *Journal of anatomy*, 209(5):707–708, 2006.
- [3] Rembrandt Bakker, Paul Tiesinga, and Rolf Kötter. The Scalable Brain Atlas: instant web-based access to public brain atlases and related content. *arXiv preprint arXiv:1312.6310*, 2013.
- [4] Mathieu Bastian, Sebastien Heymann, Mathieu Jacomy, et al. Gephi: an open source software for exploring and manipulating networks. *ICWSM*, 8:361–362, 2009.
- [5] Mihail Bota and Larry W Swanson. BAMS neuroanatomical ontology: design and implementation. *Frontiers in Neuroinformatics*, 2(2):10.3389/neuro.11.002.2008, 2008.
- [6] LM Eisenman and CR Noback. The ponto-cerebellar projection in the rat: differential projections to sublobules of the uvula. *Exp Brain Res*, 38(1):11–17, 1980.
- [7] Ernesto Estrada and Naomichi Hatano. Communicability in complex networks. *Physical Review E*, 77(3):036111, 2008.
- [8] Ernesto Estrada and Juan A Rodriguez-Velazquez. Subgraph centrality in complex networks. *Physical Review E*, 71(5):056103, 2005.

- [9] Giorgio Fagiolo. Clustering in complex directed networks. *Physical Review E*, 76(2):026107, 2007.
- [10] Robert W Floyd. Algorithm 97: Shortest Path. *Commun. ACM*, 5(6):345, June 1962.
- [11] Marc-Oliver Gewaltig and Markus Diesmann. NEST (NEural Simulation Tool). *Scholarpedia*, 2(4):1430, 2007.
- [12] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.
- [13] Patric Hagmann. *From diffusion MRI to brain connectomics*. PhD thesis, Institut de traitement des signaux PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES PAR Docteur en médecine, Université de Lausanne, 2005.
- [14] Suzana Herculano-Houzel. The remarkable, yet not extraordinary, human brain as a scaled-up primate brain and its associated cost. *Proceedings of the National Academy of Sciences*, 109(Supplement 1):10661–10668, 2012.
- [15] Suzana Herculano-Houzel and Roberto Lent. Isotropic fractionator: a simple, rapid method for the quantification of total cell and neuron numbers in the brain. *The Journal of neuroscience*, 25(10):2518–2521, 2005.
- [16] Alan L Hodgkin and Andrew F Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–544, 1952.
- [17] Mark D Humphries and Kevin Gurney. Network "small-world-ness": a quantitative method for determining canonical network equivalence. *PloS one*, 3(4):e0002051, 2008.
- [18] Eugene M Izhikevich. *Dynamical systems in neuroscience*. MIT press, 2007.
- [19] Marcus Kaiser. A tutorial in connectome analysis: Topological and spatial features of brain networks. *Neuroimage*, 57(3):892–907, Aug 2011.

- [20] Eric R Kandel, Henry Markram, Paul M Matthews, Rafael Yuste, and Christof Koch. Neuroscience thinks big (and collaboratively). *Nature Reviews Neuroscience*, 14(9):659–664, 2013.
- [21] Andrea Lancichinetti and Santo Fortunato. Consensus clustering in complex networks. *Scientific reports*, 2, 2012.
- [22] Vito Latora and Massimo Marchiori. Efficient Behavior of Small-World Networks. *Phys. Rev. Lett.*, 87:198701, Oct 2001.
- [23] Isabel Lockard. *Desk reference for neuroscience*. Springer-Verlag, 1992.
- [24] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM siggraph computer graphics*, volume 21, pages 163–169. ACM, 1987.
- [25] MA Martínez-Martínez, J Pacheco-Torres, V Borrell, and S Canals. Phenotyping the central nervous system of the embryonic mouse by magnetic resonance microscopy. *NeuroImage*, 97:95–106, 2014.
- [26] Shawn Mikula and Winfried Denk. High-resolution whole-brain staining for electron microscopic circuit reconstruction. *Nat Methods*, Apr 2015.
- [27] Andrew Nere and Mikko Lipasti. Cortical architectures on a gpgpu. In *Proceedings of the 3rd workshop on general-purpose computation on graphics processing units*, pages 12–18. ACM, 2010.
- [28] Mark EJ Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133, 2004.
- [29] Seung Wook Oh, Julie A Harris, Lydia Ng, Brent Winslow, Nicholas Cain, Stefan Mihalas, Quanxin Wang, Chris Lau, Leonard Kuan, Alex M Henry, et al. A mesoscale connectome of the mouse brain. *Nature*, 508(7495):207–214, 2014.
- [30] Jonathan Ozik, Brian R Hunt, and Edward Ott. Growing networks with geographical attachment preference: Emergence of small worlds. *Physical Review E*, 69(2):026108, 2004.

- [31] G Paxinos and C Watson. *The rat brain in stereotaxic coordinates*. Amsterdam, Elsevier Academic Press, 6th edition, 2007.
- [32] Saulo DS Reis, Yanqing Hu, Andrés Babino, José S Andrade Jr, Santiago Canals, Mariano Sigman, and Hernán A Makse. Avoiding catastrophic failure in correlated networks of networks. *Nature Physics*, 2014.
- [33] Mikail Rubinov and Olaf Sporns. Complex network measures of brain connectivity: uses and interpretations. *Neuroimage*, 52(3):1059–1069, 2010.
- [34] Rony H Salloum, Guoyou Chen, Liliya Velet, Nauman F Manzoor, Rachel Elkin, Grahame J Kidd, John Coughlin, Christopher Yurosko, Stephanie Bou-Anak, Shirin Azadi, et al. Mapping and morphometric analysis of synapses and spines on fusiform cells in the dorsal cochlear nucleus. *Frontiers in systems neuroscience*, 8:167–167, 2013.
- [35] O Schmitt, S Badurek, W Liu, Y Wang, G Rabiller, J He, P Eipert, and J Liu. cFos expression and correlation of local network parameters of an medial cerebral artery occlusion model. Brain 2015, XXVIIth international symposium on cerebral blood flow, metabolism and function, 27th-30th of june, Vancouver., 2015.
- [36] Oliver Schmitt, Peter Eipert, Richard Kettlitz, Felix Leßmann, and Andreas Wree. The connectome of the basal ganglia. *Brain Structure and Function*, pages 1–62, 2014.
- [37] Oliver Schmitt, Peter Eipert, Konstanze Philipp, Richard Kettlitz, Georg Fuellen, and Andreas Wree. The intrinsic connectome of the rat amygdala. *Front Neural Circuits*, 6:81, 2012.
- [38] Michael E Smoot, Keiichiro Ono, Johannes Ruscheinski, Peng-Liang Wang, and Trey Ideker. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics*, 27(3):431–432, 2011.
- [39] Johannes Sorger, K Buhler, Florian Schulze, Tianxiao Liu, and Barry Dickson. neuroMAP-Interactive graph-visualization of the fruit fly’s neural circuit. In *Biological Data Visualization (BioVis), 2013 IEEE Symposium on*, pages 73–80. IEEE, 2013.

- [40] Olaf Sporns. Graph theory methods for the analysis of neural connectivity patterns. In *Neuroscience databases*, pages 171–185. Springer, 2003.
- [41] Olaf Sporns. *Networks of the Brain*. MIT press, 2011.
- [42] Olaf Sporns, Giulio Tononi, and Rolf Kötter. The human connectome: a structural description of the human brain. *PLoS computational biology*, 1(4):e42, 2005.
- [43] L Swanson. *Brain Maps: Structure of the Rat Brain*. Amsterdam, Elsevier Science Ltd, 1st edition, 1992.
- [44] Erik Virtel. Motivanalyse in Graphen. Diplomarbeit, University of Rostock, Germany, 2009.
- [45] JG White, E Southgate, JN Thomson, and S Brenner. The structure of the nervous system of the nematode *Caenorhabditis elegans*: the mind of a worm. *Phil. Trans. R. Soc. Lond*, 314:1–340, 1986.

Anhang A

Eigene Publikationen

Originalarbeiten

- Schmitt O, Eipert P (2012) neuroVIISAS: Approaching multiscale simulation of the rat connectome. *Neuroinformatics* 10: 243-267.
- Schmitt O, Eipert P, Philipp K, Kettlitz R, Fuellen G, Wree A (2012) The intrinsic connectome of the rat amygdala. *Front Neural Circuits* 6:81.
- Schmitt O, Eipert P, Kettlitz R, Lessmann F, Wree A (2014) The connectome of the basal ganglia. *Brain Struct Funct*, im Druck.

Reviews

- Schmitt O, Eipert P Bildgebung in den Neurowissenschaften. In: 125 Jahre Anatomische Gesellschaft (1886-2011) Jubiläumsausgabe. Ed. Wolfgang Kühnel. 153-158.

Posterbeiträge

- Schmitt O, Eipert P, Viertel V, Philipp C (2009) Connectivities and structures of the rat central nervous system. Proceedings of the 8th Göttingen Meeting of the German Neuroscience Society (25.3.-29.3.2009), S. 1238.
- Schmitt O, Eipert P, Philipp K, Rolfs A, Wree A (2009) Connectivity analysis a generalized brain atlas framework. Third Vogt-Brodmann Symposium, Jülich 4.12.-6.12.2009, Germany.

- Schmitt O, Eipert P, Philipp K (2010) Intrinsic connectivity analysis of the rat amygdala. Annual Meeting of the Anatomical Society, 26.3.-29.3.2010, Hamburg.
- Schmitt O, Eipert P, Wree A, Schmitz K-P (2010) Interfacing neuroVIAS and NEST. 19th Annual Computational Neuroscience Meeting, July 24th-30th 2010, San Antonio, Texas.
- Schmitt O, Eipert P, Philipp K, Rolfs A, Wree A, Schmitz K-P (2010) Connectivity analysis a generalized brain atlas framework. Profillinie 4 Workshop, Rostock 30.04.-1.05.2010, Germany.
- Schmitt, O., Eipert, P., Wree, A., Schmitz, K.-P. (2011) Spike distributions of a population based hierarchical network of the rat amygdaloid complex. BMC Neurosci 12(Suppl 1): 1-2.
- Schmitt, O., Eipert, P., Wree, A., Schmitz, K.-P. (2011) Rate dynamics in highly structured populations models of the rat amygdala. Proceedings of the 9th Göttingen Meeting of the German Neuroscience Society (23.3.-27.3.2011), S. 194.
- Schmitt, O., Eipert, P., Wree, A., Schmitz, K.-P. (2011) Approaching the complete rat central nervous system connectom. Wiring the Brain: Making Connections. International Conference. 12.4-15.4.2011, Wicklow, Ireland.
- Schmitt O, Eipert P, Wree A, Schmitz K-P (2011) Connectomics of the rat central nervous system in neuroVIISAS. NeuroInformatics 2011, Boston 4.9.-6.9.2011, Boston, USA, Abstract Book: 164-165.
- Schmitt O, Philipp K, Kettlitz R, Eipert P, Wree A (2012) The connectome of the rat amygdaloid complex. 107. Annual Meeting of the Anatomical Society, 23.3.-26.3.2012, Frankfurt.
- Schmitt O, Eipert P, Kettlitz R, Philipp K, Wree A (2012) Quantitative network features of the basal ganglia and connectome based population simulations. Bernstein Conference München, 12.9.-14.9.2012.

- Schmitt O, Eipert P, Philipp K, Kettlitz R, Wree A (2012) The intrinsic and extrinsic connectome of subregions of the basal ganglia. Neuroinformatics 2012 München, 10.9.-12.9.2012.
- Schmitt O, Eipert P, Klünker A-K, Kettlitz R, Morawska P, Wree A (2013) The connectome of the spinal cord of the rat. T26-8D, 10th Göttingen Meeting of the German Neuroscience Society 2013.
- Bota M, Schmitt O, Eipert P (2013) A case study of functional-structural modules revealed by network analysis: the rat BST-amygdala macro-connectome. INCF September 2013, Stockholm.
- Schmitt O, Eipert P, Morawska P, Klünker A-C, Meinhardt J, Lessmann F, Beier J, Kadir K, Karnitzki A, Jenssen J, Kuch L, Sellner L, Wree A (2014). Integrating the peripheral and central nervous system connectomes of the rat at micro-, meso- and macroscale levels. “Micro-, Meso- and macro- connectomics of the brain“, Fondation Ipsen, Paris May 5, 2014.
- Schmitt O, Eipert P, Hoffmann R, Morawska P, Klünker A-C, Meinhardt J, Lessmann F, Beier J, Kadir K, Karnitzki A, Jenssen J, Kuch L, Sellner L, Wree A (2014) Central and peripheral monosynaptic, polysynaptic and collaterals connectivity in the rat. INCF conference, Leiden 25.8.-27.8.2014.
- Schmitt O, Eipert P, Klünker A-C, Meinhardt J, Lessmann F, Beier J, Kadir K, Karnitzki A, Jenssen J, Kuch L, Wree A (2015) The rat connectome: All known connections of the rat nervous system in one database. 11th Göttingen Meeting of the German Neuroscience Society 2015.
- Schmitt O, Badurek S, Liu W, Wang Y, Rabiller G, He J, Eipert P, Liu J (2015) cFos expression and correlation of local network parameters of an medial cerebral artery occlusion model. Brain 2015, XXVIIth international symposium on cerebral blood flow, metabolism and function, 27th-30th of June, Vancouver.

Anhang B

Erklärung

Hiermit versichere ich an Eides Statt, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe, bis auf die in der Danksagung aufgeführten Personen, angefertigt habe und andere als die von mir angegebenen Hilfsmittel und Quellen nicht benutzt habe. Die in den verwendeten Werken wörtlich oder inhaltlich entnommenen Stellen wurden als solche kenntlich gemacht.

Diese Dissertation wurde bisher keiner anderen Prüfungsbehörde in gleicher oder ähnlicher Form vorgelegt.

Ich erkläre, dass ich noch kein Promotionsverfahren an einer anderen wissenschaftlichen Einreichung erfolglos beendet habe und dass keine Aberkennung eines erworbenen Doktorgrades vorliegt.

Rostock, den 28. Oktober

2015

Ort, Datum

Peter Eipert